



Sorbonne Université

École Doctorale Informatique, Télécommunications et Electronique (ED130)

Laboratoire LIP6

Explications causales pour l'argumentation abstraite : cadres classique, temporel et bipolaire pondéré

Par Yann MUNRO

Thèse de doctorat en Informatique

Thèse soutenue le 10/12/2024

Devant le jury composé de :

Elise	BONZON	Maître de conférences, HDR	Rapporteuse
Andreas	HERZIG	Directeur de recherche	Rapporteur
Jean-Guy	MAILLY	Professeur junior	Examineur
Pierre	MARQUIS	Professeur	Examineur
Nicolas	MAUDET	Professeur	Président du jury
Isabelle	BLOCH	Professeure	Directrice
Mohamed	CHETOUANI	Professeur	Encadrant
Marie-Jeanne	LESOT	Professeure	Encadrante
Catherine	PELACHAUD	Directrice de recherche	Encadrante

Remerciements

Je remercie en premier lieu les membres de mon jury de thèse pour avoir accepté d'en faire partie, mais également pour les discussions et les questions qui ont émergé à la suite de ma soutenance. Merci à mes rapporteurs, Andreas Herzig et Elize Bonzon, pour leurs retours détaillés sur mon manuscrit.

Merci à mes différents encadrants de thèse : Isabelle Bloch, Marie-Jeanne Lesot, Catherine Pelachaud et et Mohamed Chetouani pour leur accompagnement et leur aide tout au long de ma thèse. Merci de m'avoir fait découvrir le monde de la recherche et en particulier celui de l'IA symbolique mais surtout merci pour votre bienveillance et votre patience !

Merci ensuite à Gauvain Bourgne et Camilo Sarmiento pour cette collaboration durant ma thèse et celle de Camilo. Cela m'a permis de découvrir plein de nouvelles choses, mais m'a aussi apporté des nouvelles idées et directions dans mes travaux.

Comment également ne pas remercier mes collègues de l'équipe LFI étendue, en particulier les doctorants, stagiaires et post-doctorants qui ont permis de créer un super environnement de travail, sain et sans prise de tête. C'est en grande partie grâce à vous qu'il était aussi facile de venir tous les matins au laboratoire.

Enfin, un immense merci à ma famille et mes ami(e)s pour leur soutien pendant ces années à Paris, qui, bien que loin du Sud et du soleil, auront été très lumineuses.

Table des matières

Résumé	1
Introduction	5
1 Contexte et état de l'art	11
1.1 Système d'argumentation abstraite	11
1.1.1 Définitions générales	12
1.1.2 Sémantiques à base d'extension	14
1.1.3 Sémantiques construites par étiquetage	16
1.2 Causalité en argumentation abstraite	18
1.2.1 Discussion sur la définition de causalité effective	19
1.2.2 Causalité effective en argumentation abstraite	21
1.3 Expliquer le statut d'acceptabilité d'un argument	22
1.3.1 Explications en intelligence artificielle : le domaine de l'IA explicable	22
1.3.2 ArgXAI, <i>argumentative</i> XAI	23
1.4 Temporalité en argumentation abstraite	25
1.5 Bilan	27
2 Modèles structurels causaux pour l'argumentation abstraite	29
2.1 Introduction	30
2.2 Rappels préliminaires sur les modèles structurels causaux	30
2.2.1 Définition d'un modèle structurel causal	30
2.2.2 Cause effective	32
2.2.3 Cause suffisante	33
2.2.4 Explication	35
2.2.5 Choix du modèle structurel causal à adopter	36
2.3 Passage des AAF aux graphes causaux argumentatifs	37
2.3.1 Graphes structurels causaux argumentatifs (GCA)	37
2.3.2 Transformation proposée $\mathcal{T}_{AFa2GCA}$	38
2.3.3 Exploitation des GCA pour la génération d'explications	39
2.4 Passage des GCA aux AAF	40

2.4.1	Transformation proposée $\mathcal{T}_{GCA2AFa}$	40
2.4.2	Théorème d'équivalence entre AAF et GCA	41
2.4.3	Équivalence et inverse à gauche	42
2.5	Conclusion	43
3	Temporalité en argumentation abstraite acyclique	45
3.1	Introduction	46
3.2	Préliminaires sur les langages d'action	46
3.2.1	Définition	46
3.2.2	Causalité effective	50
3.3	Passage des AAF acycliques au langage d'action	52
3.3.1	Instanciation du contexte κ	53
3.3.2	Modification de la sémantique	57
3.3.3	Implémentation en Answer Set Programming (ASP)	58
3.4	Étude des propriétés formelles de la transformation $\mathcal{T}_{AFa2AL}$	59
3.4.1	Unicité des traces	59
3.4.2	Terminaison de $\mathcal{T}_{AFa2AL}$	60
3.4.3	Correction et complétude de $\mathcal{T}_{AFa2AL}$	61
3.5	Discussion sur la génération d'explications	62
3.5.1	Représentation graphique comme support de l'explication	62
3.5.2	Construction de la chaîne causale	65
3.5.3	Vers l'extraction d'explications adaptées à partir d'une chaîne causale	67
3.6	Conclusion	68
4	Systèmes d'argumentation abstraite cycliques et langage d'action	71
4.1	Introduction	72
4.2	Passage des AAF au langage d'action	72
4.2.1	Instanciation du contexte	72
4.2.2	Modification de la sémantique du langage d'action	77
4.3	Étude formelle de la transformation proposée	78
4.3.1	Unicité des traces et cohérence des fluents	78
4.3.2	Terminaison et correction	78
4.4	Étude de la complétude	80
4.4.1	Deux définitions possibles pour la notion de complétude	80
4.4.2	Une nouvelle transformation $\mathcal{T}_{AF2AL}^{lf}$	80
4.4.3	Discussion sur les autres sémantiques	83
4.5	Application à l'exemple médical	84
4.6	Conclusion	85
5	Cadre bipolaire pondéré : sémantique agrégative et explications	87
5.1	Introduction	88
5.2	Rappels préliminaires	89
5.2.1	Définition d'un QBAF et exemple illustratif	89
5.2.2	Acceptabilité et sémantique graduelle	90
5.2.3	Axiomes classiques des sémantiques graduelles	92
5.3	Sémantique agrégative graduelle	96
5.3.1	Fonction d'agrégation et propriétés usuelles	96

5.3.2	Motivation et définition proposée	96
5.3.3	Liens et différences avec les sémantiques modulaires	98
5.4	Propriétés souhaitables des différentes fonctions d'agrégation	99
5.4.1	Les cas de $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$	100
5.4.2	Le cas de φ_f	103
5.5	Comparaison formelle avec les axiomes classiques des QBAF	105
5.5.1	Lien entre axiomes argumentatifs et propriétés des opérateurs d'agrégation	105
5.5.2	Lien entre axiomes argumentatifs et interactions des opérateurs d'agrégation	107
5.5.3	Discussion sur les sémantiques graduelles classiques	108
5.6	QBAF et modèles structurels causaux	110
5.6.1	Passage des QBAF aux modèles structurels causaux (MSC)	111
5.6.2	Que cherche-t-on réellement à expliquer?	113
5.7	Conclusion	115
6	Conclusions et perspectives	117
6.1	Conclusions	117
6.2	Perspectives	119
6.2.1	Enrichissement des transformations	120
6.2.2	Évaluation, comparaison et enrichissement des explications cau- sales pour l'argumentation abstraite	121
6.2.3	Mise en œuvre expérimentale	122
	Références	125
	Annexes	133
A	Preuves du chapitre 2	133
B	Preuves du chapitre 3	136
C	Preuves du chapitre 4	139
D	Annexes du chapitre 5	143
D.1	Preuves du chapitre 5	143
D.2	Tableau récapitulatif des axiomes de l'argumentation	147

Résumé : Cette thèse s'inscrit dans le domaine de l'IA explicable (XAI) pour des systèmes d'argumentation abstraite (Abstract Argumentation Framework, AAF). Il s'agit d'un formalisme permettant de représenter et de raisonner sur des informations contradictoires par l'intermédiaire d'éléments abstraits, des arguments, et d'une relation binaire, appelée attaque, représentant une contradiction ou bien un conflit asymétrique entre deux arguments. Un argument peut alors être acceptable ou inacceptable selon un ensemble de règles appelé une sémantique. Dans ce contexte, nous travaillons sur la génération d'explications du statut d'acceptabilité d'un ensemble d'arguments, adaptées à son destinataire. Pour cela, des travaux en sciences sociales ont montré qu'explication et causalité sont deux notions fortement reliées bien que différentes. De fait, dans cette thèse, nous proposons plusieurs méthodes permettant de générer des explications, dites causales, c'est-à-dire déduites d'un raisonnement causal complexe, pour l'argumentation. Nous considérons trois cas : le cadre classique des systèmes d'argumentation abstraite, un cadre dynamique, c'est-à-dire qui tient compte de l'ordre d'énonciation des arguments, et un cadre bipolaire pondéré, incluant une deuxième relation binaire appelée support ainsi qu'une pondération sur les arguments.

Nous établissons tout d'abord une équivalence entre les AAF acycliques et un type spécifique de modèles structurels causaux. Nous proposons une transformation permettant de passer de l'un à l'autre et réciproquement. Cette transformation permet d'une part de comparer une définition existante de la notion d'explication en argumentation avec celle proposée dans le cadre des modèles structurels causaux. D'autre part, en réécrivant un AAF dans ce formalisme causal, nous pouvons générer des explications causales prenant en compte les connaissances du destinataire de l'explication.

Cependant, ces deux modèles ne prennent pas en compte l'ordre d'énonciation des arguments, propriété importante au cours d'un dialogue mais également pour le raisonnement causal. Pour cette raison, notre deuxième contribution consiste à enrichir un AAF, acyclique d'abord puis en levant cette contrainte, dans un formalisme logique dynamique, appelé langage d'action, en incluant l'ordre d'énonciation des arguments dans le modèle. Ce formalisme permet ainsi de modéliser l'évolution du dialogue à partir de laquelle il est possible d'extraire l'ensemble des causes de l'acceptabilité des différents arguments. Nous discutons ensuite de la façon d'utiliser ces chaînes causales pour la génération d'explications adaptées au destinataire.

Enfin, dans le cadre d'un système d'argumentation bipolaire pondéré (QBAF), le statut d'acceptabilité d'un argument n'est plus défini de façon binaire mais par une valeur numérique, souvent entre 0 et 1. Le choix de la fonction permettant de calculer cette valeur vient remplacer les sémantiques binaires par ce qu'on appelle des sémantiques graduelles. Nous proposons une telle famille de fonctions que nous appelons des sémantiques agrégatives graduelles permettant de décomposer le calcul de l'acceptabilité d'un argument en trois étapes d'agrégation : le calcul du poids global des attaquants, celui du poids global des supporteurs et enfin l'agrégation de ces deux poids avec la force intrinsèque de l'argument. Cela permet de conserver le caractère bipolaire tout au long du calcul, mais également d'ajouter de nouvelles propriétés pour caractériser les sémantiques graduelles permettant de guider le choix de cette fonction. Enfin, nous étendons la transformation d'un AAF en modèle structurel causal à ce cadre bipolaire pondéré permettant ainsi, pour des QBAF acycliques, de générer des explications causales.

Mots-clés : IA explicable, Argumentation abstraite, Causalité, Ordre d'énonciation, Langage d'action, Argumentation bipolaire pondérée

Causal explanations in abstract argumentation : classical, temporal and quantitative bipolar cases

Abstract : This thesis is part of the field of Explainable AI (XAI) for Abstract Argumentation Frameworks (AAF). The latter is a formalism suitable for representing and reasoning about contradictory pieces of information by means of abstract elements, called arguments, and a binary relation, called attack, which represents a contradiction or an asymmetric conflict between two arguments. An argument can then be either acceptable or unacceptable according to a set of rules called semantics. In this context, we work on generating explanations of the acceptability status of a set of arguments that are adapted to the explainee. To this end, work in the social sciences has shown that explanation and causality are two closely related, albeit distinct, notions. In this thesis, we propose several methods for generating so-called causal explanations, i.e. based on complex causal reasoning, for argumentation. We consider three cases : the classical abstract argumentation framework, a dynamic framework, that takes into account the order in which arguments are enunciated, and finally a quantitative bipolar argumentation framework, i.e. an AAF that includes a second binary relation called support as well as a weighting of the arguments.

We first establish an equivalence between acyclic AAF and a specific type of structural causal models. We propose a transformation that allows us to move from one to the other and vice versa. This makes it possible to compare an existing definition of the notion of explanations in argumentation with that proposed in the framework of structural causal models. Secondly, by rewriting an AAF in this causal formalism, we can generate causal explanations that take into account the knowledge of the explainee.

However, these two models do not take into account the order in which arguments are enunciated, an important property during a dialogue, but also for causal reasoning. For this reason, our second contribution consists in enriching an AAF, first acyclic and then by relaxing this constraint, in a dynamic logic formalism, called Action Language,

by including the order of enunciation of arguments in the model. This formalism makes it possible to model the evolution of the dialogue, from which it is possible to extract the set of causes of the acceptability of the different arguments. We then discuss how these causal chains can be used to generate explanations adapted to the explaineed.

Finally, in a Quantitative Bipolar Argumentation Framework (QBAF), the acceptability status of an argument is no longer a binary notion. It is defined by a numerical value, often between 0 and 1. The choice of the function used to compute this value replaces the binary semantics with so called gradual semantics. We propose a family of such functions, which we call gradual aggregative semantics, which breaks down the calculation of the acceptability of an argument into three aggregation stages : the calculation of the global weight of the attackers, the calculation of the global weight of the supporters, and finally the aggregation of these two weights with the intrinsic strength of the argument. This makes it possible to maintain the bipolar aspect throughout the calculation, but also to add new properties to characterise this semantics, guiding the choice for this semantics. Finally, we adapt the transformation of an AAF into a structural causal model to this extended framework, making it possible to generate causal explanations for acyclic QBAFs.

Keywords : Explainable AI, Abstract argumentation, Causality, Enunciation order, Action Language, Quantitative Bipolar Argumentation

Contexte

Durant chacune de nos interactions au quotidien, qu'il s'agisse d'un appel professionnel, d'un débat animé en famille ou encore d'une partie de jeu de société entre amis, nous échangeons des idées, des gestes, des croyances personnelles. Chacun de ces éléments sert à enrichir cet échange et apporte un nouveau sujet ou vient contredire ou soutenir un élément introduit précédemment. Dans un jeu à rôle secret par exemple, dans lequel un traître se cache parmi les joueurs, chaque parole prononcée ou sourire en coin pourra être un argument en faveur de l'innocence ou de la culpabilité d'un participant. Ces différents arguments peuvent également servir à en attaquer d'autres pouvant ainsi inverser l'issue de la discussion ou l'amener dans une impasse. Ce genre d'interactions peut être représenté informatiquement grâce au formalisme des *systèmes d'argumentation abstraite*.

Introduit par Dung (1995), ce formalisme permet de modéliser une interaction, c'est-à-dire d'en créer une représentation abstraite, comme un échange d'informations appelées des *arguments*. Ces derniers sont une abstraction d'éléments qui peuvent être de nature variée : des énoncés en langage naturel, des croyances personnelles, du langage non verbal, des actions, etc. De plus, afin de modéliser les différentes accusations et critiques à l'encontre d'autres arguments, une relation binaire, appelée *relation d'attaque*, est ajoutée. À l'aide d'un tel modèle, il est ensuite possible de déterminer les différentes issues de l'interaction, c'est-à-dire un ensemble d'arguments que l'on peut, selon des règles préalablement choisies, accepter collectivement. Dans l'exemple du jeu à rôle secret, à la fin de la phase de discussion, durant laquelle de nombreux arguments sont échangés, arrive une phase de vote. Chaque joueur doit alors prendre une décision et choisir un autre joueur autour de la table qu'il juge coupable. Dans certains cas, cela est assez facile car les arguments pointent vers une unique conclusion. Il n'y a alors qu'un seul ensemble d'arguments acceptables collectivement et la décision est, en général, unanime. Heureusement, la majorité du temps ce n'est pas si simple (c'est ce qui, selon moi, rend d'ailleurs ce genre de jeux amusant). Certains arguments se contredisent mutuellement et il n'y a pas d'éléments supplémentaires pour trancher. Dans le formalisme de Dung (1995), on parle de systèmes d'argumentation *cycliques*. Il faut alors faire

un choix parmi les différentes issues, choix qu'il faudra, à la fin de la partie, évidemment expliquer à la personne injustement accusée.

Pour cela, une possibilité peut être d'exposer tous les arguments ayant causé ce choix. Cependant, la liste peut être longue et l'explication peut en devenir incompréhensible. Il peut également arriver que des connaissances supplémentaires aient impacté la décision finale, comme des habitudes de comportements de certains participants traduisant par expérience la culpabilité ou l'innocence. Sans partager ces connaissances supplémentaires, l'explication peut une fois de plus ne pas être compréhensible pour son destinataire, appelé *explainee*. Cette interaction entre celui qui explique et celui qui reçoit l'explication, bien que très fréquente, est une tâche toujours délicate. Une illustration frappante de cette difficulté est le métier d'enseignant, dont l'une des tâches principales est d'expliquer. Que ce soit en tant qu'étudiant ou en tant qu'enseignant, nous avons tous déjà fait l'expérience d'une explication qui ne parvient pas à transmettre ce qui est souhaité au destinataire. L'algorithme de descente de gradient peut par exemple être expliqué de plusieurs manières : une approche mathématique, le détail du pseudo-code, une analogie avec une randonnée ou encore une illustration graphique. Cet exemple illustre que les connaissances requises pour comprendre l'explication varient mais la forme de celle-ci change aussi. Ainsi une explication qui est bonne pour l'un peut être mauvaise pour l'autre. Par conséquent, trouver une bonne explication peut s'avérer difficile. Cela soulève en particulier la question de la définition ce qu'est une "bonne explication".

Cette question se pose également en Intelligence Artificielle (IA) (Miller, 2019) et a donné naissance à un nouveau domaine de recherche, domaine dans lequel se situe cette thèse à savoir celui de l'IA explicable ou *eXplainable Artificial Intelligence* (XAI) en anglais. En plus des difficultés rencontrées entre humains, les systèmes d'IA sont utilisés pour accomplir des tâches variées : prédictions, génération de contenus, aide à la décision ou encore interaction avec un humain. Pour chacune de ces tâches des explications peuvent être demandées voire être nécessaires, notamment pour que l'interaction entre l'utilisateur et le système d'IA se passe au mieux. Cela entraîne l'existence d'une diversité importante de choses à expliquer. Dans cette thèse, nous nous concentrons sur le cas d'explications pour un système d'argumentation abstraite. Plus précisément, nous cherchons, comme dans l'exemple du jeu à rôle secret précédent, à répondre à la question "pourquoi cet ensemble d'arguments est-il acceptable (respectivement inacceptable)?".

L'utilisation de l'adverbe interrogatif "pourquoi" indique une volonté de connaître les causes qui ont effectivement conduit à cet état final précis. La réponse fournie doit donc nécessairement être construite à partir de ces causes, sans forcément en faire la liste. Nous nous intéressons plus particulièrement à de telles explications pour l'argumentation, explications que nous appelons *explications causales* par opposition à d'autres méthodes de la littérature que nous présentons dans le chapitre suivant.

Une autre composante importante au cours de nos interactions quotidiennes et que nous avons jusqu'ici ignorée est la dimension temporelle des arguments que nous énonçons. En effet, une interaction est en réalité un échange ordonné dans le temps d'arguments abstraits que nous appelons par analogie un *dialogue*. Dans un tel échange, l'ordre d'énonciation peut évidemment avoir une influence majeure sur l'issue du dialogue. De plus, même dans le cas où l'issue ne dépend pas de l'ordre, modifier cette séquence peut tout de même modifier les causes ayant conduit à une certaine conséquence, et donc modifier les explications causales qui en découlent. Considérons par exemple une partie de cartes entre Alice, Bob et Ève. Alice joue une première carte puis Bob en joue une

plus forte avant que Ève n'en joue une encore plus forte, remportant ainsi le pli. Dans un modèle sans temporalité, une explication à la question "pourquoi Alice n'a-t-elle pas remporté le pli?" serait que Ève et Bob ont joué une carte de valeur supérieure. Toutefois, dans le cas où l'ordre des coups est modélisé, Bob ayant joué une meilleure carte que celle d'Alice, cette dernière ne pouvait pas remporter la levée, et ce peu importe la carte de Ève. Si maintenant on considère un scénario où Ève joue avant Bob, alors l'issue resterait la même. Pourtant, c'est Ève qui cette fois-ci aurait été la cause de la perte du pli pour Alice. En modélisant cette situation avec un graphe d'argumentation abstraite, nous ne retrouvons pas nécessairement ces explications causales avec les méthodes de la littérature. En effet, ce formalisme donne une représentation de la partie à un instant donné qui omet l'ordre des coups joués. Il est donc essentiel de se poser la question de comment intégrer cette temporalité lors de la recherche d'explications causales.

Enfin, dans un système d'argumentation abstraite classique, la seule relation qui existe pour lier deux arguments est la relation d'attaque modélisant un conflit potentiellement asymétrique entre arguments. Toutefois, n'utiliser que la relation d'attaque peut être restrictif pour représenter des situations complexes. En effet, au cours d'une enquête de la plus haute importance pour savoir qui a fini la dernière tablette de chocolat noir, la confirmation visuelle d'un alibi par une personne tierce viendra renforcer cet alibi sans pour autant attaquer un autre argument. De manière générale, il a été avancé qu'un certain nombre d'applications, notamment celles qui relèvent du raisonnement humain, nécessitent une relation supplémentaire qui est opposée à l'attaque (Cacioppo *et al.*, 1997; Benferhat *et al.*, 2002). De plus, selon les points de vue, une vidéo ou une photo confirmant l'alibi a plus d'impact sur la décision finale qu'un témoignage d'un potentiel complice de ce crime. De manière générale, les différents arguments n'ont pas nécessairement la même influence sur une décision selon par exemple la personne qui l'énonce, selon son contenu ou encore selon son alignement avec les valeurs ou les croyances d'une personne.

Afin de modéliser ces éléments supplémentaires, une extension des systèmes d'argumentation abstraite, appelée *systèmes d'argumentation bipolaire pondérée*, a été proposée (Baroni *et al.*, 2019). Dans ce formalisme, une deuxième relation est ajoutée à l'attaque : nommée *support*, elle permet d'établir un lien modélisant le soutien qu'un argument, appelé *supporter*, apporte à un autre. Chaque argument est de plus enrichi avec une *force intrinsèque* dont l'interprétation peut varier et qui permet de représenter cette différence d'influence sur une décision. Par conséquent, dans un système d'argumentation bipolaire pondérée, le statut binaire d'acceptabilité d'un argument est également enrichi en un degré d'acceptabilité numérique qui est déterminé à l'aide d'une fonction, appelée *sémantique graduelle*. Pour choisir cette fonction, des propriétés venant remplacer l'ensemble de règles des systèmes classiques d'argumentation abstraite ont été proposées (Amgoud & Ben-Naim, 2018). Elles permettent de caractériser le comportement souhaité de cette fonction. Par exemple, si un support est ajouté alors l'argument doit être plus acceptable, et inversement si un attaquant est ajouté. Malgré ces règles, la forme, plus particulièrement le mécanisme de calcul, que peut prendre la sémantique graduelle reste très variée, autorisant ainsi des fonctions peu compréhensibles pour un humain. Cela peut être problématique pour la génération d'explications causales dans ce nouveau cadre. En effet, même si l'explication causale fournie pour expliquer une valeur d'acceptabilité est "bonne", si le mécanisme expliqué n'est lui-même pas compréhensible alors l'explication pourrait finalement ne plus être satisfaisante.

Questions de recherche et contributions

Dans cette thèse nous examinons les trois questions suivantes :

- Q1** – Comment générer des explications dites causales, c’est-à-dire intégrant un raisonnement causal complexe dans le processus, adaptées à leur destinataire pour un système d’argumentation abstraite?
- Q2** – Comment prendre en compte la temporalité en argumentation, c’est-à-dire l’ordre d’énonciation des arguments, dans un formalisme permettant un raisonnement causal complexe adapté à la génération d’explications?
- Q3** – Comment générer des explications causales dans un cadre bipolaire pondérée? En particulier comment choisir une sémantique graduelle adaptée à cette tâche dans un contexte donné?

Notre première contribution concerne la question **Q1** et propose une équivalence entre des systèmes d’argumentation abstraite acycliques, c’est-à-dire dont les arguments n’attaquent pas de façon directe ou indirecte leurs attaquants, et les modèles structurels causaux (Halpern, 2016). Ces derniers constituent un formalisme qui propose des définitions formelles des notions de causalité effective, causalité suffisante et d’explication. Celles-ci permettent de résoudre les situations dans lesquelles des définitions plus simples de causalité se trompent, comme la *surdétermination*, comme nous l’illustrons dans le chapitre 1. Un autre avantage de ce formalisme est que le destinataire de l’explication est inclus dans ces définitions par l’intermédiaire d’un ensemble d’éléments, appelés des *contextes*, qui modélisent sa connaissance. Afin d’exploiter ces propriétés des modèles structurels causaux dans le cadre des systèmes d’argumentation abstraite, nous proposons une transformation permettant de passer des systèmes d’argumentation abstraite à un cas particulier de modèles structurels causaux et réciproquement et nous prouvons une équivalence entre ces deux formalismes. De plus, grâce à la transformation permettant de passer dans le formalisme des modèles structurels causaux, il est possible de générer des explications causales adaptées aux connaissances du destinataire concernant l’acceptabilité des arguments dans le cadre des systèmes d’argumentation abstraite.

La contribution suivante, réalisée en collaboration avec Camilo Sarmiento¹, concerne l’ajout de la temporalité en argumentation abstraite (**Q2**). Nous proposons d’utiliser un langage d’action (Gelfond & Lifschitz, 1998) pour réécrire un graphe d’argumentation pour lequel on connaît l’ordre d’énonciation des arguments. Ce formalisme permet de modéliser l’évolution d’un dialogue de façon discrète par l’intermédiaire d’états et d’événements : on passe d’un état au suivant sous l’action d’événements qui sont réalisés, par exemple énoncer un nouvel argument, ou bien qui sont déclenchés automatiquement comme conséquence d’événements passés. Ainsi, contrairement à un système d’argumentation abstraite, nous obtenons, grâce à notre proposition, un historique des états passés du monde et des événements qui s’y sont produits. De plus, nous étudions les propriétés formelles de cette réécriture, d’abord dans le cas des graphes acycliques puis dans le cas général. En particulier, nous prouvons que la transformation que nous proposons est correcte, c’est-à-dire qu’elle respecte les règles de l’argumentation, qu’elle

1. doctorant au LIP6 au moment de notre collaboration

se termine, c'est-à-dire qu'il existe bien un état final pour le dialogue, et enfin qu'elle est complète, c'est-à-dire qu'aucune information n'est perdue en passant de l'argumentation abstraite au langage d'action. Enfin, ce dernier permet d'effectuer des raisonnements causaux complexes (Sarmiento *et al.*, 2022), nous permettant de discuter de la génération d'explications causales dans ce cadre.

Enfin, concernant l'extension au système d'argumentation bipolaire pondérée (**Q3**), nous proposons tout d'abord une instanciation des sémantiques graduelles que nous appelons des *sémantiques graduelles agrégatives*. Elles décomposent le calcul du degré d'acceptabilité d'un argument en trois étapes : le calcul de la force globale de ses attaquants, celui de la force globale de ses supporters, et enfin la combinaison de ces deux valeurs avec sa force intrinsèque. Cette décomposition crée un mécanisme modulable permettant de spécifier le comportement attendu pour chacune des fonctions et d'avoir un traitement asymétrique des attaques et des supports. Pour cela, nous discutons des propriétés que ces trois fonctions peuvent ou doivent vérifier selon le contexte d'application et leur lien avec les propriétés existantes de la littérature. Enfin, nous sommes revenus sur la première contribution, l'équivalence entre systèmes d'argumentation abstraite et modèles structurels causaux, pour l'étendre au cas des systèmes d'argumentation bipolaire pondérée acycliques.

Organisation du document

Le chapitre 1 présente formellement les contextes et principaux outils dans lesquels se placent les contributions de notre thèse : il rappelle d'abord les définitions théoriques des systèmes d'argumentation abstraite de Dung (1995). Une deuxième partie présente le domaine de l'IA explicable ainsi que les méthodes associées en argumentation abstraite. Dans un troisième temps, il décrit brièvement le contexte lié à la notion de causalité effective et notamment deux approches utilisées dans la suite. Enfin, il décrit les différents travaux de la littérature qui prennent en compte une forme de causalité et de temporalité en argumentation. Les chapitres 2, 3, 4 et 5 présentent ensuite les contributions de cette thèse. Le chapitre 2 présente l'équivalence entre les systèmes d'argumentation abstraite acycliques et un cas particulier de modèles structurels causaux que nous appelons des *graphes causaux argumentatifs*, répondant à la question **Q1**. Ensuite, dans les chapitres 3 et 4, nous répondons à la question **Q2** avec notre deuxième contribution : dans le chapitre 3, après des rappels sur le langage d'action que nous utilisons, nous proposons une réécriture des systèmes d'argumentation abstraite acycliques dans ce langage d'action ainsi qu'une étude de ses propriétés formelles. Nous discutons également de la génération d'explications causales adaptées à l'*explainée* avec cette transformation. Dans le chapitre suivant, nous levons la contrainte d'acyclicité en énonçant deux nouvelles réécritures dans ce formalisme ainsi que les propriétés formelles qu'elles vérifient. Le chapitre 5 présente la contribution concernant la définition et l'étude des sémantiques graduelles agrégatives ainsi qu'une extension du chapitre 2 au cas des systèmes d'argumentation bipolaire pondérée acycliques. Il répond ainsi à la question **Q3**. Enfin, le chapitre 6 conclut la thèse et présente les perspectives qu'elle ouvre.

Publications

Journal international avec comité de relecture

Munro, Y., Sarmiento, C., Bloch, I., Bourgne, G., Lesot, M.- J. Integrating Temporality and Causality into Acyclic Argumentation Frameworks using a Transition System. *AI Communications*, in press

Conférence internationale avec comité de relecture

Munro, Y., Sarmiento, C., Bloch, I., Bourgne, G., Pelachaud, C., Lesot, M.- J. (2024). An action language-based formalisation of an abstract argumentation framework. *Dans Proc. of the 25th International Conference on Principles and Practice of Multi-Agent Systems (PRIMA 2024)*

Workshop international avec comité de relecture

Munro, Y., Sarmiento, C., Bloch, I., Bourgne, G., Lesot, M.- J. (2023). Dynamic Argumentation and Action Languages : Towards Explanations. *Dans the 4th International Workshop on Explainable Logic-Based Knowledge Representation XLoKR@KR2023*

Munro, Y., Bloch, I., Chetouani, M., Lesot, M.- J., Pelachaud, C. (2022). Argumentation and Causal Models in Human-Machine Interaction : A Round Trip. *Dans 8th International Workshop on Artificial Intelligence and Cognition (AIC)*

Conférence nationale avec comité de relecture

Munro, Y., Bloch, I., Chetouani, M., Pelachaud, C., Lesot, M.- J. (2024). Sémantique agrégative graduelle pour les systèmes d'argumentation bipolaires pondérés. *Dans Journées d'Intelligence Artificielle Fondamentale (JIAF-JFPDA 2024)*

Munro, Y., Sarmiento, C., Bloch, I., Bourgne, G., Lesot, M.- J. (2023). Temporalité et causalité en argumentation abstraite. *Dans Journées d'Intelligence Artificielle Fondamentale (JIAF-JFPDA 2023)*

Munro, Y., Bloch, I., Chetouani, M., Lesot, M.- J., Pelachaud, C. (2022). De l'équivalence entre les modèles structurels causaux et les systèmes abstraits d'argumentation. *Dans Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA 2022)*.

Pré-print

Munro, Y., Sarmiento, C., Bloch, I., Bourgne, G., Lesot, M. - J. (2023). Integrating Temporality and Causality into Acyclic Argumentation Frameworks using a Transition System. *arXiv preprint arXiv :2303.09197*

Sommaire

1.1 Système d'argumentation abstraite	11
1.1.1 Définitions générales	12
1.1.2 Sémantiques à base d'extension	14
1.1.3 Sémantiques construites par étiquetage	16
1.2 Causalité en argumentation abstraite	18
1.2.1 Discussion sur la définition de causalité effective	19
1.2.2 Causalité effective en argumentation abstraite	21
1.3 Expliquer le statut d'acceptabilité d'un argument	22
1.3.1 Explications en intelligence artificielle : le domaine de l'IA explicable	22
1.3.2 ArgXAI, <i>argumentative</i> XAI	23
1.4 Temporalité en argumentation abstraite	25
1.5 Bilan	27

Dans ce chapitre nous introduisons formellement et discutons des travaux existants sur les contextes et principaux outils dans lesquels se placent les contributions de cette thèse, comme présenté dans l'introduction : les systèmes d'argumentation abstraite (section 1.1), deux approches différentes de la causalité effective (section 1.2), la notion d'explications (section 1.3) et enfin la temporalité en argumentation (section 1.4). Les systèmes d'argumentation bipolaire pondérée n'étant traités que dans le chapitre 5, nous les introduisons formellement dans ce chapitre.

1.1 Système d'argumentation abstraite

Dans cette section, nous rappelons quelques éléments clés des systèmes d'argumentation abstraite ou *Abstract Argumentation Framework* (AAF) (Dung, 1995) et justifions,

par là même, le choix d’utiliser ce formalisme comme modèle d’une interaction. Un premier avantage des AAF est leur proximité avec la cognition humaine. En effet, il a été établi que les raisonnements humains (Mercier & Sperber, 2011) ainsi que la majorité des énoncés dans les explications (Antaki & Leudar, 1992) sont argumentatifs. L’utilisation des systèmes d’argumentation abstraite constituent donc un cadre propice pour produire des explications qui sont adaptées à l’humain.

1.1.1 Définitions générales

Un AAF représente le monde à l’aide d’éléments abstraits, des *arguments*, et d’une relation binaire, l’*attaque*, qui indique un conflit pouvant être asymétrique entre des arguments. Depuis son introduction par Dung (1995), ce formalisme a largement été repris et étendu à des cadres plus larges : bipolaire (Cayrol & Lagasquie-Schiex, 2005b), incomplet (Coste-Marquis *et al.*, 2007; Mailly, 2022), pondéré (Dunne *et al.*, 2011), probabiliste (Li *et al.*, 2011), flou (Janssen *et al.*, 2008), basé sur des préférences (Amgoud & Cayrol, 2002), etc. De même, de nombreuses applications ont été proposées : pour des systèmes de prise de décision (Amgoud *et al.*, 2008), pour modéliser des stratégies de persuasion (Bonzon *et al.*, 2014), pour l’approximation de réseaux de neurones (Potyka, 2021), pour des systèmes de dialogue (Black *et al.*, 2021), pour modéliser la propagation d’une information (de Tarle, 2023), pour des systèmes de recommandations (Rago *et al.*, 2018) ou encore comme outils d’aide à la détection de fraude en ligne pour la police néerlandaise (Bex *et al.*, 2016).

Formellement, un AAF est défini de la manière suivante.

Définition 1.1 (Système d’argumentation abstraite). *Un système d’argumentation abstraite est un couple $(\mathcal{A}, \mathcal{R})$ tel que :*

- *$\mathcal{A}$ est un ensemble fini d’arguments, c’est-à-dire un ensemble d’objets abstraits désignés par un identifiant ;*
- *$\mathcal{R}$ est une relation binaire sur $\mathcal{A} \times \mathcal{A}$. On appelle $\mathcal{R}$ la relation d’attaque et on dit qu’un argument $a \in \mathcal{A}$ attaque $b \in \mathcal{A}$ si $(a, b) \in \mathcal{R}$, aussi écrit $\mathcal{R}(a, b)$.*

On introduit de plus la notation suivante : étant donné $AF = (\mathcal{A}, \mathcal{R})$ un AAF et $a \in \mathcal{A}$ un argument, $Att_a^{\mathcal{R}}$ représente l’ensemble des attaquants directs de a pour la relation $\mathcal{R}$. Formellement : $Att_a^{\mathcal{R}} = \{b \in \mathcal{A} \mid (b, a) \in \mathcal{R}\}$. Quand une seule relation d’attaque est définie, on note simplement Att_a . De plus, on dit qu’un argument c est un *défenseur* direct de l’argument a si et seulement si c attaque un attaquant direct de l’argument a .

Remarque – Ce formalisme n’impose rien quant à la structure interne d’un argument, ni sur la nature d’une attaque. Ainsi, un argument peut aussi bien être un énoncé en langage naturel que du langage non verbal ou encore une action. Cette grande souplesse constitue un avantage de plus justifiant le choix d’utiliser ce formalisme.

Définition 1.2 (Graphe d’argumentation abstraite). *Soit $AF = (\mathcal{A}, \mathcal{R})$, un AAF. Comme $\mathcal{R}$ est une relation binaire à support fini, on peut représenter un système d’argumentation abstraite sous la forme d’un graphe où $\mathcal{A}$ donne l’ensemble des nœuds du graphe et $\mathcal{R}$ l’ensemble des arêtes.*

Pour illustrer ces notions, nous introduisons ici un exemple fictif mais réaliste issu d’une discussion entre un médecin demandeur, D, et un radiologue, R, à propos d’un

examen d’un jeune bébé atteint de la pathologie X. Cet exemple sert de fil rouge au cours des chapitres ultérieurs de la thèse.

Exemple 1 (Examen médical). *Le dialogue entre le médecin et le radiologue est retranscrit ici :*

D : Peux-tu me faire un scanner pour ce bébé ? (a)

R : Il vaut mieux éviter les radiations ionisantes pour les jeunes bébés. (b)

R : Je peux te proposer une IRM dans deux jours. (c)

D : On peut voir X sur une IRM ? (d)

R : Oui bien sûr ! Si tu veux une confirmation, regarde le guide des bonnes pratiques en radiologie. (e)

D : Mais puisqu’il s’agit d’un bébé, il risque de bouger et donc on pourrait manquer l’information que l’on cherche car l’image ne sera pas très nette. (f)

R : Ne t’inquiète pas, j’ai l’habitude de faire ce genre d’examen pour des bébés. (g)

D : Est-ce que cela ne coûte pas beaucoup plus cher à l’hôpital de faire une IRM ? (h) Il faut aussi que je voie avec la famille du patient car ça pourrait leur revenir plus cher (i).

R : Aucun problème dans ces cas là. Ce coût élevé englobe l’expérience acquise par mon équipe, de sorte qu’à l’avenir, elle puisse réaliser ce type d’examen délicat sans moi. (j)

D : Je viens de discuter avec la famille, aucun problème avec l’IRM elle est couverte pour ça. (k)

D : Cependant, la famille n’est pas rassurée de devoir attendre deux jours, peux-tu faire l’IRM dans la journée ? (l)

R : Non je n’ai vraiment plus de place. Mon prochain créneau est dans deux jours comme je te l’ai dit. (m)

À la suite de cet échange, la décision est donc arrêtée sur une IRM programmée dans deux jours. Mais plus tard dans la journée, le médecin reçoit un appel de la famille pour prévenir que le bébé ne va vraiment pas bien et insister sur l’urgence de l’examen. Le médecin recontacte donc le radiologue pour ajouter un dernier argument :

D : C’est vraiment urgent pour le bébé, il faut une place aujourd’hui. (n)

À partir de ce dialogue, on peut extraire manuellement des arguments et les relations entre eux afin d’obtenir l’AAF représenté en figure 1.1 avec les arguments suivants : $\{a$: Scanner, b : Radiations ionisantes, c : IRM dans deux jours, d : X non visible par IRM, e : X visible par IRM, f : Conditions difficiles, g : Expérience, h : Coût élevé pour l’hôpital, i : Coût élevé pour le patient, j : Pas problématique pour l’hôpital, k : Famille couverte pour une IRM, l : IRM aujourd’hui, m : Pas de disponibilité aujourd’hui, n : C’est une urgence $\}$. Les arguments a , c , l sont appelés variables de décision, leur acceptation étant le critère déclenchant une décision : scanner, IRM dans deux jours, ou IRM aujourd’hui. Même si ces trois arguments ont un rôle particulier par rapport aux autres, ce sont des entités abstraites comme tous les autres arguments du graphe. Pour cette raison, nous n’ajoutons aucune restriction formelle sur eux, contrairement à ce qui peut être fait dans des contextes tels que la négociation basée sur les arguments (Amgoud *et al.*, 2007).

Le système d’argumentation obtenu, illustré sur la figure 1.1, est un graphe possible qui peut être extrait de ce dialogue. Ici, le processus d’extraction a été effectué manuellement pour créer cet AAF. Cependant, ce processus peut être réalisé automatiquement en utilisant des méthodes dites de fouille d’arguments ou *argument mining*, voir par exemple Lippi & Torroni (2016).

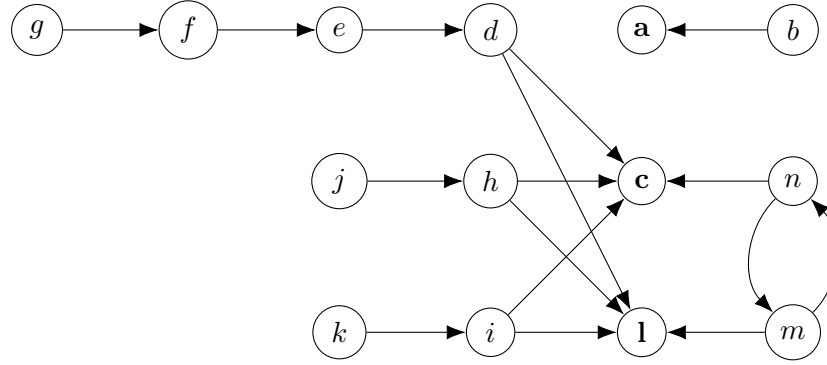


FIGURE 1.1 – Graphe d’argumentation associé à l’exemple 1.

L’AAF correspondant à cet exemple illustre le cas des AAF *cycliques* car l’argument m attaque son attaquant n . De façon générale, un AAF est cyclique s’il contient des arguments qui attaquent, de façon directe ou indirecte, un de leurs attaquants.

Notion d’acceptabilité – L’objectif des systèmes d’argumentation abstraite est, étant donné un AAF, d’évaluer les différents arguments. Plus précisément, la question est de savoir quels sont les arguments qui sont *acceptables*, c’est-à-dire ceux qu’aucune information également acceptable ne contredit. En particulier, dans l’exemple précédent, nous souhaitons connaître quelles sont les décisions possibles pour l’examen parmi les trois envisagées, ce qui est équivalent d’un point de vue argumentatif à savoir quels arguments parmi a, c, l sont acceptables.

Dans le cas des graphes cycliques, comme c’est le cas de l’exemple 1, il s’agit d’une question complexe pour laquelle plusieurs définitions pouvant conduire à des résultats différents existent (Baroni *et al.*, 2011). Dans les sections 1.1.2 et 1.1.3 ci-dessous, nous présentons les deux familles d’approches principales en argumentation abstraite classique.

1.1.2 Sémantiques à base d’extension

Pour déterminer le statut d’acceptabilité des arguments, des règles de base ont été proposées (Dung, 1995). Tout d’abord, un argument est *acceptable* si et seulement si il n’est pas attaqué ou que tous ses attaquants sont *inacceptables*. Ainsi, un argument attaqué par au moins un argument acceptable est *inacceptable*. Cependant ces règles de base ne sont pas suffisantes pour résoudre tous les conflits, en particulier en présence de cycles dans le graphe d’argumentation. De fait, des règles supplémentaires ont été établies afin de déterminer des ensembles d’arguments collectivement acceptables, appelés des *extensions*. Ainsi, un objectif de l’argumentation abstraite est de déterminer l’ensemble des extensions, c’est-à-dire l’ensemble des ensembles d’arguments qui vérifient certaines règles choisies. Ces règles s’appuient en particulier sur les définitions suivantes :

- Un ensemble d’arguments S est *sans conflit* s’il ne contient pas de couple d’arguments tels que l’un attaque l’autre :

$$\forall (a, b) \in S^2, (a, b) \notin \mathcal{R}$$

- Un argument $a \in \mathcal{A}$ est *acceptable* par un ensemble S si S attaque tous les attaquants de a :

$$\forall b \in \text{Att}_a, \exists c \in S \cap \text{Att}_b$$

- Un ensemble d’arguments S sans conflit et tel que tous ses éléments sont acceptables par S est dit *admissible* :

$$\forall (a, b) \in S^2, (a, b) \notin \mathcal{R} \text{ et } \forall a \in S, \forall b \in \text{Att}_a, \exists c \in S \cap \text{Att}_b$$

- Un ensemble d’arguments S est dit *admissible lié* s’il est admissible et si au moins un de ses arguments est attaqué :

$$S \text{ est admissible et } \exists x \in S \text{ tel que } \text{Att}_x \neq \emptyset.$$

Un tel x est alors appelé un *sujet* de S .

À partir de ces définitions, différentes *sémantiques* ont été proposées pour déterminer les ensembles d’arguments acceptables (Baroni *et al.*, 2011). Dans cette thèse, nous nous concentrons sur celles basées sur la notion d’admissibilité de Dung (1995) :

- Sémantique *complète* Σ_c : une extension S est complète si et seulement si S est admissible, et $\forall a \in \mathcal{A}$, si a est défendu par S alors $a \in S$. On note Ext_c l’ensemble des extensions complètes.
- Sémantique *préférée* Σ_p : une extension S est préférée si et seulement si S est une extension complète maximale au sens de l’inclusion. On note Ext_p l’ensemble des extensions préférées.
- Sémantique *de base* Σ_b : une extension S est de base si et seulement si S est l’extension complète minimale au sens de l’inclusion. Cette dernière existe et est unique pour tout AAF. On note Ext_b l’ensemble des extensions de base.
- Sémantique *stable* Σ_s : une extension S est stable si et seulement si S est sans conflit et $\forall a \in \mathcal{A} \setminus S, \exists b \in S \cap \text{Att}_a$. On note Ext_s l’ensemble des extensions stables.

Dans le cas d’un graphe acyclique, toutes ces définitions conduisent à une même extension : il existe un unique état final, c’est-à-dire que le statut d’acceptabilité de chacun des arguments du graphe est unique. Pour l’obtenir, il suffit de partir des arguments non attaqués, qui sont acceptables par défaut, puis de parcourir le graphe en appliquant uniquement les règles de base de l’argumentation.

Cela n’est plus le cas dès que le graphe possède un cycle. En effet, il n’y a plus unicité de l’ensemble des arguments acceptables entre les différentes sémantiques mais également pour une sémantique donnée. Pour illustrer ces difficultés, on introduit deux exemples jouets : un AAF avec un cycle de longueur paire et un AAF avec un cycle de longueur impaire.

Exemple 2 (Cycle de longueur paire). *On considère l’AAF suivant : $AF = (\mathcal{A}, \mathcal{R})$ avec $\mathcal{A} = \{a, b, c, d\}$ et $\mathcal{R} = \{(a, b), (b, a), (a, c), (b, c), (c, d)\}$. Son graphe associé est représenté à gauche de la figure 1.2.*

Comme a attaque b et que b attaque a , les règles de base de l’argumentation ne permettent pas de choisir : on peut accepter l’un et rejeter l’autre, ou le contraire, ou bien rejeter les deux.

Considérons le cas où a est considéré comme acceptable et posons $S_1 = \{a\}$. Alors a est défendu par S_1 car $\text{Att}_a = \{b\}$ et a attaque b . Donc S_1 attaque bien tous les attaquants de a . De plus, a ne s’attaque pas lui-même donc S_1 est sans conflit. On obtient un ensemble

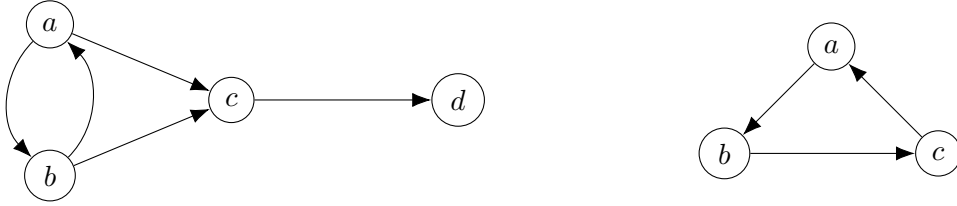


FIGURE 1.2 – Graphe d'argumentation associé à l'exemple 2 à gauche et à l'exemple 3 à droite

d'arguments sans conflit et dont tous les éléments sont acceptables par cet ensemble c'est-à-dire que S_1 est admissible. Toutefois, S_1 n'est pas une extension complète. En effet, S_1 défend également d car a attaque c . En ajoutant d à S_1 , c'est-à-dire $S_1 = \{a, d\}$, on obtient une extension complète. De la même façon en considérant b acceptable plutôt que a , on obtient que $S_2 = \{b, d\}$ est une extension complète. Enfin, $S_3 = \emptyset$ est également une extension complète car elle défend trivialement tous ses éléments et tous les éléments qu'elle défend sont dans l'ensemble. Notons que cela n'est pas le cas dès qu'il y a des arguments non attaqués.

En appliquant les définitions rappelées ci-dessus, on obtient les ensembles suivants :

- $Ext_c = \{\{a, d\}, \{b, d\}, \emptyset\}$
- $Ext_p = \{\{a, d\}, \{b, d\}\}$
- $Ext_b = \{\emptyset\}$
- $Ext_s = \{\{a, d\}, \{b, d\}\}$

Exemple 3 (Cycle de longueur impaire). Dans cet exemple, on s'intéresse au cas des cycles de longueur impaire. On considère pour cela l'AAF suivant : $AF = (\mathcal{A}, \mathcal{R})$ avec $\mathcal{A} = \{a, b, c\}$ et $\mathcal{R} = \{(a, b), (b, c), (c, a)\}$. Son graphe associé est représenté à droite de la figure 1.2.

À nouveau, les règles de base de l'argumentation ne suffisent pas à déterminer quels sont les arguments acceptables et inacceptables. Chaque argument jouant un rôle symétrique par rapport aux autres, essayons de construire une extension contenant l'argument a . Posons donc $S_1 = \{a\}$. Comme c attaque a et que b attaque c alors pour que a soit défendu par S_1 , il faut inclure b dans l'ensemble : $S_1 = \{a, b\}$. Cependant, a attaque b donc S_1 n'est pas sans conflit et ne peut donc être un ensemble admissible. En raisonnant de la même manière pour tous les arguments, on obtient les ensembles suivants :

- $Ext_c = \{\emptyset\}$
- $Ext_p = \{\emptyset\}$
- $Ext_b = \{\emptyset\}$
- $Ext_s = \emptyset$

On peut noter avec cet exemple qu'il n'existe pas toujours d'extension stable.

1.1.3 Sémantiques construites par étiquetage

Une autre façon de déterminer le statut d'acceptabilité des arguments d'un système d'argumentation abstraite est d'utiliser une fonction d'étiquetage (Caminada, 2006). Cette fonction associe à chaque argument du graphe une étiquette correspondant à son statut d'acceptabilité. Formellement :

Définition 1.3 (Fonction d'étiquetage). *Soit $AF = (\mathcal{A}, \mathcal{R})$ un AAF.*

$\mathcal{L}$ est une fonction d'étiquetage ou labelling sur AF si et seulement si $\mathcal{L}$ associe à chaque argument une étiquette dans l'ensemble $\{IN, OUT, UNDEC\}$ c'est-à-dire :

$$\mathcal{L} : \mathcal{A} \rightarrow \{IN, OUT, UNDEC\}$$

Remarque – Il est aussi possible de définir une fonction d'étiquetage comme un ensemble de couples de la forme (argument, étiquette).

Alors que les méthodes par extension n'autorisent que deux statuts (acceptable si l'argument est dans l'extension, inacceptable sinon), les fonctions d'étiquetage utilisent trois étiquettes correspondant à trois statuts d'acceptabilité :

- *IN* : un argument étiqueté *IN* est considéré comme acceptable ;
- *OUT* : un argument étiqueté *OUT* est considéré comme inacceptable ;
- *UNDEC* : un argument étiqueté *UNDEC* est dans un état indéfini, il n'est ni acceptable ni inacceptable.

À partir d'une fonction d'étiquetage on peut partitionner l'ensemble des arguments d'un graphe comme $\mathcal{A} = in(\mathcal{L}) \cup undec(\mathcal{L}) \cup out(\mathcal{L})$ avec :

- $in(\mathcal{L}) = \{a \in \mathcal{A} \mid \mathcal{L}(a) = IN\}$
- $out(\mathcal{L}) = \{a \in \mathcal{A} \mid \mathcal{L}(a) = OUT\}$
- $undec(\mathcal{L}) = \{a \in \mathcal{A} \mid \mathcal{L}(a) = UNDEC\}$

Sans contrainte supplémentaire sur cette fonction, n'importe quel étiquetage peut être créé et donc les statuts d'acceptabilité ainsi obtenus ne respecteraient pas nécessairement les règles de l'argumentation. Pour cette raison, on introduit un sous-ensemble de fonctions d'étiquetage appelées des *reinstatement labellings* (Caminada, 2006) ou fonctions d'étiquetage complet (nous en expliquons la raison avec le théorème 1.1) définies de la façon suivante :

Définition 1.4 (Étiquetage complet). *Soit $AF = (\mathcal{A}, \mathcal{R})$ un AAF et $\mathcal{L}$ une fonction d'étiquetage sur AF .*

$\mathcal{L}$ est un étiquetage complet sur AF si et seulement si :

- $\forall a \in \mathcal{A}, \mathcal{L}(a) = IN$ si et seulement si $\forall b \in Att_a, \mathcal{L}(b) = OUT$;
- $\forall a \in \mathcal{A}, \mathcal{L}(a) = OUT$ si et seulement si $\exists b \in Att_a, \mathcal{L}(b) = IN$.

Exemple 2 (Cycle de longueur paire). *(suite) – En appliquant les deux conditions précédentes au graphe d'argumentation présenté à gauche de la figure 1.2, on obtient les trois étiquetages complets suivants :*

- $\mathcal{L}_1 = \{(a, IN), (b, OUT), (c, OUT), (d, IN)\}$
- $\mathcal{L}_2 = \{(a, OUT), (b, IN), (c, OUT), (d, IN)\}$
- $\mathcal{L}_3 = \{(a, UNDEC), (b, UNDEC), (c, UNDEC), (d, UNDEC)\}$

Les deux conditions définissant un étiquetage complet sont en réalité des réécritures des règles de base de l'argumentation dans le formalisme des fonctions d'étiquetage. Prenons l'exemple de la première condition : celle-ci exprime qu'un argument est acceptable, c'est-à-dire *IN*, si et seulement si tous ses attaquants sont inacceptables, c'est-à-dire *OUT*. L'avantage de cette méthode est que les arguments qui ne vérifient aucun des deux critères sont déclarés comme *UNDEC*. De cette façon, on obtient une procédure permettant de retrouver l'ensemble des extensions complètes d'un système d'argumentation abstraite.

En effet, Caminada (2006) a démontré qu’il y a une équivalence entre les étiquetage complets et les extensions complètes. Pour cela on définit d’abord les transformations suivantes permettant de passer d’une fonction d’étiquetage à une extension et réciproquement :

- $LabToExt(\mathcal{L}) = in(\mathcal{L})$
- $ExtToLab(S) = \{(a, IN) \mid a \in S\} \cup \{(b, OUT) \mid \exists a \in S, a \in Att_b\} \cup \{(c, UNDEC) \mid c \notin S \text{ et } \forall a \in S, a \notin Att_c\}$

À partir de cette transformation, Caminada (2006) énonce le théorème suivant :

Théorème 1.1 (Équivalence entre étiquetage complet et extension complète). *Soit $AF = (\mathcal{A}, \mathcal{R})$ un AAF, S une extension complète de AF et $\mathcal{L}$ un étiquetage complet sur AF . Alors :*

- *$ExtToLab(S)$ est un étiquetage complet ;*
- *$LabToExt(\mathcal{L})$ est une extension complète de AF .*

En raison de cette équivalence, un *reinstatement labellings* peut aussi être appelé un étiquetage complet.

En ajoutant des contraintes supplémentaires à un étiquetage complet, il est possible d’obtenir une équivalence similaire pour chacune des sémantiques à base d’extension présentées dans la section précédente (Caminada, 2006). Dans la suite de la thèse, nous nous concentrons principalement sur la notion d’extension complète et seule la notion d’étiquetage complet sera utilisée. Pour cette raison, nous ne présentons pas ici les contraintes additionnelles qui sont des traductions dans le formalisme des étiquetages des sémantiques à base d’extension correspondantes.

1.2 Causalité en argumentation abstraite

Avant d’introduire la notion d’explications causales utilisée dans cette thèse, il faut introduire la notion de causalité, ce qui est l’objet de cette section, et la notion d’explication, introduite dans la section suivante. Dans cette section, nous décrivons en particulier la notion de cause effective. Ensuite, nous présentons les recherches existantes sur sa formalisation en argumentation abstraite ainsi que sur l’inclusion de la temporalité.

La définition de la notion de causalité est une question étudiée depuis l’Antiquité en philosophie (Aristote, 350 AEC) et plus récemment en droit (Wright, 1985), mathématiques (Pearl, 2000) et informatique (Batusov & Soutchanski, 2018) entre autres. S’agissant d’une notion très vaste, l’objectif de cette section n’est pas de faire un état de l’art complet sur la question mais plutôt de donner une idée générale de deux catégories d’approches existantes, reprises en informatique et dans cette thèse en particulier (Halpern, 2016; Wright, 2011). Nous ne donnons pas non plus les définitions formelles de ces deux approches. En effet, chacune nécessite d’introduire un formalisme supplémentaire, les modèles structurels causaux (Halpern & Pearl, 2005) et les langages d’action (Gelfond & Lifschitz, 1998). Par conséquent, nous les présentons en même temps que ces formalismes, dans les chapitres 2 et 3.

Tout d’abord, on distingue deux types de causalité : la causalité *générale* et la causalité *effective*. La première vise à déterminer des lois sur le fonctionnement du monde, comme par exemple la loi de la gravitation universelle selon laquelle deux corps s’attirent avec une force proportionnelle à leurs masses et inversement proportionnelle au carré de la distance qui les sépare. La causalité effective, contrairement à la causalité

générale, ne répond pas à un tel objectif global, mais vise à déterminer les faits particuliers qui se sont produits étant donnée une situation spécifique et qui ont provoqué la conséquence étudiée. C'est ce type de causalité qui nous intéresse dans la suite de la thèse.

Pour illustrer la différence entre causalité effective et causalité globale, considérons un scénario très simple dans lequel un jardinier, en train de tailler un pommier, donne un coup à une pomme de l'arbre qui tombe au sol. En règle générale, une pomme tombe d'un arbre à cause de la force de gravitation exercée par la Terre sur la pomme. Cependant, dans la situation présente, la pomme est tombée de l'arbre car le jardinier l'a touchée pendant qu'il taillait le pommier. La gravitation est donc une cause générale de la chute de la pomme, là où c'est le jardinier qui en est la cause effective dans cette situation spécifique.

1.2.1 Discussion sur la définition de causalité effective

Au sein de la recherche sur la notion de causalité effective, on distingue principalement deux familles d'approche : les approches *contrefactuelles* et les approches *par régularité*.

Approches contrefactuelles – Elles définissent une cause effective comme un événement qui fait la différence entre ce qui s'est réellement passé et ce qui se serait passé sans lui. En particulier, elles examinent si la conséquence se produit même si la cause effective candidate ne se produit pas : si c'est le cas, la candidate est rejetée. Ce raisonnement définit ce que l'on appelle un test *but-for* et constitue une forme simple de raisonnement contrefactuel.

Toutefois, ce test renvoie un résultat différent d'une réponse intuitive quand plusieurs éléments ou ensembles d'éléments suffisent individuellement à produire la conséquence recherchée. Cette situation est appelée la *surdétermination*. Pour Beckers (2021), ce test échoue car il présuppose d'une équivalence entre une dépendance causale et une cause effective. Pour illustrer l'échec de ce test, nous reprenons un exemple classique tiré de la littérature en causalité (Lewis, 2004).

Exemple 4 (Lancer de pierre). *Suzy et Billy lancent tous les deux une pierre en direction d'une bouteille de verre. Ils sont tous les deux parfaitement précis et sont donc sûrs de toucher la bouteille s'ils lancent effectivement la pierre. Si l'une des pierres atteint la bouteille alors celle-ci se casse. Enfin, on fait l'hypothèse que dans le cas où les deux lancent une pierre, la pierre de Suzy touche la bouteille toujours la première.*

On se place dans la situation où les deux lancent la pierre. Celle de Suzy arrive donc en premier et casse la bouteille. Intuitivement, le lancer de Suzy est une cause de la bouteille cassée. Toutefois, ce n'est pas le résultat obtenu par la définition précédente de test but-for. En effet, celui-ci examine la situation où la cause présumée ne se produit pas, c'est-à-dire si Suzy ne lance pas. Or ici, la conséquence se produit quand même puisque le lancer de Billy casse la bouteille. D'après ce test, Suzy n'est donc pas une cause effective du fait que la bouteille se casse, c'est Suzy et Billy qui en sont une. Dans cet exemple, le fait que Suzy lance une pierre ainsi que le fait que Billy lance une pierre suffisent individuellement à casser le bouteille. Il s'agit bien d'un cas de surdétermination. On retrouve d'ailleurs,

comme énoncé avant l'exemple, que la dépendance causale de l'état de la bouteille, c'est-à-dire le fait que Suzy lance sa pierre et Billy lance sa pierre, est équivalente à la notion de cause effective.

Plusieurs autres définitions par approche contrefactuelle inspirées de ce test ont été proposées. Dans cette thèse, nous nous basons sur celle introduite par Halpern (2016), qui figure parmi les approches de causalité effective les plus utilisées aujourd'hui dans la littérature en informatique. Cette définition, que nous présentons formellement dans le chapitre 2, reprend l'idée du test *but-for* en introduisant un élément supplémentaire, l'interventionnisme. Il s'agit d'autoriser de fixer arbitrairement l'occurrence de certains événements dans une situation donnée même si ces événements n'auraient pas dû se produire. Avec cet ajout, la définition proposée par Halpern (2016), après plusieurs itérations (Halpern & Pearl (2005) et Halpern (2015)), se base sur le test suivant : si la cause effective présumée ne s'était pas produite et sachant que malgré cela certains événements (autres que la conséquence) se produisent quand-même (intervention), la conséquence se produit-elle quand même aussi ?

Exemple 4 (Lancer de pierre). *(suite) – Le problème du test but-for sur cet exemple est que le fait que la pierre de Suzy touche la bouteille empêche la pierre de Billy de la toucher. Cependant si Suzy ne lance plus sa pierre, alors celle de Billy va toucher la bouteille. Or dans la situation réelle, la pierre de Billy ne touche pas la bouteille. On considère donc l'évènement contrefactuel suivant : si Suzy n'avait pas lancé sa pierre et sachant que Billy n'a pas touché la bouteille, la bouteille serait-elle cassée ? Dans ce scénario, la bouteille reste intacte et donc le fait que Suzy ait lancé une pierre est une cause effective du fait que la bouteille soit cassée.*

Approches par régularité – Contrairement aux approches contrefactuelles, celles par régularité se rapprochent de la notion de causalité générale. En effet, elles se basent sur le fait que si une cause est systématiquement suivie par un effet, alors il est raisonnable d'inférer que cet effet est dû à la cause (Andreas & Guenther, 2021). Cependant, les approches par régularité se sont heurtées à de nombreux problèmes dans des situations simples comme par exemple le fait que le chant d'un coq précède toujours un lever de soleil et pourtant il n'en n'est pas la cause. De nombreux raffinements ont de fait été proposés comme l'utilisation de "lois de la nature" écrites sous la forme de disjonctions de conjonctions où chacune des conjonctions est un ensemble minimalement suffisant mais non nécessaire. De plus, chacun des littéraux de ces conjonctions est non suffisant mais nécessaire pour que la conjonction soit effectivement suffisante, c'est-à-dire que la conjonction soit vraie dans le contexte étudié. Pour certaines approches par régularité, ces littéraux constituent des causes effectives (Mackie, 1980). Cette dernière version répond à la majorité des difficultés rencontrées à l'exception d'une : deux événements qui sont les conséquences d'une cause commune seront causalement reliés.

Exemple 5 (Chant du coq et lever du Soleil). *Pour illustrer cette difficulté, considérons la situation suivante : deux événements se sont produits : le coq chante E_1 et le soleil se lève E_2 . Nous introduisons également des événements supplémentaires C_1, C_2, C_3, B, D qui permettent d'illustrer la difficulté précédente mais auxquels nous n'associons pas de sens particulier. Les lois de la nature pour E_1 et E_2 sont les suivantes :*

- $(C_1 \wedge C_2) \vee D \leftrightarrow E_1$

- $(C_1 \wedge C_3) \vee B \leftrightarrow E_2$

On considère la situation où C_2 et C_3 sont vrais. Alors $E_1 \wedge \neg D \wedge C_3$ est effectivement suffisant pour E_2 . Dans ce cas, d'après l'approche de Mackie (1980), E_1 est une cause effective de E_2 c'est-à-dire que le fait que le coq chante est une cause effective du fait que le soleil se lève.

Dans cette thèse, nous utilisons une définition de causalité effective proposée par Wright (1985) et qui reprend les principes précédents en résolvant ce dernier problème. Cette définition est basée sur un test NESS (*Necessary Element for the Sufficiency of a Set*). Ce dernier exprime qu'une condition c est la cause d'une conséquence e si et seulement si elle est nécessaire à la suffisance causale d'un ensemble de conditions préalables existantes qui est suffisant pour l'occurrence de e . Autrement dit, cette définition reprend la définition précédente à la différence près que celle de Wright utilise la notion de suffisance causale, différente de la suffisance logique. En effet, la suffisance causale intègre une notion de temporalité : la cause doit précéder la conséquence. D'autre part, pour qu'une conjonction soit suffisante causalement, il faut qu'elle soit effectivement suffisante c'est-à-dire que tous ses littéraux soient vrais à un moment donné dans le contexte considéré. Nous revenons sur l'exemple du chant du coq afin d'illustrer cette nouvelle méthode permet de résoudre le problème de la définition précédente.

Exemple 5 (Chant du coq et lever du Soleil). *(suite) – Nous avons vu que $E_1 \wedge \neg D \wedge C_3$ est effectivement suffisant pour E_2 . D'après l'approche de Mackie (1980), E_1 est une cause effective de E_2 car c'est un élément nécessaire à la suffisance de $E_1 \wedge \neg D \wedge C_3$. Néanmoins, comme E_1 ne précède pas temporellement E_2 , c'est-à-dire que le chant du coq ne précède pas le lever du soleil (ce dernier chante notamment en même temps que les premiers rayons du Soleil), alors E_1 n'est pas nécessaire à la suffisance causale de E_2 . Il ne s'agit donc pas d'une cause effective au sens du test NESS.*

1.2.2 Causalité effective en argumentation abstraite

Pour étudier la génération d'explications causales en argumentation abstraite, il faut d'abord disposer d'une méthode permettant de déterminer les causes effectives de l'acceptabilité d'un argument.

Bochman (2005) établit une équivalence entre les systèmes d'argumentation abstraite et un formalisme logique qu'il appelle un système de raisonnement causal. Ne permettant initialement que des raisonnements basés sur un test *but-for*, ce système a ensuite été étendu pour permettre un raisonnement causal basé une approche contre-factuelle mais cette fois-ci plus complexe (Bochman, 2021). Ses travaux se basent sur un formalisme proche de la logique modale qui diffère de ceux que nous utilisons dans la thèse (les modèles structurels causaux dans les chapitres 2 et 5 et un langage d'action dans les chapitres 3 et 4).

Bengel *et al.* (2022) construisent une base de connaissance causale basée sur un raisonnement contrefactuel pour l'argumentation structurée, formalisme basé les AAF mais qui ajoute une structure logique aux arguments et à la relation d'attaque. Leur définition est l'application directe de la définition de Pearl pour les modèles causaux (Pearl, 2000), et à nouveau similaire au test *but-for*. Ces travaux posent deux problèmes dans le cadre de cette thèse. Tout d'abord, les systèmes d'argumentation structurée étant une instanciation particulières des AAF ajoutant une structure aux arguments et à la notion

d'attaque, il s'agit d'un formalisme offrant une moins bonne expressivité que l'argumentation abstraite que nous considérons ici. Deuxièmement, la définition de causalité effective basée sur le test *but-for* n'est pas toujours adaptée comme nous l'avons vu dans la section précédente.

1.3 Expliquer le statut d'acceptabilité d'un argument

La question de la recherche d'explication en IA a conduit au développement du domaine appelé intelligence artificielle explicable ou *eXplainable Artificial Intelligence* (XAI). L'objectif principal de nos travaux est de générer des explications causales pour l'argumentation abstraite et, plus précisément, nous cherchons à proposer des méthodes permettant d'expliquer le statut d'acceptabilité d'un argument. Avant de présenter les différentes méthodes qui ont été proposées pour répondre à cette tâche, nous introduisons brièvement le domaine de l'XAI puis nous présentons des travaux réalisés autour d'une question centrale dans ce domaine à savoir celle de définir la notion de bonne explication.

1.3.1 Explications en intelligence artificielle : le domaine de l'IA explicable

Le domaine de l'IA explicable est un domaine très vaste qui cherche à enrichir le résultat donné par un système d'IA avec des informations complémentaires, pour fournir à un utilisateur des éléments de réponses à la question "Pourquoi ce résultat est-il généré". Cette question "pourquoi" peut être comprise de nombreuses manières, et les explications générées peuvent être de forme très variée et dépendent, entre autres, de la forme du type de résultats fourni par le système d'IA. Contrairement à nos travaux, une grande partie du domaine se concentre sur des prédicteurs qui résolvent des tâches de classification, régression, segmentation ou parfois de détection d'anomalies ou de clustering. Les méthodes associées pour générer des explications sont très variées (voir par exemple les travaux de Minh *et al.* (2022) ou Ali *et al.* (2023) pour des vues d'ensemble de la littérature) et peuvent conduire à des explications qui diffèrent voire sont en désaccord (Krishna *et al.*, 2024). Une façon de guider le choix et le développement de meilleures explications est, en s'inspirant de travaux en sciences cognitives et en sciences sociales par exemple, de tenter de définir ce qu'est une bonne explication. Il s'agit en fait d'une question centrale dans le domaine de l'XAI.

Chercher à définir la notion d'explication ainsi que ce qui rend une explication "bonne" n'est pas né avec le domaine de l'IA explicable. En effet, la littérature à ce sujet est très riche et va de la philosophie à l'informatique en passant par la psychologie, les sciences sociales, les sciences de l'éducation ou encore les sciences cognitives. En philosophie, on trouve des traces de ces questions dès l'Antiquité avec notamment Aristote qui identifie quatre modes d'explication, qu'il appelle les quatre causes (Aristote, 350 AEC). En psychologie, de nombreuses théories sur la nature d'une explication ont été proposées : lien avec la notion de preuves déductives (Hempel & Oppenheim, 1948), de relations causales (Keil, 2006) ou encore avec la notion de représentation interne ou modèle mental (Holland, 1986). En sciences cognitives, il existe de nombreux travaux étudiant la définition même d'explication, l'objectif des explications ou les caractéristiques des bonnes explications (Srinivasan & Chander, 2021; Byrne, 2019).

Le but de cette thèse n’étant pas de répondre à ces questions mais plutôt de proposer une méthode de génération de bonnes explications en argumentation abstraite, nous nous basons principalement sur les travaux de Miller (2019). Ce dernier, en s’appuyant sur une revue de la littérature en sciences sociales sur la notion de bonne explication, propose des recommandations et souligne notamment quatre caractéristiques principales pour définir une bonne explication :

- Une explication est contrastive c’est-à-dire qu’elle répond à des questions de la forme "Pourquoi X plutôt que Y?"
- Une explication doit être sélective. Il n’est pas attendu d’une explication qu’elle fournisse la totalité des causes qui ont conduit à une conséquence mais plutôt une partie plus courte.
- L’explication la plus probable n’est pas forcément la meilleure explication. Il peut être plus efficace de se référer aux relations causales plutôt qu’à des corrélations.
- Une explication est un processus social de communication. C’est une interaction entre celui qui explique et celui qui reçoit l’explication ou *explainée*.

Un autre point très important repris par Miller (2019) est le lien entre explication et causalité. Bien que différentes d’une explication, les relations causales constituent un des éléments nécessaires pour construire une bonne explication. En effet, cette notion de cause apparaît par exemple dans les deuxième et troisième caractéristiques précédentes et, de façon indirecte, dans l’utilisation de l’adverbe "pourquoi" dans la première. On retrouve d’ailleurs ce lien dès Aristote (350 AEC) avec les quatre causes. C’est cette relation étroite entre ces deux notions qui motive l’approche choisie pour ces travaux de thèse. En particulier, comme nous allons le voir dans la section suivante, on ne retrouve pas ce lien dans beaucoup de méthodes de la littérature en argumentation.

1.3.2 ArgXAI, *argumentative* XAI

Grâce à l’expressivité des méthodes d’IA symbolique pour représenter et raisonner sur des connaissances, une partie de la communauté en IA explicable propose d’utiliser des approches symboliques (voir par exemple Audemard *et al.* 2022; Ferreira *et al.* 2022). Parmi ces méthodes, l’argumentation abstraite a déjà fait ses preuves dans des domaines variés comme le droit (Sartor *et al.*, 2022) ou la médecine (Sassoon *et al.*, 2021). En effet, on trouve dans la littérature en XAI de nombreux travaux sur la génération d’explications en lien avec l’argumentation (voire Čyras *et al.* 2021, Vassiliades *et al.* 2021). L’objectif est le suivant : expliquer l’acceptabilité d’un argument, c’est-à-dire son appartenance à une extension pour une sémantique donnée. Čyras *et al.* (2021) proposent deux critères pour classer ces méthodes : le premier concerne le formalisme de départ, le second concerne la forme de l’explication.

En ce qui concerne le premier critère, Čyras *et al.* (2021) proposent de trier les méthodes d’ArgXAI selon le modèle initial. En effet, certains travaux en argumentation proposent de construire une approximation d’un modèle existant sous la forme d’un graphe d’argumentation (Potyka, 2021; Rago *et al.*, 2022), là où d’autres partent directement d’un modèle sous forme d’AAF. Dans les deux cas, l’explication fournie concerne le statut d’acceptabilité d’un ensemble d’arguments. Toutefois, dans le cas d’une approximation par un graphe d’argumentation abstraite, les arguments renvoient par exemple à des neurones d’un réseau de neurones (Potyka, 2021). Ils peuvent en particulier correspondre aux neurones de la dernière couche d’un tel système d’IA, et l’explication du

statut d’acceptabilité de cet ensemble d’arguments est en fait une explication du résultat fourni par le réseau.

En ce qui concerne le premier critère, Čyras *et al.* (2021) proposent de classer les approches existantes en quatre catégories, détaillées ci-dessous, selon que les explications sont des extensions, des sous-graphes extraits, des dialogues ou des modifications hypothétiques de l’AAF considéré.

Méthodes par extensions – Une première possibilité est donc de renvoyer directement une extension comme explication (voir Gao *et al.* 2016, Zeng *et al.* 2018, Liao & Van Der Torre 2020). Par exemple, considérons $AF = (\mathcal{A}, \mathcal{R})$ et $x \in \mathcal{A}$ un argument. D’après Fan & Toni (2014), une explication S de l’acceptabilité de x est un ensemble admissible lié de sujet x , c’est-à-dire un ensemble d’arguments collectivement acceptables et qui défendent x contre tous ses attaquants : S peut donc être considéré comme expliquant pourquoi x est acceptable. De plus, une explication de x est dite compacte si elle est minimale, et elle est dite verbeuse si elle est maximale au sens de l’inclusion.

Un problème avec cette méthode, et les méthodes par extension de manière générale, est l’absence de raisonnement causal explicite. De fait, même si, comme nous le montrons dans le chapitre 2, la définition de Fan & Toni (2014) est liée à la notion d’explication causale, elle inclut des causes redondantes venant augmenter inutilement la taille de l’explication. Récemment, Borg & Bex (2022, 2024) proposent différentes méthodes par extension incluant des formes de raisonnement causal, par exemple la contrastivité. Ces définitions sont construites à partir d’un ensemble qui regroupe tous les arguments qui jouent un rôle dans le statut d’acceptabilité d’un argument (Borg & Bex, 2021), formant un ensemble similaire à une chaîne causale, c’est-à-dire à une suite de causes ayant conduit à une conséquence. Des explications en sont extraites selon des contraintes de contraste ou bien de nécessité ou de suffisance.

Méthodes par extraction de sous-graphes – Une autre catégorie consiste à identifier des sous-graphes vérifiant un certain nombre de propriétés formelles (Cocarascu *et al.*, 2018). Un type de sous-graphe fréquemment utilisé dans la littérature est appelé les arbres de conflits ou *dispute trees* (Dung *et al.*, 2007). Il s’agit d’un arbre dont chaque nœud correspond à un couple (statut, nom) où le statut du nœud peut valoir partisan (*proponent*) ou opposant (*opponent*), et la seconde composante donne le nom de l’argument. La racine de l’arbre est l’argument dont on cherche à expliquer l’acceptabilité avec un statut de partisan. Chaque argument partisan possède un enfant opposant pour chacun de ses attaquants et chaque argument opposant possède au moins un enfant partisan qui attaque cet opposant. L’objectif est de justifier l’acceptabilité de l’argument initial en invalidant toutes les attaques directes et indirectes contre ce dernier.

Ces objets sont souvent utilisés comme des explications (voir Modgil & Caminada 2009, Fan & Toni 2015a, Čyras *et al.* 2019) car ils vérifient certaines propriétés formelles jugées intéressantes par les auteurs : l’existence d’une explication, sa correction et sa pertinence au sens où chaque argument joue un rôle dans l’acceptabilité de l’argument ciblé. Contrairement aux premières méthodes par extensions, un arbre de conflit peut expliquer l’inacceptabilité d’un argument. Cependant, les arbres de conflits sont également plus proches de la notion de chaînes causales que d’une bonne explication. Par exemple, si l’arbre de conflit est trop profond, le renvoyer entièrement peut nuire à

la compréhension. Une telle explication peut donc être de grande taille, et ne pas respecter la propriété de sélectivité (Miller, 2019).

Méthodes par génération de dialogues – Un troisième type d'explication argumentative est basée sur la notion de dialogue argumentatif. Ce dernier consiste en une discussion fictive entre le *proponent* et l'*opponent* dans laquelle les participants échangent des arguments du graphe selon un ensemble de règles (Black *et al.*, 2021) jusqu'à obtenir la victoire, c'est-à-dire l'acceptabilité de l'argument à expliquer. Ces dialogues peuvent être construits à partir des arbres de conflit, auquel cas le contenu est similaire et seule la forme change. Ils peuvent aussi être construits directement à partir du graphe d'argumentation (Madumal *et al.*, 2019). Cette dernière méthode a l'avantage de proposer un protocole d'explication interactif faisant de l'explication un processus social.

Méthodes "contrefactuelles" – Enfin, une dernière famille de méthodes se concentre sur les changements à faire sur le graphe, c'est-à-dire l'ajout ou la suppression d'arguments ou d'attaques, pour obtenir un changement sur le statut d'acceptabilité d'un argument (voir les travaux de Fan & Toni 2015b, Saribatur *et al.* 2020, Kampik *et al.* 2024). Ces méthodes peuvent être vues comme un raisonnement contrefactuel, les plaçant ainsi dans la famille des explications contrefactuelles, qui respectent la caractéristique de contrastivité souhaitée pour une explication. Toutefois, le raisonnement contrefactuel effectué s'apparente à une forme de test *but-for*, c'est-à-dire qu'il étudie le résultat d'un scénario hypothétique dans lequel la cause présumée ne s'est pas produite. Cependant, comme nous le détaillons dans la section 1.2.1, ce test ne parvient pas à résoudre des situations causales complexes : les causes identifiées par ce test ne sont pas satisfaisantes dans les cas de surdétermination.

La relation d'attaque en argumentation abstraite peut être vue comme une forme de dépendance causale. En effet, un argument ne peut être acceptable que si tous ses attaquants sont inacceptables. Pour cette raison, toutes les méthodes décrites dans cette section renvoient une explication liée à la chaîne causale du statut d'acceptabilité d'un argument. Toutefois, d'une part certaines ne se basent pas sur un cadre causal formel permettant des raisonnements causaux complexes, et les "causes" identifiées peuvent par conséquent ne pas être satisfaisantes. D'autre part, comme nous en discutons dans le chapitre 3, extraire la chaîne causale du statut d'acceptabilité d'un argument n'est que la première étape pour la génération d'explications causales pour l'argumentation abstraite. En effet, bien qu'étroitement liées les notions de causalité et d'explications sont deux notions différentes (Miller, 2019). Une étape d'extraction d'explications selon par exemple les différentes recommandations de Miller (2019) est nécessaire.

1.4 Temporalité en argumentation abstraite

Comme nous l'avons présenté dans l'introduction de cette thèse, un problème des systèmes classiques d'argumentation abstraite, c'est-à-dire tels qu'introduits par Dung (1995), est qu'il s'agit d'un cadre statique qui ne prend pas en compte l'ordre d'énonciation des arguments. Or, la temporalité constitue une notion essentielle en causalité et elle apparaît même explicitement dans la définition d'une cause effective selon le test NESS (voir section 1.2.1). Dans cette section, nous décrivons trois familles d'approches

qui permettent d’étendre les systèmes d’argumentation abstraite pour inclure une notion de temporalité dans un AAF. Dans cette section, contrairement aux dialogues argumentatifs de la section précédente, on utilise le mot *dialogue* pour désigner une séquence d’arguments abstraits ordonnée temporellement.

Ajout d’une date – Une première famille de méthodes se base sur les outils de l’argumentation pour introduire la temporalité. Ainsi, Cobo *et al.* (2010) étendent les AAF en ajoutant une fonction de disponibilité qui, à chaque argument, associe un intervalle pendant lequel il est présent. Un argument est alors acceptable si et seulement s’il est disponible et que tous ses attaquants sont soit indisponibles soit inacceptables. Une autre possibilité est d’ajouter une date aux AAF en eux-mêmes, c’est-à-dire de considérer des suites d’AAF, comme proposé par Kampik *et al.* (2024) pour des systèmes d’argumentation bipolaire pondérée (voir chapitre 5). En particulier, ils proposent de créer un graphe d’argumentation à chaque pas de temps. Cette séquence de graphes ainsi créée, un historique des différents états du dialogue est obtenu.

Utilisation d’opérateurs de révision de croyances – Une deuxième famille de méthodes cherche à représenter les changements à réaliser sur un graphe d’argumentation abstraite pour atteindre un état donné avec des contraintes fixées (Doutre & Mailly, 2018) en passant généralement par un autre formalisme logique. Par exemple, la méthode YALLA (Dupin de Saint-Cyr *et al.*, 2016) propose d’utiliser des opérateurs de révision de croyances pour mettre à jour un système d’argumentation abstraite. Elle offre un langage très expressif qui permet d’identifier les relations d’attaque ou les arguments qui devraient être supprimés ou ajoutés pour atteindre un statut d’acceptabilité donné au pas de temps suivant. Un tel raisonnement peut être utilisé pour représenter l’évolution d’un AAF. Cependant, le nombre de termes manipulés par YALLA augmente de manière exponentielle avec le nombre d’arguments, ce qui ne permet pas de modéliser l’ensemble d’un dialogue. Dans la même catégorie de méthodes, Doutre *et al.* (2017) proposent d’utiliser un langage propositionnel dynamique pour modéliser le graphe d’argumentation et son évolution à travers le dialogue.

Représentation sous forme de système de transition d’états – Enfin, Barringer *et al.* (2005) proposent différentes échelles de temporalité en argumentation abstraite : une au niveau des arguments en eux-mêmes et une autre au niveau du graphe d’argumentation en général. Ils proposent ensuite plusieurs méthodes pour inclure cette notion selon l’échelle recherchée. Parmi celles-ci, ils suggèrent, sans la formaliser, d’inclure une action permettant d’ajouter des arguments. Cette idée a été reprise dans des travaux qui visent à modéliser la persuasion ou encore la négociation entre différents agents. En particulier, Arisaka & Satoh (2018) proposent une extension des AAF intitulés systèmes d’argumentation et de persuasion abstraite (*Abstract Persuasion Argumentation* ou APA). Ces derniers, en plus d’ajouter une relation ternaire modélisant le processus de persuasion, définissent un opérateur de transition permettant de passer d’un état, c’est-à-dire un APA, au suivant en ajoutant ou retirant des arguments et des relations.

Dans cette thèse, nous proposons une autre formalisation du principe de Barringer *et al.* (2005), basée sur un formalisme logique adapté, les langages d’action (voir

chapitre 3). Ils permettent d’obtenir une description de la temporalité du système d’argumentation abstraite au niveaux des arguments eux-mêmes mais aussi au niveau du graphe d’argumentation.

1.5 Bilan

Dans ce chapitre, nous avons présenté différents travaux existant dans la littérature qui répondent en partie aux questions que nous nous posons dans cette thèse. Il existe en particulier de nombreux travaux en argumentation abstraite pour expliquer le statut d’acceptabilité d’un argument. Toutefois pour chacune de ces méthodes, nous identifions au moins deux défauts auxquels les méthodes que nous proposons dans cette thèse tentent de répondre :

1. elles ne se basent pas sur un raisonnement causal complexe permettant d’identifier les chaînes causales avant d’en extraire des explications ;
2. à l’exception de l’approche de Kampik *et al.* (2024), elles se concentrent sur les systèmes classiques d’argumentation abstraite qui ne prennent pas en compte l’ordre d’énonciation des arguments.

Pour cette raison, nous avons tout d’abord identifié plusieurs travaux en particulier ceux de Bochman (2021) qui permettent d’effectuer un raisonnement causal complexe adapté à l’argumentation (Bochman, 2005). Comme nous l’avons présenté dans la section 1.3.2, ce raisonnement est la première étape vers la génération d’explications causales. Cependant, ce dernier n’offre pas les mêmes outils que le formalisme de Halpern (2016) : les modèles structurels causaux. En effet, ces modèles proposent une définition formelle de la notion d’explication basée sur la notion de cause effective et qui permet en plus d’inclure le destinataire de l’explication en restreignant l’espace des scénarios contrefactuels à ceux compatibles avec la connaissance de ce destinataire. Pour cette raison, nous proposons dans le chapitre 2 d’utiliser les modèles structurels causaux pour générer des explications causales adaptées au destinataire.

Cependant, le formalisme de Bochman (2021) tout comme les modèles structurels causaux ne permettent pas de modéliser l’ordre d’énonciation des arguments. Nous avons donc également étudié les différentes approches existantes en argumentation abstraite permettant de représenter un dialogue. Elles n’incluent pas de notion de causalité nécessaire pour la génération d’explications causales. Nous proposons donc dans les chapitres 3 et 4 d’utiliser un langage d’action, formalisme logique naturellement dynamique, pour enrichir un système d’argumentation abstraite avec l’ordre d’énonciation des arguments. De plus, ce langage d’action (Sarmiento *et al.*, 2022) propose une définition d’une cause effective basée sur le test NESS, permettant ainsi d’extraire les chaînes causales du statut d’acceptabilité d’un argument. Enfin, à notre connaissance aucun travail en argumentation abstraite ne propose de lien avec les approches par régularité comme par exemple le test NESS.

Pour conclure, nous proposons dans cette thèse d’utiliser deux formalismes causaux différents, une approche contrefactuelle et une approche par régularité, pour la génération d’explications causales pour l’argumentation abstraite dans un cadre classique puis dans un cadre temporel, et enfin dans un cadre bipolaire pondéré.

Graphes d'argumentation abstraite acycliques et modèles structurels causaux

Ce chapitre présente une équivalence entre les systèmes d'argumentation abstraite acycliques et un cas particulier de modèles structurels causaux que nous appelons modèles structurels causaux argumentatifs. Pour cela, nous proposons une transformation et sa réciproque permettant de passer d'un formalisme à l'autre. À l'aide de ces deux transformations, nous démontrons un théorème établissant cette équivalence et nous proposons de générer des explications causales sur l'état d'acceptabilité d'un ensemble d'arguments en transformant un AAF en graphe modèle structurel causal argumentatif.

Ces travaux ont fait l'objet de deux publications :

- *dans le workshop international AI and Cognition (Munro et al., 2022a)*
- *dans la conférence française RJCIA (Munro et al., 2022b).*

Sommaire

2.1	Introduction	30
2.2	Rappels préliminaires sur les modèles structurels causaux	30
2.2.1	Définition d'un modèle structurel causal	30
2.2.2	Cause effective	32
2.2.3	Cause suffisante	33
2.2.4	Explication	35
2.2.5	Choix du modèle structurel causal à adopter	36
2.3	Passage des AAF aux graphes causaux argumentatifs	37
2.3.1	Graphes structurels causaux argumentatifs (GCA)	37
2.3.2	Transformation proposée $\mathcal{T}_{AFa2GCA}$	38
2.3.3	Exploitation des GCA pour la génération d'explications	39
2.4	Passage des GCA aux AAF	40
2.4.1	Transformation proposée $\mathcal{T}_{GCA2AFa}$	40
2.4.2	Théorème d'équivalence entre AAF et GCA	41
2.4.3	Équivalence et inverse à gauche	42

2.1 Introduction

Pour répondre à la question de la génération d'explications causales en argumentation abstraite, deux méthodes sont envisageables : proposer une définition formelle de la notion de cause effective en argumentation, ou bien, comme l'a fait Bochman (2005), utiliser un autre formalisme dans lequel une telle définition existe. C'est cette deuxième option que nous avons choisie, en nous basant sur les modèles structurels causaux proposés par Halpern (2016) et présentés de façon informelle dans la section 1.2.1 du chapitre précédent. La section 2.2 décrit formellement les modèles structurels causaux et leur utilisation pour générer des explications. Dans la section 2.3, nous présentons une transformation permettant de passer d'un AAF à un cas particulier de modèle structurel causal et nous discutons de l'apport de ces modèles pour la génération d'explications causales pour l'argumentation abstraite. Enfin, dans la section 2.4, nous présentons la transformation réciproque et établissons l'équivalence entre les AAF et ce cas particulier de modèles structurels causaux.

2.2 Rappels préliminaires sur les modèles structurels causaux

Dans cette section, nous rappelons la définition des modèles structurels causaux définis par Halpern & Pearl (2005). Ils offrent un cadre formel adapté à une définition de causalité effective basée sur la notion de scénario contrefactuel. À partir de cette définition, Halpern (2016) propose également une formalisation de la notion d'explication.

2.2.1 Définition d'un modèle structurel causal

Afin de pouvoir raisonner sur les causes ayant conduit à une conséquence, il est nécessaire de déterminer une base commune sur laquelle ce raisonnement aura lieu. Une telle base commune est proposée par les modèles structurels causaux. Ces derniers décrivent le monde par l'intermédiaire de variables et de relations causales entre ces variables.

Définition 2.1 (Modèle structurel causal (Halpern & Pearl, 2005)).

Un modèle structurel causal est un triplet $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ où :

- *$\mathcal{U}$ est un ensemble fini de variables exogènes, c'est-à-dire un ensemble de variables dont les valeurs sont déterminées par des facteurs extérieurs au modèle ;*
- *$\mathcal{V}$ est un ensemble fini de variables endogènes, c'est-à-dire un ensemble de variables dont les valeurs peuvent être calculées à partir des valeurs des autres variables ;*
- *$\mathcal{F}$ est l'ensemble des équations structurelles du modèle. Il contient une équation pour chaque variable de $\mathcal{V}$. Ces équations permettent d'associer une valeur à chacune des variables endogènes en fonction des valeurs des autres variables du modèle.*

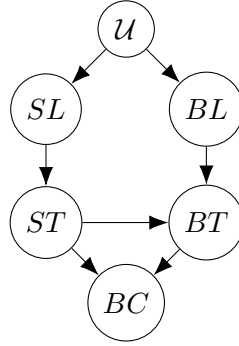


FIGURE 2.1 – Exemple du lancer de pierre.

Définition 2.2 (Graphe causal associé). Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ un modèle structurel causal.

En associant à chaque variable de $\mathcal{U}$ et de $\mathcal{V}$ un nœud et en traçant des arcs entre ces nœuds pour indiquer les dépendances fonctionnelles de $\mathcal{F}$, on obtient une représentation du modèle structurel M sous la forme d'un graphe orienté. On appelle ce graphe le graphe causal associé à M .

Exemple 4 (Lancer de pierre). (suite) – Nous reprenons ici l'exemple de Suzy et Billy présenté au chapitre précédent.

Pour modéliser cet exemple sous la forme d'un modèle structurel causal, on introduit les variables suivantes : SL (respectivement BL) et ST (respectivement BT) représentent "Suzy (respectivement Billy) lance une pierre" et "Suzy (respectivement Billy) touche la bouteille". Enfin, BC représente "la bouteille se casse".

On introduit également l'ensemble $\mathcal{U} = \{SL^u, BL^u\}$ de variables exogènes qui représentent les facteurs extérieurs au problème qui influencent le fait que Billy ou Suzy lancent la pierre.

Les fonctions de $\mathcal{F}$ complètent la modélisation du problème. La première hypothèse est que Suzy touche la bouteille si et seulement si elle a lancé. Cela se traduit par l'équation structurelle suivante : $ST = SL$. Billy quant à lui ne touche la bouteille que dans le cas (et uniquement dans ce cas) où il a lancé une pierre et où Suzy n'a pas touché la bouteille c'est-à-dire : $BT = BL \wedge \neg ST$. Enfin si Suzy ou Billy touchent la bouteille alors celle-ci se casse : $BC = ST \vee BT$.

On obtient le modèle structurel causal suivant :

- $\mathcal{U} = \{SL^u, BL^u\}$
- $\mathcal{V} = \{SL, BL, ST, BT, BC\}$
- $\mathcal{F} = \{(SL = SL^u), (BL = BL^u), (ST = SL), (BT = BL \wedge \neg ST), (BC = ST \vee BT)\}$

Le graphe causal associé à ce modèle structurel est donné par la figure 2.1.

Dans la suite, les définitions et notations supplémentaires suivantes sont utilisées :

- Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$, un modèle structurel causal. On appelle *contexte*, noté $\mathbf{u}$, une affectation des variables de $\mathcal{U}$. Le couple $(M, \mathbf{u})$ est appelé *monde*. Dans la suite, on note $\mathcal{K}$ un ensemble de contextes.
- Soit $X \in \mathcal{U} \cup \mathcal{V}$ une variable. On note $\mathcal{R}(X)$ l'ensemble des valeurs que peut prendre la variable X dans M . Par abus de notation, on note également $\mathcal{R}(\mathbf{X})$ l'ensemble des valeurs que peut prendre l'ensemble de variables $\mathbf{X}$.

- Soit $\mathbf{X}$ un ensemble de variables de $\mathcal{V}$ et $\mathbf{x} \in \mathcal{R}(\mathbf{X})$ un ensemble de valeurs de $\mathbf{X}$, on note $\mathbf{X} = \mathbf{x}$ une affectation des variables de $\mathbf{X}$ avec les valeurs de $\mathbf{x}$.
- Soit $\mathbf{u} \in \mathcal{K}$, on note $(M, \mathbf{u}) \models \mathbf{X} = \mathbf{x}$ si $\mathbf{X} = \mathbf{x}$ est l'unique solution aux équations de $\mathcal{F}$ dans $\mathbf{u}$.
- Soit $\mathbf{X} \subseteq \mathcal{V}$ et $\mathbf{x} \in \mathcal{R}(\mathbf{X})$ des valeurs de $\mathbf{X}$. On note $\mathcal{K}_{\mathbf{X}=\mathbf{x}}$, l'ensemble des contextes $\mathbf{u}'$ de $\mathcal{K}$ tels que $(M, \mathbf{u}') \models \mathbf{X} = \mathbf{x}$.
- Soit $\mathbf{u} \in \mathcal{K}$, la notation $(M, \mathbf{u}) \models [\mathbf{X} = \mathbf{x}](\mathbf{Y} = \mathbf{y})$ signifie que l'on se place dans le monde $(M', \mathbf{u})$ correspondant au monde $(M, \mathbf{u})$ dans lequel les équations de $\mathcal{F}$ portant sur les variables de $\mathbf{X}$ sont remplacées par l'équation $\mathbf{X} = \mathbf{x}$. Dans ce monde contrefactuel $(M', \mathbf{u})$, $\mathbf{Y} = \mathbf{y}$ est l'unique solution aux équations de $\mathcal{F}$.

Exemple 4 (Lancer de pierre). *(suite) – On se place dans le cas où Suzy et Billy ont tous les deux lancé une pierre c'est-à-dire le contexte $\mathbf{u} = \{SL^u = 1, BL^u = 1\}$. Dans ce monde, le fait que la bouteille se casse est la seule possibilité pour l'état de la bouteille, c'est-à-dire que $(M, \mathbf{u}) \models BC = 1$. Cependant, si on considère le scénario contrefactuel dans lequel ni Suzy ni Billy n'ont lancé la pierre alors la bouteille ne peut pas s'être cassée. Formellement, $(M, \mathbf{u}) \models [(SL, BL) = (0, 0)](BC = 0)$.*

2.2.2 Cause effective

À partir de ce formalisme, plusieurs définitions de la notion de causalité effective, notées définitions HP, ont été proposées (Halpern, 2016) : la définition HP originelle, la version mise à jour de HP et la version modifiée. Dans cette thèse, nous présentons et utilisons seulement la version modifiée (Halpern, 2016) car il s'agit de la dernière en date, et elle permet de couvrir des scénarios plus complexes que les deux autres comme la surdétermination par exemple.

Définition 2.3 (Cause effective). *Soit φ une affectation de la forme $\mathbf{Y} = \mathbf{y}$, l'affectation $\mathbf{X} = \mathbf{x}$ est une cause effective de la conséquence φ dans le monde $(M, \mathbf{u})$ si les trois conditions suivantes sont vérifiées :*

AC1 $(M, \mathbf{u}) \models (\mathbf{X} = \mathbf{x}) \wedge \varphi$, c'est-à-dire la cause et la conséquence sont toutes les deux vraies dans le monde considéré.

AC2 Il existe un ensemble $\mathbf{W}$ de variables endogènes, une configuration $\mathbf{x}'$ pour l'ensemble de variable $\mathbf{X}$ et pour toutes valeurs $\mathbf{w}$ de $\mathbf{W}$, si $(M, \mathbf{u}) \models (\mathbf{W} = \mathbf{w})$ alors :

$$(M, \mathbf{u}) \models [\mathbf{X} = \mathbf{x}', \mathbf{W} = \mathbf{w}] \neg \varphi.$$

AC3 $\mathbf{X}$ est minimal au sens de l'inclusion : il n'existe pas de sous-ensemble strict de $\mathbf{X}$ qui satisfasse **AC1** et **AC2**. Cette dernière condition empêche d'avoir des variables inutiles dans la cause.

Les conditions **AC1** et **AC3** sont identiques quelle que soit la version de cette définition. Le point critique se situe dans la condition **AC2**. En effet, c'est elle qui représente le raisonnement contrefactuel utilisé. Comme présenté dans le chapitre 1, un test *but-for*, qui aurait la même forme en omettant $\mathbf{W}$, n'est pas suffisant pour couvrir les cas de surdétermination. Pour y remédier, un ensemble de variables *témoins* différentes de la cause et de la conséquence étudiées, $\mathbf{W}$ dans la définition, est choisi. Cet ensemble

garde les même valeurs que dans le scénario original quel que soit le scénario hypothétique envisagé. Utiliser ces variables témoins garantit que les effets potentiels que la cause étudiée aurait pu avoir sur d'autres variables n'ont pas lieu. Ainsi, s'il existe un scénario contrefactuel dans lequel la cause présumée ne s'est pas produite ($\mathbf{X} = \mathbf{x}'$) et éventuellement d'autres événements se sont quand même produits ($\mathbf{W} = \mathbf{w}$), tels que la conséquence ne se produise pas, alors la cause présumée est bien une cause effective.

Exemple 4 (Lancer de pierre). *(suite) – Comme nous l'avons vu dans le chapitre 1, une cause intuitive du fait que la bouteille se casse est le fait que Suzy a lancé la pierre. Pour illustrer qu'un test but-for ne suffit pas ici, on peut se poser la question : si Suzy n'avait pas lancé sa pierre, la bouteille se serait-elle cassée ? On pose donc $\varphi = (BC = 1)$ et la cause présumée $SL = 1$ c'est-à-dire $\mathbf{X} = SL$, $\mathbf{x} = 1$ et $\mathbf{x}' = 0$. La réponse est oui car Billy aurait alors touché la bouteille : $(M, \mathbf{u}) \models [SL = 0](BT = 1 \wedge BC = 1)$ car $BT = BL \wedge \neg ST$ et $BC = ST \vee BT$.*

*Ce problème vient du fait que, dans le contexte réel, Billy n'a pas touché la bouteille car Suzy l'a touchée. Ici, la cause présumée a un impact sur d'autres variables que la conséquence étudiée. Il faut donc envisager le scénario contrefactuel suivant : si Suzy n'avait pas touché la bouteille **et sachant que** Billy ne l'a pas touchée, la bouteille se serait-elle cassée ? Pour y répondre, on considère comme témoin la variable BT et la valeur 0, c'est-à-dire que $(\mathbf{W} = \mathbf{w}) = (BT = 0)$. Dans ce cas, on a $(M, \mathbf{u}) \models [SL = 0 \wedge BT = 0](BC = 0)$. La condition **AC2** est bien vérifiée c'est-à-dire que la réponse est effectivement non. De plus comme **AC1** et **AC3** sont également vérifiées on obtient le fait que Suzy ait lancé sa pierre est bien une cause effective du fait que la bouteille se casse.*

2.2.3 Cause suffisante

Avant de définir formellement ce qu'est une explication dans ce formalisme, il faut introduire la notion de cause suffisante. Intuitivement, contrairement à une cause effective pour laquelle la définition se base un scénario contrefactuel, une cause suffisante correspond à un ensemble d'événements qui, peu importe la situation du monde, permet d'obtenir la conséquence dont on cherche les causes. Formellement :

Définition 2.4 (Cause suffisante). *Soit $\mathcal{K}$ un ensemble de contextes et $\mathbf{u} \in \mathcal{K}$. L'affectation $\mathbf{X} = \mathbf{x}$ est une cause suffisante de φ dans le monde $(M, \mathbf{u})$ si les quatre conditions suivantes sont vérifiées :*

SC1 $(M, \mathbf{u}) \models (\mathbf{X} = \mathbf{x}) \wedge \varphi$.

SC2 *Il existe une partie de $\mathbf{X}$, $\mathbf{X}' = \mathbf{x}'$, et une autre conjonction $(\mathbf{Y} = \mathbf{y})$ (éventuellement vide) telles que $(X = x) \wedge (\mathbf{Y} = \mathbf{y})$ est une cause effective de φ dans $(M, \mathbf{u})$, c'est-à-dire une partie de $\mathbf{X}$ est une partie d'une cause effective dans le monde considéré.*

SC3 $(M, \mathbf{u}') \models [\mathbf{X} = \mathbf{x}]\varphi$ pour tous les contextes $\mathbf{u}' \in \mathcal{K}$, c'est-à-dire si l'on a $\mathbf{X} = \mathbf{x}$ alors on a φ quel que soit le contexte considéré.

SC4 $\mathbf{X}$ est minimal au sens de l'inclusion.

La condition SC3 est le point clef de cette définition et renvoie à l'intuition précédente puisque, quel que soit le contexte du monde, si les événements $\mathbf{X} = \mathbf{x}$ se produisent alors la conséquence φ a également lieu.

Exemple 4 (Lancer de pierre). (suite) – Considérons l'ensemble de tous les contextes possibles c'est-à-dire : $\forall X \in \mathcal{U}, \forall x \in \mathcal{R}(X), \exists \mathbf{u} \in \mathcal{K}$ tel que $(M, \mathbf{u}) \models X = x$. En particulier, pour cet exemple, $\mathcal{K} = \{\{SL^u = 1, BL^u = 1\}, \{SL^u = 1, BL^u = 0\}, \{SL^u = 0, BL^u = 1\}, \{SL^u = 0, BL^u = 0\}\}$. On se place dans le contexte $\mathbf{u}$ décrit précédemment c'est-à-dire $u = \{SL^u = 1, BL^u = 1\}$.

Comme $BC = ST \vee BT$ alors $BT = 1$ et $ST = 1$ vérifie la condition SC3. Cependant, comme $(M, \mathbf{u}) \not\models BT = 1$ alors bien que le fait que Billy touche la bouteille soit suffisant pour que la bouteille se casse, il ne s'agit pas d'une cause suffisante dans ce contexte. D'autre part, $ST = 1$ est une partie d'une cause effective de $BC = 1$ et est de plus minimal, donc il s'agit d'une cause suffisante.

La condition SC2 sert uniquement à restreindre la recherche de causes suffisantes parmi les ensembles d'événements dont une partie est une partie d'une cause effective. Bien que ces deux concepts soient liés, une cause effective est locale et dépend d'un contexte donné, alors qu'une cause suffisante est globale et dépend d'un ensemble de contextes. Pour illustrer l'apport de la condition SC2 de façon plus concrète, on considère un autre exemple introduit par Halpern (2016).

Exemple 6 (Feu de forêt). On se place dans un monde où deux personnes, Alice et Bob, sont dans une forêt avec chacun une allumette allumée dans la main. On note $AL = 1$ (respectivement $BL = 1$) le fait que Alice (respectivement Bob) lâche l'allumette, le cas $AL = 0$ (respectivement $BL = 0$) représentant le fait qu'elle (respectivement il) l'éteigne. Pour savoir si la forêt prend feu, on considère deux scénarios :

CC - Cas conjonctif : la forêt s'embrase si et seulement si les deux lâchent l'allumette, c'est-à-dire $FF = AL \wedge BL$.

CD - Cas disjonctif : il suffit que l'un des deux la lâche pour que la forêt prenne feu, c'est-à-dire $FF = AL \vee BL$.

On considère en particulier le contexte $\mathbf{u}$ où Alice et Bob lâchent l'allumette. Selon le cas considéré, on obtient les causes effectives suivantes :

CC - $AL = 1$ et $BL = 1$ sont deux causes effectives de $FF = 1$, c'est-à-dire que Alice et Bob sont chacun une cause effective du feu de forêt.

CD - $AL = 1 \wedge BL = 1$ est une cause effective de $FF = 1$, c'est-à-dire que Alice et Bob sont la cause effective du feu de forêt.

Maintenant, si on s'intéresse aux causes suffisantes pour que la forêt brûle, la situation s'inverse. En effet, avec $\mathcal{K}$ l'ensemble de tous les contextes possibles, on obtient :

CC - $AL = 1 \wedge BL = 1$ est une cause suffisante de $FF = 1$. En effet, il suffit que Alice et Bob lâchent l'allumette pour déclencher un feu de forêt. Pourtant, Alice et Bob sont indépendamment des causes effectives de ce feu, c'est-à-dire que seule une partie de la cause suffisante est une cause effective de $FF = 1$.

CD - $AL = 1$ et $BL = 1$ sont des causes suffisantes de $FF = 1$. En effet, il suffit que Alice ou Bob lâche l'allumette pour déclencher un feu de forêt. Pourtant, Alice et Bob ensemble sont la cause effective de ce feu, c'est-à-dire qu'une cause suffisante est une seulement une partie de la cause effective de $FF = 1$.

Remarque – Il existe une deuxième version de la définition de cause suffisante proposée par Miller (2021). Il stipule qu'une cause suffisante est une cause effective non minimale, c'est-à-dire qui ne vérifie que **AC1** et **AC2**. La différence majeure se situe

dans **SC3**. Cette version de la notion de cause suffisante se concentre uniquement sur le contexte en cours, contrairement à celle de Halpern qui définit une cause suffisante sur un ensemble de contextes donnés. Nous faisons le choix ici de considérer la définition de Halpern, notamment car en affaiblissant **SC3** on peut définir une notion de pouvoir explicatif utile pour comparer les explications générées (Halpern, 2016).

2.2.4 Explication

Comme nous l'avons rappelé dans le chapitre d'introduction, une explication est un processus social, une interaction entre au moins deux entités : la personne qui fournit l'explication et la personne qui la reçoit appelée l'*explainée*. Lorsque l'on fournit une explication, il est donc normal de tenir compte du destinataire de l'explication. En particulier, il serait peu intéressant de fournir une explication à partir d'une situation que l'interlocuteur considère comme impossible. Pour cette raison, dans la définition d'une explication, la recherche de causes effectives et de causes suffisantes est restreinte à un ensemble de contextes $\mathcal{K}$ déterminé par ce que l'*explainée* considère comme possible.

Définition 2.5 (Explication). *L'affectation $X = x$ est une explication de φ relative à l'ensemble de contextes $\mathcal{K}$ si les trois conditions suivantes sont vérifiées :*

EX1 $X = x$ est une cause suffisante pour tous les contextes u dans $\mathcal{K}$ qui vérifient $(X = x) \wedge \varphi$.

EX2 X est minimal.

EX3 $\mathcal{K}_{(X=x) \wedge \varphi} \neq \emptyset$, c'est-à-dire les contextes considérés comme possibles par l'*explainée* sont compatibles avec l'explication.

L'explication est dite non triviale si elle vérifie de plus :

EX4 $(M, u') \models \neg(X = x)$ pour certains contextes $u' \in K_\varphi$.

L'ensemble de contextes $\mathcal{K}$ est déterminé par l'*explainée* et correspond à ses connaissances sur le monde. Il est donc possible que cet ensemble soit trop restrictif. Dans ce cas et avec cette définition, il n'existe aucune explication adaptée au destinataire. En effet, s'il n'existe pas de causes suffisantes dans au moins un contexte de $\mathcal{K}$ (c'est-à-dire $\mathcal{K}_{(X=x) \wedge \varphi} = \emptyset$) alors, il n'y a pas d'explication.

Pour remédier à ce problème, il existe une version plus générale de cette définition (Halpern, 2016). Elle prend également en compte le fait que le destinataire de l'explication n'a pas une connaissance parfaite du modèle, et donc l'explication doit apporter une connaissance supplémentaire. L'explication n'est alors plus seulement une cause suffisante selon un ensemble de contextes. Des équations structurelles permettant au destinataire de mieux comprendre ce modèle sont ajoutées à l'explication. Ainsi, si aucune cause suffisante n'existe pas dans l'ensemble de contextes $\mathcal{K}$ considéré par l'*explainée*, alors renvoyer une formule en plus peut permettre à ce dernier d'élargir l'ensemble $\mathcal{K}$ des contextes possibles. En particulier, si il n'existe pas d'explication au sens de la définition précédente, on renvoie ici un couple formé de l'ensemble vide et d'une ou plusieurs connaissances supplémentaires, en particulier des équations structurelles, permettant d'étendre cet ensemble de contextes. Par conséquent, si cela est souhaité par l'*explainée*, il peut demander une nouvelle explication avec un ensemble $\mathcal{K}' \neq \mathcal{K}$ dans lequel une explication non vide existe peut-être.

Dans la suite, cette généralisation n'est pas considérée.

Remarque – Afin d'améliorer la qualité des explications causales générées, plusieurs extensions peuvent être ajoutées à cette définition. Tout d'abord, une notion de normalité et d'anormalité a également été définie par Halpern (2016) permettant de privilégier les événements anormaux pour des explications. Une extension contrastive de ces définitions a également été proposée (Miller, 2021) ainsi qu'une extension floue (Bloch & Lesot, 2022). Dans cette thèse, nous n'étudions que la définition 2.5 et laissons ces dernières comme piste d'amélioration.

2.2.5 Choix du modèle structurel causal à adopter

Le formalisme des modèles structurels causaux est un cadre très intéressant pour la génération d'explications adaptées à l'*explainée*. Par exemple, la connaissance de l'utilisateur est prise en compte lors de la recherche d'explications ce qui permet de les adapter au destinataire. Toutefois, pour que ces explications soient correctes, il faut que le modèle structurel causal choisi représente le monde de façon "adaptée". Il s'agit d'un point très important car une modélisation un peu différente d'un problème identique peut conduire à des causes qui n'en sont plus ou inversement. Considérons l'exemple suivant :

Exemple 7 (Circuit électrique). *On considère un circuit électrique composé de deux sources d'électricité A et B et d'un commutateur électrique permettant de connecter une lampe à l'une des deux sources d'énergie. On note A et B les deux variables endogènes telles que $A = 1$ (respectivement $B = 1$) représente le fait que la source d'énergie A (respectivement B) est allumée. On note également S tel que $S = 0$ (respectivement $S = 1$) indique que la lampe est reliée à la source d'énergie A (respectivement B). Enfin, $L = 1$ signifie que la lampe est allumée. Finalement, on définit l'équation structurelle de fonctionnement de la lampe suivante : $L = (A \wedge \neg S) \vee (B \wedge S)$.*

Dans le contexte u tel que $A = B = S = 1$, c'est-à-dire que les deux sources sont allumées et que la lampe est reliée à B , le fait que la source B soit allumée est une cause effective du fait que la lampe est allumée. Cependant, le fait que le commutateur relie la source B à la lampe n'en est pas une, là où le fait que la source A est active et que le commutateur relie la source B à la lampe en est une.

Exemple 7 (Circuit électrique). *(modifié) – Ajoutons maintenant deux variables intermédiaires pour représenter d'où vient l'électricité qui alimente la lampe. On pose $G = A \wedge \neg S$ (respectivement $D = B \wedge S$) pour dire que le courant vient de la gauche (respectivement de la droite). L'équation structurelle associée à S devient $S = G \vee D$.*

Avec cette nouvelle modélisation, toujours dans le même contexte, "la source d'énergie B est active" est toujours une cause effective du fait que la lampe est allumée. Mais cette fois-ci, le fait que le commutateur relie la source B avec la lampe ($S = 1$) est également une cause effective. En raison de **AC3**, le fait que la source A est active et que le commutateur relie la source B à la lampe ne peut alors pas être une cause effective. Ainsi, alors que les deux modélisations précédentes représentent exactement le même scénario et reproduisent les mêmes résultats que la réalité, les causes effectives trouvées sont différentes.

Afin de pouvoir appliquer le formalisme des modèles structurels causaux dans un cas précis, nous avons d'abord besoin de construire un tel modèle. Or comme nous l'illustrons

avec l'exemple précédent, cela peut s'avérer très difficile. Une solution possible serait de réussir à lier ce formalisme à celui des systèmes d'argumentation abstraite pour lesquels il existe des méthodes de fouilles d'arguments permettant d'obtenir un AAF. Cela permettrait d'utiliser le modèle d'argumentation obtenu pour générer ensuite un modèle structurel causal. C'est ce que nous proposons dans les sections 2.3 et 2.4.

2.3 Passage des AAF aux graphes causaux argumentatifs

Dans cette section, nous proposons une transformation d'un AAF acyclique en graphe causal argumentatif, un cas particulier de modèle structurel causal. Une comparaison de la définition d'une explication de Fan & Toni (2014) à celle de Halpern (2016) est également présentée et illustrée grâce à la transformation proposée.

2.3.1 Graphes structurels causaux argumentatifs (GCA)

Nous proposons de considérer un cas particulier des modèles structurels causaux, que nous appelons graphes structurels causaux argumentatifs (GCA), définis de la façon suivante :

Définition 2.6 (Graphe causal argumentatif). *Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ un modèle structurel causal. M est appelé un graphe causal argumentatif (GCA) si il vérifie :*

1. *Les variables sont à valeurs binaires. Les équations structurelles s'écrivent donc comme des formules logiques.*
2. *Ces formules ne contiennent pas de disjonction.*
3. *Le graphe associé est acyclique.*

La première et la dernière hypothèses sont conformes aux choix classiques des modèles structurels causaux. En effet, le choix des variables binaires est le plus fréquent (Halpern, 2016; Miller, 2021). De plus, l'acyclicité du graphe causal, aussi appelée récursivité du modèle causal, est le cas d'étude principal considéré dans les travaux de Halpern. Selon lui, il y a, dans la majorité des cas, une notion de direction à la causalité impliquant l'anti-symétrie de la relation de dépendance causale, justifiant l'hypothèse d'acyclicité du graphe. Il propose néanmoins une approche dans le cas d'un graphe cyclique (Halpern, 2016). Cette extension se base sur le constat suivant : pour un contexte donné, c'est-à-dire une instantiation des variables exogènes fixées, l'existence et l'unicité des valeurs des variables endogènes ne sont plus garanties. Par conséquent, dire que $X = x$ est vrai si et seulement si x est l'unique solution des équations structurelles étant donné un modèle M et un contexte u n'est plus possible. Il ne faut donc plus considérer uniquement le contexte u mais aussi un ensemble de valeurs déterminé pour les variables endogènes. Sur cette base, Halpern propose une généralisation de la condition **AC2** pour ce nouveau cadre mais par laquelle "il n'est pas convaincu en raison du manque d'exemples convainquants dans lesquels la définition pour les modèles non récursifs est en accord avec notre intuition" (Halpern, 2016).

Enfin, la condition sur l'absence de disjonction est à première vue plus restrictive que les deux autres. Cependant, il s'agit d'une contrainte qui n'intervient que dans un seul sens de l'équivalence à savoir le passage des GCA aux AAF discuté dans la section 2.4.

Toutefois, cette transformation ne produit aucune disjonction. Or, l'objectif initial est la génération d'un modèle structurel causal à partir de la modélisation d'une interaction avec un AAF. De fait, restreindre le cadre permettant de transformer un GCA en AAF n'est pas contraignant pour l'application visée.

2.3.2 Transformation proposée $\mathcal{T}_{AFa2GCA}$

Pour transformer un graphe d'argumentation acyclique en graphe causal argumentatif, il faut établir une correspondance entre les nœuds et les arcs de chacun de ces graphes. Dans un GCA, un nœud est une variable endogène ou exogène binaire quelconque. En argumentation, un nœud représente un argument. De plus, pour chaque argument on peut ajouter une valeur binaire indiquant s'il est acceptable ou non. En utilisant ce statut d'acceptabilité comme variable causale, on obtient une correspondance entre les nœuds des graphes. En effet, à chaque argument a on associe une variable booléenne X_a telle que $X_a = 1$ est interprété comme "l'argument a est acceptable". Ces variables constituent l'ensemble des variables endogènes.

Concernant les arcs, en argumentation ils indiquent une relation d'attaque d'un argument vers un autre. Dans un modèle structurel causal, ils indiquent une dépendance causale de l'enfant envers ses parents. Cette dépendance est traduite par l'équation structurelle associée à l'enfant. Pour compléter la transformation, il faut donc définir une équation structurelle pour chaque argument, ce qui revient à traduire la relation d'attaque sous forme de dépendance causale entre l'acceptabilité des arguments, c'est-à-dire les variables endogènes. Dans le cas d'un graphe d'argumentation acyclique, une telle dépendance existe : un argument est acceptable si et seulement si il n'est pas attaqué ou que tous ses attaquants sont non acceptables.

Enfin, un modèle structurel causal permet de raisonner sur différents contextes. C'est une caractéristique essentielle de ce formalisme. Dans le cas de l'argumentation, des scénarios contrefactuels peuvent être associés à des graphes d'argumentation dans lesquels l'un ou l'autre des arguments n'est pas énoncé. Pour les modéliser, on crée une variable exogène $\tilde{X}_a$ pour chaque argument a , se traduisant par "l'argument a a été énoncé". Il s'agit dans ce cas d'une variable exogène car le fait qu'un argument soit énoncé au cours d'une interaction dépend de facteurs extérieurs au modèle, liés, par exemple, à celui qui a énoncé cet argument.

Formellement :

Définition 2.7 (Transformation d'un AAF en GCA – $\mathcal{T}_{AFa2GCA}$). *Soit $AF = (\mathcal{A}, \mathcal{R})$ et son graphe associé que l'on suppose acyclique. On pose :*

- $\mathcal{U} = \{\tilde{X}_a \mid a \in \mathcal{A}\},$
- $\mathcal{V} = \{X_a \mid a \in \mathcal{A}\},$
- $\mathcal{F} = \{F_{X_a} \mid X_a \in \mathcal{V}\}$ avec :
 - ◊ $\forall a \in \mathcal{A} \text{ tel que } Att_a = \emptyset, F_{X_a} = (X_a = \tilde{X}_a),$
 - ◊ $\forall a \in \mathcal{A} \text{ tel que } Att_a \neq \emptyset, F_{X_a} = \left(X_a = \tilde{X}_a \wedge \left(\bigwedge_{b \in Att_a} \neg X_b \right) \right).$

On note $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ le triplet ainsi obtenu, résultat de la transformation de AF par $\mathcal{T}_{AFa2GCA}$ notée $\mathcal{T}_{AFa2GCA}(AF)$.

Proposition 2.1. *Soit $AF = (\mathcal{A}, \mathcal{R})$ et son graphe associé que l'on suppose acyclique et $M = (\mathcal{U}, \mathcal{V}, \mathcal{F}) = \mathcal{T}_{AFa2GCA}(AF)$. Alors M est un GCA.*

Démonstration. Tout d'abord, M est un modèle structurel causal selon la définition 2.1. De plus, comme AF est acyclique alors, par construction de $\mathcal{F}$, M est acyclique. Enfin, toujours par construction, les équations structurelles de $\mathcal{F}$ n'utilisent pas de disjonctions. On a donc un modèle structurel causal acyclique n'utilisant que des variables binaires et dont les équations structurelles ne contiennent pas de disjonctions, c'est-à-dire que M est bien un GCA. $\square$

Avec cette définition, on introduit des informations supplémentaires à celles fournies par le graphe d'argumentation seul : les contextes. En effet, en argumentation un seul contexte est considéré, celui où tous les arguments du graphe ont été énoncés. Nous appelons cet unique contexte de l'argumentation le *contexte par défaut*, et on le note u^* . Formellement, il s'agit du contexte dans lequel toutes les variables exogènes valent 1 c'est-à-dire $u^* = \{(\tilde{X}_a = 1) \mid \tilde{X}_a \in \mathcal{U}\}$.

2.3.3 Exploitation des GCA pour la génération d'explications

L'objectif initial de cette transformation est de permettre de générer des explications causales du statut d'acceptabilité d'un argument. Ces deux formalismes possédant chacun leur propre définition de la notion d'explication, nous allons comparer la définition proposée par Halpern (2016) sur $\mathcal{T}_{AFa2GCA}(AF)$ à celle de Fan & Toni (2014). Pour rappel, en considérant $AF = (\mathcal{A}, \mathcal{R})$ un AAF et a un argument de $\mathcal{A}$ tel que $Att_a \neq \emptyset$, on dit qu'un ensemble d'argument S est admissible lié de sujet a si S est sans conflit et que tous les arguments attaqués de S sont défendus par au moins un autre argument de S . D'après Fan & Toni (2014), un ensemble admissible lié de sujet a , minimal au sens de l'inclusion est une explication compacte de l'acceptabilité de a .

Grâce à la transformation $\mathcal{T}_{AFa2GCA}$ que nous proposons, nous pouvons traduire cette définition sur un GCA et ainsi étudier sa compatibilité avec celle de Halpern (2016). Une preuve détaillée de la proposition suivante est proposée en annexe A, et seules les grandes lignes en sont données ici.

Proposition 2.2 (Explication compacte et modèle structurel). *Soit $AF = (\mathcal{A}, \mathcal{R})$ un système abstrait d'argumentation dont le graphe est acyclique.*

Soit $a^ \in \mathcal{A}$ tel qu'il existe un ensemble admissible dont il est le sujet. Soit S une explication compacte de a^* .*

Soit $M = \mathcal{T}_{AFa2GCA}(AF) = (\mathcal{U}, \mathcal{V}, \mathcal{F})$, le graphe causal argumentatif issu de la transformation $\mathcal{T}_{AFa2GCA}$ (définition 2.7).

On pose :

- $\varphi = (X_{a^*} = 1)$,
- $X_{arg} = S \setminus \{a^*\}$ et $\mathbf{X} = \{X_a \mid a \in X_{arg}\}$,
- $\mathcal{K}$ l'ensemble des contextes considérés comme possibles par l'explaine. On fait l'hypothèse que le contexte par défaut u^* appartient à $\mathcal{K}$.

*Alors $\mathbf{X} = 1$ est une explication, au sens causal, non minimale de φ relative à $\mathcal{K}$, c'est-à-dire $\mathbf{X} = 1$ vérifie **EX1** et **EX3** dans $\mathcal{K}$.*

Principe de la preuve. Pour vérifier **EX3**, on montre que le contexte u^* est compatible avec l'explication. Pour **EX1**, on vérifie **SC1** par construction, et **SC3** par contradiction. Enfin, pour **SC2**, on construit une cause effective de φ puis on montre qu'un élément de $\mathbf{X}$ est inclus dans cette cause effective vérifiant cette condition. $\square$

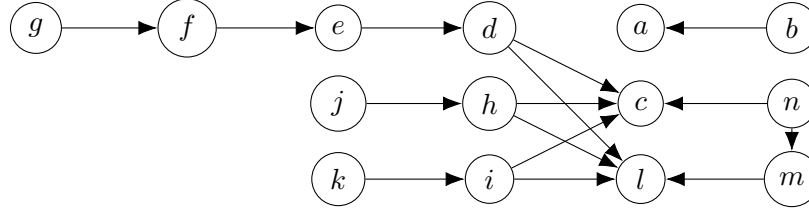


FIGURE 2.2 – Graphe d'argumentation associé à l'exemple 1 simplifié.

Dans cette proposition, nous avons réintroduit la notion de destinataire de l'explication. En effet, $\mathcal{K}$ représente l'ensemble des contextes considérés par l'*explainée*. Nous imposons seulement que u^* appartienne à $\mathcal{K}$. Cette hypothèse semble raisonnable car il s'agit de l'unique contexte considéré lorsque l'on travaille d'un point de vue purement argumentatif. Sous cette hypothèse, une explication argumentative au sens de Fan & Toni (2014) est une explication causale non minimale. L'absence de la minimalité vient du fait que l'explication argumentative renvoie tous les défenseurs intermédiaires d'un chemin causal. Avec la définition de cause suffisante, en particulier à cause du critère de minimalité, seul un sous-ensemble de ces variables peut suffire.

Exemple 1 (Examen médical simplifié). *Pour illustrer cette proposition, nous reprenons l'exemple 1 qui introduisait un débat entre un radiologue et un médecin sur le sujet d'un examen médical à faire passer à un bébé. Toutefois, la transformation $\mathcal{T}_{AFa2GCA}$ ne s'appliquant qu'à des graphes acycliques, on fait l'hypothèse que le fait que ce soit une urgence prime sur le fait qu'il n'y ait plus de disponibilité, c'est-à-dire que l'attaque de m sur n est supprimée. On obtient le nouveau graphe d'argumentation représenté sur la figure 2.2.*

Avec ce graphe, une explication compacte de l'acceptabilité de l , c'est-à-dire un ensemble admissible lié de sujet l et minimal au sens de l'inclusion, est donnée par l'ensemble suivant $S = \{n, k, j, e, g\}$. Après la transformation $\mathcal{T}_{AFa2GCA}$, nous avons donc l'explication suivante : $\{X_n = 1, X_k = 1, X_j = 1, X_e = 1, X_g = 1\}$. En utilisant la définition d'une explication présentée dans la section 2.2.4 à tous les contextes possibles, on obtient parmi les explications causales possibles, l'explication suivante $\{X_n = 1, X_k = 1, X_j = 1, X_g = 1\}$. On observe que l'explication obtenue par la définition de Halpern (2016) est incluse dans celle obtenue après transformation avec la définition de Fan & Toni (2014). En particulier, $X_e = 1$ fait partie de l'explication compacte, ce qui correspond bien au fait que tous les défenseurs le long d'un chemin causal, ici entre g et l , sont inclus contrairement à l'explication causale.

2.4 Passage des GCA aux AAF

Dans cette section, nous proposons une transformation réciproque à $\mathcal{T}_{AFa2GCA}$, ainsi qu'une preuve de l'équivalence entre ces deux cadres formels. Une preuve détaillée de ce théorème est proposée en annexe A.

2.4.1 Transformation proposée $\mathcal{T}_{GCA2AFa}$

Pour construire un graphe d'argumentation abstraite, il faut définir deux ensembles : les arguments et la relation d'attaque. Un des avantages de ce formalisme est la quasi-

absence de contraintes sur la nature d'un argument. Ainsi, n'importe quel ensemble fini d'éléments peut convenir à condition que cela ait un sens d'accepter ou de refuser ces éléments. Les variables d'un GCA étant des variables binaires, elles décrivent des propriétés qui peuvent être vraies ou fausses. Pour chaque variable endogène, on peut donc associer un argument qui correspond à la propriété en question. Si cet argument est accepté, la propriété est vraie. S'il est refusé, elle est fausse.

La relation d'attaque est une relation binaire sur l'ensemble des arguments, elle est, elle aussi, abstraite. Il n'y a donc pas de contraintes sur la nature de l'attaque. Toutefois, une des règles fondamentales de l'argumentation est qu'un argument est acceptable si et seulement s'il n'est pas attaqué ou qu'il n'est attaqué que par des arguments non acceptables. Il faut donc s'assurer lors de la construction de cette relation que, si la variable endogène associée à l'attaquant présumé est vraie dans le modèle structurel causal, alors l'argument attaqué est faux dans ce modèle. Pour cela, quel que soit le contexte, on peut étudier le scénario contrefactuel dans lequel toutes les variables endogènes sauf l'attaquant et l'attaqué présumé sont fausses, donc tous les arguments sont refusés. Si, lorsque l'attaquant testé est vrai, l'attaqué potentiel est faux, alors la relation d'attaque est validée.

Formellement :

Définition 2.8 (Transformation d'un GCA en AAF – $\mathcal{T}_{GCA2AFa}$). *Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ un graphe causal argumentatif.*

Posons :

- $\mathcal{A}' = \{a_X \mid X \in \mathcal{V}\}$,
- $\mathcal{R}' = \{(b, a) \in \mathcal{A}' \times \mathcal{A}' \mid \forall \mathbf{u}, (M, \mathbf{u}) \models [X_b = 1, \mathbf{Y} = \mathbf{0}](X_a = 0) \text{ avec } \mathbf{Y} = \mathcal{V} \setminus \{X_a, X_b\}\}$

On note $AF' = \mathcal{T}_{GCA2AFa}(M) = (\mathcal{A}', \mathcal{R}')$ la transformation de M par $\mathcal{T}_{GCA2AFa}$.

Proposition 2.3. *Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ un graphe causal argumentatif et $AF' = \mathcal{T}_{GCA2AFa}(M)$. Alors, AF' est un système d'argumentation abstraite.*

Démonstration. Par construction de AF' , $\mathcal{A}'$ est un ensemble fini d'éléments formant l'ensemble des arguments. De plus, une relation binaire sur cet ensemble est définie. Pour obtenir un AAF, il faut vérifier que la sémantique associée à cette relation binaire est la même que celle de la relation d'attaque. Or par construction de $\mathcal{R}'$, un argument a est acceptable si et seulement s'il n'est pas attaqué ou qu'il n'est attaqué que par des arguments non acceptables. Donc $\mathcal{R}'$ est bien une relation d'attaque. $\square$

2.4.2 Théorème d'équivalence entre AAF et GCA

Proposition 2.4 (Équivalence entre AAF et GCA). *Soit $AF = (\mathcal{A}, \mathcal{R})$ un système d'argumentation abstraite acyclique.*

Alors :

$$AF = \mathcal{T}_{GCA2AFa}(\mathcal{T}_{AFa2GCA}(AF)).$$

Principe de la preuve. On note $(\mathcal{A}', \mathcal{R}') = \mathcal{T}_{GCA2AFa}(\mathcal{T}_{AFa2GCA}(AF))$. Par construction, on a $\mathcal{A} = \mathcal{A}'$. Il faut donc montrer que $\mathcal{R} = \mathcal{R}'$. Cette preuve est faite de façon directe par double inclusion. $\square$

Cette proposition fait le lien entre les deux cadres formels que nous étudions : les graphes d’argumentation abstraite et les graphes causaux argumentatifs. La raison principale pour cela est l’équivalence entre la notion d’attaque et la dépendance causale de l’acceptabilité d’un argument vis-à-vis de ses attaquants.

Remarque – Nous avons montré dans la section précédente qu’une explication compacte (Fan & Toni, 2014) est une explication causale non minimale. Le résultat réciproque est beaucoup moins direct. En effet, en argumentation abstraite, l’objectif n’est pas de modéliser le destinataire de l’explication. Il s’agit plutôt d’une retranscription d’un échange d’arguments entre plusieurs entités. L’explication sert alors à justifier pourquoi un argument peut être accepté en renvoyant les différents arguments qui sont intervenus pour défendre ce dernier. En particulier, il n’est pas question de contexte. De fait, la transformation $\mathcal{T}_{GCA2AFa}$ ne peut pas être appliquée directement pour traduire la définition d’une explication causale dans le formalisme des AAF. Pour transformer une explication causale, au moins non minimale, il faudrait pouvoir retranscrire la connaissance que possède l’*explaine*e sur l’univers d’une autre façon puis établir le lien avec la notion de contexte. Nous n’avons cependant pas étudié cette question car cette définition d’explications proposée dans le cadre argumentatif propose moins d’outils que celle dans le cadre des modèles structurels causaux pour la génération d’explications adaptées à l’*explaine*e. Par conséquent, la question de l’équivalence sur les définitions d’explications reste ouverte.

2.4.3 Équivalence et inverse à gauche

Formellement, la proposition 2.4 établit qu’étant donné un AAF, nous retrouvons l’AAF de départ en appliquant successivement les deux transformations. Elle montre donc que $\mathcal{T}_{GCA2AFa}$ est une inverse à gauche de $\mathcal{T}_{AFa2GCA}$. Pour obtenir une équivalence, il faudrait également démontrer que $\mathcal{T}_{GCA2AFa}$ est une inverse à droite de $\mathcal{T}_{AFa2GCA}$. Dans le cadre de cette thèse, le point de départ est toujours un AAF et donc le résultat d’inverse à gauche est suffisant. Pour cette raison, nous ne donnons pas la preuve formelle pour l’inverse à droite mais en donnons seulement les éléments principaux.

La transformation $\mathcal{T}_{GCA2AFa}$ ne fait pas intervenir les variables exogènes. Par conséquent, il est possible d’envisager un GCA possédant une unique variable exogène ou de très nombreuses variables exogènes pour chaque argument. Dans les deux cas, en appliquant la transformation $\mathcal{T}_{AFa2GCA}$, on retrouverait exactement le même modèle structurel causal, différent de celui de départ. Pour y remédier, on pourrait vouloir ajouter une condition dans la définition d’un GCA : pour chaque variable endogène, il existe une unique variable exogène et celle-ci n’intervient que dans l’équation structurelle de “sa” variable endogène associée. Cela permettrait alors d’identifier cette variable comme l’énonciation ou non d’un argument comme dans la transformation $\mathcal{T}_{AFa2GCA}$.

Toutefois, il s’agit d’une hypothèse très forte et réductrice et qui n’est pas nécessaire car les conditions présentes dans la définition d’un GCA suffisent. En effet, les variables exogènes permettent de représenter des facteurs “extérieurs” au modèle dont on ne connaît pas l’origine. Pour cette raison, leur valeur est fournie sous la forme d’un contexte. De plus, dans le cadre d’un GCA, les équations structurelles ne contiennent pas de disjonctions. Dans ce cas, toutes les variables exogènes jouent un rôle identique dans

les équations structurelles et leur contribution peut s'écrire sous la forme $(\bigwedge U_i^V)$, avec U_i^V les variables exogènes d'une variable endogène V . En notant $U^V = (\bigwedge U_i^V)$, nous obtenons, sans perte d'informations pour les variables endogènes, une unique variable exogène pour chaque variable endogène correspondant à l'énonciation d'un argument et qui n'intervient que dans l'équation structurelle de cette dernière. À l'aide de cette transformation, et sans ajouter d'hypothèse supplémentaire, il est possible d'obtenir la propriété d'inverse à droite et par conséquent l'équivalence entre ces deux cadres formels.

2.5 Conclusion

Dans ce chapitre, nous avons proposé une méthode permettant de générer des explications causales en argumentation abstraite, sous hypothèse d'acyclicité du graphe d'argumentation. Pour cela, nous avons construit une transformation permettant de représenter un système d'argumentation abstraite acyclique sous la forme d'un modèle structurel causal particulier appelé graphe causal argumentatif. Cette transformation offre deux avantages majeurs. Tout d'abord, elle répond à la difficulté d'établir un graphe causal pour un problème donné en utilisant les outils développés en argumentation. Ensuite, elle offre au domaine de l'argumentation abstraite un cadre permettant d'effectuer des raisonnements causaux complexes et propice à la génération d'explications causales. Nous avons également comparé la définition d'explications proposée par Fan & Toni (2014) à celle de Halpern (2016). Nous avons en particulier établi qu'une explication argumentative au sens de Fan et Toni était une explication causale non minimale au sens de Halpern. Enfin, nous avons proposé une transformation réciproque et établi une équivalence entre les AAF et les GCA.

Les systèmes d'argumentation abstraite étant un cadre statique, nous proposons dans les chapitres 3 et 4 de les enrichir en incluant l'ordre d'énonciation des arguments dans le modèle.

Intégration de la temporalité en argumentation abstraite : l'apport des langages d'action

Ce chapitre aborde la question de l'ajout de la temporalité dans les graphes d'argumentation abstraite acycliques. En particulier, nous proposons une réécriture des graphes d'argumentation abstraite acycliques dans un langage d'action. De cette façon, nous modélisons l'évolution d'un AAF lorsque des arguments sont énoncés et établissons les relations causales entre l'énonciation de ces arguments et leurs conséquences directes comme indirectes. Enfin, nous discutons de la génération d'explications adaptées au destinataire avec ce nouveau formalisme. Ces travaux ont été menés en collaboration étroite avec Camilo Sarmiento qui s'intéresse à la formalisation des raisonnements éthiques et en particulier à la modélisation, représentation et automatisation du raisonnement causal.

Cela a fait l'objet de trois publications :

- *dans la conférence française JIAF (Munro et al., 2023b)*
- *dans le workshop international XLoKR (Munro et al., 2023a)*
- *dans le journal international AI Communications (Munro et al., 2025).*

Sommaire

3.1 Introduction	46
3.2 Préliminaires sur les langages d'action	46
3.2.1 Définition	46
3.2.2 Causalité effective	50
3.3 Passage des AAF acycliques au langage d'action	52
3.3.1 Instanciation du contexte κ	53
3.3.2 Modification de la sémantique	57
3.3.3 Implémentation en Answer Set Programming (ASP)	58
3.4 Étude des propriétés formelles de la transformation $\mathcal{T}_{AFa2AL}$	59
3.4.1 Unicité des traces	59
3.4.2 Terminaison de $\mathcal{T}_{AFa2AL}$	60

3.4.3	Correction et complétude de $\mathcal{T}_{AFa2AL}$	61
3.5	Discussion sur la génération d'explications	62
3.5.1	Représentation graphique comme support de l'explication	62
3.5.2	Construction de la chaîne causale	65
3.5.3	Vers l'extraction d'explications adaptées à partir d'une chaîne causale	67
3.6	Conclusion	68

3.1 Introduction

Dans un dialogue, la notion d'ordre est essentielle. En effet, les arguments sont énoncés les uns à la suite des autres selon un ordre temporel précis. Afin de modéliser son évolution, puis d'expliquer l'acceptabilité des arguments, il faut prendre en compte cette temporalité dans le cadre formel choisi. Pour cela, nous proposons d'utiliser un système de transition d'états étiqueté, et en particulier un langage d'action qui enregistre par l'intermédiaire de deux *traces* toutes les étapes par lequel le dialogue est passé.

Dans la section 3.2, nous introduisons le langage d'action que nous utilisons (Sarmiento *et al.*, 2022) ainsi que la définition de causalité effective qui lui est associée. Nous proposons ensuite dans la section 3.3, une transformation, que nous appelons $\mathcal{T}_{AFa2AL}$, permettant d'enrichir un système d'argumentation abstraite acyclique avec l'ordre d'énonciation des arguments. Nous établissons, dans la section 3.4, la terminaison, la correction et la complétude de cette transformation. Enfin, dans la section 3.5, nous discutons de la génération d'explications causales pour l'acceptabilité d'un argument dans le langage d'action.

3.2 Préliminaires sur les langages d'action

Un langage d'action est un formalisme logique permettant de modéliser l'évolution du monde étant donné un ensemble d'actions réalisées délibérément par des agents (Gelfond & Lifschitz, 1998). Plus précisément, d'une part l'état du monde est défini comme une collection de variables le décrivant. D'autre part, les transitions entre les états du monde sont le résultat de l'occurrence d'un ensemble d'évènements. L'évolution du monde peut ainsi être vue comme une suite d'états qui s'enchaînent les uns après les autres au fur et à mesure que des évènements se produisent.

Dans cette section, nous présentons le langage d'action tel que défini par Sarmiento *et al.* (2022). Nous introduisons ensuite brièvement la définition formelle de causalité effective associée, fondée sur une approche par régularité (voir section 1.2.1), qui a motivé le choix de ce langage d'action.

3.2.1 Définition

Formellement, le langage d'action que nous utilisons (Sarmiento *et al.*, 2022) s'appuie sur une décomposition du monde en deux ensembles $\mathbb{F}$ et $\mathbb{E}$:

- L'ensemble $\mathbb{F}$ comporte les variables décrivant l'état du monde et ses propriétés, pouvant varier dans le temps. Elles sont appelées des *fluents*.

- L'ensemble $\mathbb{E}$ contient les variables décrivant les transitions dont l'occurrence modifie les fluents. Ces variables sont appelées des *événements*.

Un littéral de fluent est soit un fluent $f \in \mathbb{F}$, soit sa négation $\neg f$. L'ensemble des littéraux de fluents dans $\mathbb{F}$ est noté $Lit_{\mathbb{F}}$ et il est formellement défini par $Lit_{\mathbb{F}} = \mathbb{F} \cup \{\neg f \mid f \in \mathbb{F}\}$. Le complément d'un littéral de fluent l est défini comme $\bar{l} = \neg f$ si $l = f$, ou $\bar{l} = f$ si $l = \neg f$.

Exemple 7 (Circuit électrique). (*modélisation dans un langage d'action* –) Reprenons l'exemple du circuit électrique présenté dans le chapitre 2. Le monde en question est constitué de deux sources d'énergie A et B , d'un commutateur électrique S et d'une lampe L . Chaque source d'énergie pouvant être éteinte ou allumée, on crée un premier fluent o_A (respectivement o_B) qui signifie que la source A (respectivement B) est allumée. De la même façon pour la lampe, on crée le fluent o_L . Enfin, le commutateur est soit à gauche soit à droite. On ajoute donc les deux fluents g_S et d_S qui indiquent que le commutateur est à gauche (respectivement à droite). On a donc $\mathbb{F} = \{o_A, o_B, o_L, g_S, d_S\}$. L'ensemble des événements $\mathbb{E}$ est décrit dans la suite.

Définition 3.1 (État S). Un ensemble $L \subseteq Lit_{\mathbb{F}}$ est un état si :

- il est cohérent : $\forall l \in L, \bar{l} \notin L$;
- il est complet : $\forall f \in \mathbb{F}, f \in L$ ou $\neg f \in L$.

Un état est donc un ensemble $L \subseteq Lit_{\mathbb{F}}$ donnant la valeur de chaque fluent décrivant le monde. Le temps est modélisé de façon linéaire de sorte à obtenir un état $S(t)$ pour chaque pas de temps t de l'ensemble $\mathbb{T} = \{-1, 0, \dots, N\}$, avec $S(0)$ l'état initial. Il s'agit d'une formalisation bornée dans le passé d'un problème réel qui lui n'est pas borné. Pour avoir la formalisation la plus fidèle possible, tous les états précédant $t = 0$ sont recueillis dans un état $S(-1) = Lit_{\mathbb{F}} \setminus S(0)$. Cela permet d'assurer que chaque fluent possède au moins une cause effective.

Exemple 7 (Circuit électrique). (*suite*) – La situation initiale décrite est celle où les deux sources sont allumées o_A, o_B et le commutateur électrique est à droite $\neg g_S$ et d_S . Enfin, la lampe est également allumée o_L . On a donc $S(0) = \{o_A, o_B, \neg g_S, d_S, o_L\}$.

Un événement $e \in \mathbb{E}$ est une formule atomique caractérisée par trois composantes : des *préconditions* indiquant les conditions devant être satisfaites par l'état S pour que l'événement puisse se déclencher ; des *conditions de déclenchement* donnant toutes les conditions devant être satisfaites au temps t pour que l'événement puisse se déclencher, conditions dont la singularité par rapport aux préconditions est détaillée ci-dessous ; des *effets* précisant les changements de l'état du monde attendus si l'événement se produit. On utilise ici le mot "attendu" car un événement peut avoir moins d'effets que ceux formalisés lorsqu'il se produit dans certains contextes. Nous revenons sur ce point dans la définition 3.3.

Les préconditions et les effets sont respectivement représentés par des formules des langages $\mathcal{P} ::= l|\psi_1 \wedge \psi_2| \psi_1 \vee \psi_2$ et $\mathcal{E} ::= l|\varphi_1 \wedge \varphi_2$ où l est un symbole terminal et (φ, ψ) sont non terminaux. Les fonctions associant à chaque événement préconditions, conditions de déclenchement et effets sont respectivement notées *pre*, *tri* et *eff*, et sont définies comme : $pre : \mathbb{E} \rightarrow \mathcal{P}$, $tri : \mathbb{E} \rightarrow \mathcal{P}$, et $eff : \mathbb{E} \rightarrow \mathcal{E}$. $\mathcal{E}$ est décomposé en deux ensembles disjoints $\mathbb{A}$ et $\mathbb{U}$ qui en forment :

- $\mathbb{A}$ contient les actions réalisées par des agents et donc soumises à leur volition ;

- $\mathbb{U}$ contient les événements dits exogènes qui se déclenchent aussitôt que leurs préconditions *pre* sont satisfaites, sans qu'un agent n'ait besoin de les réaliser.

Pour les événements exogènes, il n'y a pas de différence entre *pre* et *tri*. À l'opposé, les conditions de déclenchement des actions ne se limitent pas aux préconditions, il faut en plus la volonté de réaliser l'action de la part de l'agent, ou une sorte de manipulation d'un agent tiers qui s'y substituerait.

Exemple 7 (Circuit électrique). *(suite) – Pour faire évoluer le monde, on imagine qu'un agent peut décider d'activer ou désactiver les sources A et B et également mettre le commutateur en position gauche ou droite. Cela conduit donc aux six actions suivantes :*

$$\mathbb{A} = \{\text{activer}_*, \text{désactiver}_*, \text{positionGauche}, \text{positionDroite}\} \text{ avec } * \in \{A, B\} :$$

- $\text{pre}(\text{activer}_*) \equiv \neg o_*$; $\text{eff}(\text{activer}_*) \equiv o_*$
- $\text{pre}(\text{désactiver}_*) \equiv o_*$; $\text{eff}(\text{désactiver}_*) \equiv \neg o_*$
- $\text{pre}(\text{positionGauche}) \equiv \neg g_S$; $\text{eff}(\text{positionGauche}) \equiv \neg d_S \wedge g_S$
- $\text{pre}(\text{positionDroite}) \equiv \neg d_S$; $\text{eff}(\text{positionDroite}) \equiv \neg g_S \wedge d_S$

Par ailleurs, chacune de ces actions peut avoir un impact sur l'état de la lampe en l'éteignant ou en l'allumant. Pour rappel, la lampe est allumée quand la source A est activée et que le commutateur est à gauche ou que la source B est activée et que le commutateur est à droite. Sinon, la lampe est éteinte. Cela revient à créer les deux événements exogènes suivants :

$$\mathbb{U} = \{\text{allumerLampe}, \text{éteindreLampe}\} :$$

- $\text{tri}(\text{allumerLampe}) \equiv \neg o_L \wedge ((o_A \wedge g_S) \vee (o_B \wedge d_S))$; $\text{eff}(\text{allumerLampe}) \equiv o_L$
- $\text{tri}(\text{éteindreLampe}) \equiv o_L \wedge ((\neg o_A \vee \neg g_S) \wedge (\neg o_B \vee \neg d_S))$; $\text{eff}(\text{éteindreLampe}) \equiv \neg o_L$

Ces événements ne sont pas soumis à la volition de l'agent, ils se produisent automatiquement quand leurs conditions de déclenchement sont vérifiées.

L'ensemble contenant tous les événements se produisant au pas de temps t est noté $E(t)$. Avec ces notations, ce langage d'action peut être résumé comme étant un système de transition classique, où $E(t)$ génère la transition entre les états $S(t)$ et $S(t+1)$. Ainsi, les états s'enchaînent au fur et à mesure que les événements se produisent, simulant l'évolution du monde.

Exemple 7 (Circuit électrique). *(suite) –*

Reprenons l'état du monde initial décrit par $S(0) = \{o_A, o_B, \neg g_S, d_S, o_L\}$. Imaginons maintenant que l'agent décide de désactiver la source A et de passer le commutateur à gauche. Il peut réaliser ces actions car l'état $S(0)$ vérifie les préconditions pour les réaliser. Formellement, on note $S(0) \models \text{pre}(\text{désactiver}_A)$ et $S(0) \models \text{pre}(\text{positionGauche})$. On obtient donc que l'ensemble des événements qui se produisent au temps $t = 0$ est : $E(0) = \{\text{désactiver}_A, \text{positionGauche}\}$. En appliquant les effets de ces actions, on arrive à l'état suivant $S(1) = \{\neg o_A, o_B, g_S, \neg d_S, o_L\}$.

À ce moment, même si l'agent ne réalise plus aucune action, comme le commutateur est à gauche et que la source A est désactivée alors l'état actuel $S(1)$ vérifie les conditions pour que la lampe s'éteigne, c'est-à-dire $S(1) \models \text{tri}(\text{éteindreLampe})$. On a donc que $E(1) = \{\text{éteindreLampe}\}$ et on passe à l'état $S(2) = \{\neg o_A, o_B, g_S, \neg d_S, \neg o_L\}$.

Un des avantages de ce langage d'action est qu'il autorise la concurrence d'évènements (plus d'un évènement peut avoir lieu à chaque pas de temps). Néanmoins, cela peut être la cause de conflits potentiels entre ces évènements. Pour les résoudre, un ordre partiel strict $\succ_E$ est introduit, qui garantit la priorité de déclenchement d'un évènement par rapport à un autre.

Définition 3.2 (Contexte κ). *Le contexte noté κ est l'octuple $(\mathbb{E}, \mathbb{F}, pre, tri, eff, S(0), \succ_E, \mathbb{T})$, où $\mathbb{E}, \mathbb{F}, pre, tri, eff, S(0), \succ_E$, et $\mathbb{T}$ ont été définis précédemment.*

Afin d'être en mesure d'obtenir des relations causales en accord avec la conception communément admise en philosophie, et cela malgré le fait d'avoir une formalisation bornée dans le passé, il est nécessaire que les évènements s'étant produits avant $t = 0$ soient représentés. Pour chaque littéral $l \in S(0)$ on introduit un évènement $ini_l \in \mathbb{E}$ tel que $eff(ini_l) = l$. On note alors $E(-1) = \{ini_l, l \in S(0)\}$ qui vérifie $eff(E(-1)) = S(0)$.

Comme nous l'avons mentionné lors de leur introduction, les évènements peuvent avoir moins d'effets que ceux attendus. Or, pour raisonner sur les causes effectives ayant conduit à un certain état du monde, il est important de déterminer les *effets effectifs* ou effets réels d'un évènement et non pas seulement ces effets attendus.

Définition 3.3 (Effets effectifs $actualEff(E, L)$). *Étant donné un état $L \subseteq Lit_{\mathbb{F}}$ et un ensemble d'évènements $E \subseteq \mathbb{E}$, le prédicat $actualEff(E, L)$ associe à un couple (E, L) un état défini comme :*

$$actualEff(E, L) = \{l \in Lit_{\mathbb{F}} \mid \exists e \in E, l \in eff(e) \text{ et } l \notin L\}$$

Ce prédicat permet, étant donné un état, de déterminer l'état suivant si un ensemble d'évènement se produisait. Il fournit donc déjà des informations causales simples : quelles conséquences un ensemble d'évènements E qui se produisent à l'état L a eues parmi ses effets intrinsèques ou attendus. De telles conséquences sont appelées des effets effectifs. Grâce à cette définition, il est maintenant possible de définir un opérateur de mise à jour :

Définition 3.4 (Opérateur de mise à jour $\triangleright$). *Étant donné un contexte κ , un état $S(t)$ et un ensemble d'évènements $E \in \mathbb{E}$, l'opérateur $\triangleright$ renvoie le nouvel état résultant des effets effectifs des évènements :*

$$S(t) \triangleright E = S(t) \setminus \overline{actualEff(E, S(t))} \cup actualEff(E, S(t)).$$

Cet opérateur définit les transitions d'états qui se produisent sous l'effet d'un ensemble d'évènements. Il permet donc de représenter l'évolution du monde comme une succession de mises à jour de l'état $S(0)$ jusqu'à l'état final $S(N+1)$ à cause des évènements $E(0) \dots E(N)$. Une telle séquence est appelée une exécution.

Définition 3.5 (Exécution valide). *Une exécution est une séquence alternant entre ensemble d'évènements et état du monde $E(-1), S(0), E(0), \dots, E(N), S(N+1)$.*

Elle est dite valide étant donné un contexte κ si elle vérifie, $\forall t \in \mathbb{T}$:

1. $S(t) \subseteq Lit_{\mathbb{F}}$ est un état au sens de la définition 3.1.
2. $E(t) \subseteq \mathbb{E}$ vérifie :
 - (a) $\forall e \in E(t), S(t) \models pre(e)$;

- (b) $\nexists (e, e') \in E(t)^2, e \succ_E e'$;
 - (c) $\forall e \in E \text{ tel que } S(t) \models tri(e), e \in E(t) \text{ ou } \exists e' \in E(t), e' \succ_E e$.
3. $S(t+1) = S(t) \triangleright E(t)$.

Pour un contexte κ donné, il existe potentiellement plus d'une exécution valide. En effet, aucune spécification du moment où les actions sont réalisées n'est incluse dans le contexte. Ainsi, leurs préconditions peuvent être satisfaites, et donc des exécutions peuvent être valides. Cependant, la volition de l'agent est manquante et donc leurs conditions de déclenchement (*tri*) ne sont pas satisfaites. Il faut donc rajouter une deuxième entrée au langage en plus du contexte κ , appelée *le scénario*.

Définition 3.6 (Scénario σ). *Un scénario, noté σ , est un ensemble fini d'actions couplées à des dates : $\sigma \subseteq \mathbb{A} \times \mathbb{T}$.*

Un tel *scénario* représente la volition des agents et permet d'obtenir une unique exécution valide. D'une telle exécution il est possible d'extraire deux types de traces.

Définition 3.7 (Traces $\tau_{\sigma, \kappa}^e$ et $\tau_{\sigma, \kappa}^s$). *Étant donné un scénario σ et un contexte κ , la trace d'évènements $\tau_{\sigma, \kappa}^e$ de σ, κ est la séquence d'évènements $E(-1), E(0), \dots, E(N)$ contenue dans une des exécutions valides étant donné κ , telle que :*

$$\forall t, \forall e \in E(t), e \in \mathbb{A} \Leftrightarrow (e, t) \in \sigma.$$

La trace d'états $\tau_{\sigma, \kappa}^s$ est la séquence d'états $S(0), S(1), \dots, S(N+1)$ correspondant à $\tau_{\sigma, \kappa}^e$.

Ces deux traces constituent la sortie du langage d'action et fournissent toute l'histoire du monde avec la succession d'états $\tau_{\sigma, \kappa}^s$ et les évènements qui se sont produits à chaque instant $\tau_{\sigma, \kappa}^e$. Elles forment la brique de base pour le raisonnement causal permis par ce langage d'action.

3.2.2 Causalité effective

La définition de causalité effective proposée par Sarmiento *et al.* (2022) est une formalisation du test *NESS* adaptée au langage d'action présenté dans la section précédente. Ce formalisme représente le monde comme une succession d'états produits par des occurrences d'évènements. Il introduit donc des états entre les évènements. Pour cette raison, en plus de la relation de causalité effective qui relie deux occurrences d'évènements entre elles, comme communément accepté par les philosophes, il est nécessaire de définir des relations causales entre les actions et les états du monde, c'est-à-dire des relations où les causes sont des occurrences d'évènements, et les effets sont la véracité de formules du langage $\mathcal{P}$ à un temps donné. C'est pour cette étape que le test *NESS* est utilisé pour définir ces relations intermédiaires.

Sarmiento *et al.* (2022) introduisent trois relations causales qui permettent de déterminer les causes de la véracité d'une formule (pour les détails voir Sarmiento *et al.* (2022)). Elles sont illustrées graphiquement sur la figure 3.1 :

- (i) Les **NESS-causes directes** donnent des informations basiques sur les effets que l'occurrence d'un évènement a réellement eus, qui à nouveau ne sont pas nécessairement les mêmes que ceux qui sont attendus. Comme mentionné précédemment, il s'agit donc d'une relation entre l'occurrence d'un évènement et la véracité

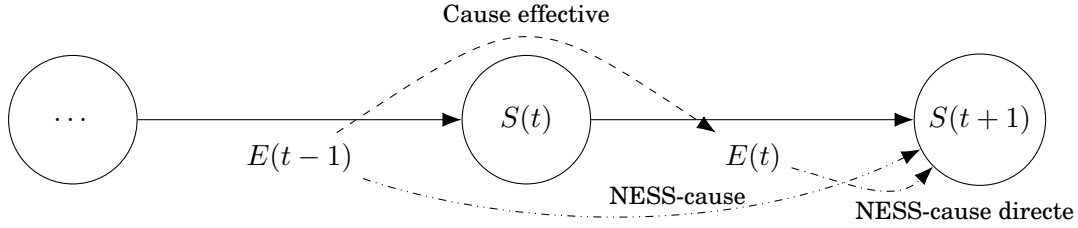


FIGURE 3.1 – Différentes relations causales

de formules du langage $\mathcal{P}$. Malgré leur aspect indispensable, ces relations ne sont pas toujours les plus intéressantes. En effet, l'ensemble des NESS-causes directes d'une formule peut contenir un certain nombre d'événements exogènes qui ne sont pas nécessairement pertinents. Il est souhaitable de remonter à des actions, c'est-à-dire d'établir une chaîne causale en remontant le temps de sorte à retrouver l'ensemble d'actions qui ont conduit à la véracité de la formule de $\mathcal{P}$.

- (ii) Les **NESS-causes** permettent de retrouver cette chaîne causale. En notant $\psi \in \mathcal{P}$ la formule considérée à l'instant t_ψ et C l'ensemble des NESS-causes directes de (ψ, t_ψ) , la NESS-cause identifie ce qui a causé $(tri(C), t)$, où $t < t_\psi$ nécessairement. Il faut noter que par définition les NESS-causes directes sont un type particulier de NESS-causes.
- (iii) Enfin, l'occurrence d'un premier événement est considéré comme une **cause effective** de l'occurrence d'un second d'événement si l'occurrence du premier est une NESS-cause des conditions de déclenchement du deuxième. On peut en déduire que, si l'occurrence (e', t_2) est une NESS-cause directe de (ψ, t_3) et que l'occurrence (e, t_1) est une cause effective de (e', t_2) avec $t_1 < t_2 < t_3$, alors l'occurrence (e, t_1) est une NESS-cause de (ψ, t_3) .

Des définitions formelles de ces trois relations ont été proposées par Sarmiento *et al.* (2023). En pratique, pour déterminer si un événement est une NESS-cause, on peut utiliser une caractérisation plus simple (Sarmiento *et al.*, 2023), donnée par la proposition suivante.

Proposition 3.1 (NESS-cause directe d'un littéral). *Étant donné un couple $\chi = (\sigma, \kappa)$ appelé configuration causale, des instants $t, t_l \in \mathbb{T}$, un événement $e \in E^\chi(t)$, et un littéral de fluent $l \in Lit_{\mathbb{F}}$ vrai au temps t_l , (e, t) est une NESS-cause directe de (l, t_l) si et seulement si :*

- (c₁) $(S^\chi(t) \triangleright \{e\}) \models l$;
- (c₂) $\forall t', t < t' \leq t_l, S^\chi(t') \models l$;
- (c₃) $S^\chi(t) \not\models l$.

Avec cette proposition, une NESS-cause directe d'un littéral de fluent l est un événement se produisant à la date t tel que, dans tous les états suivants jusqu'à la date t_l , l est vrai dans ces états alors qu'à la date t ce n'était pas le cas. Cet événement est donc l'élément qui a causé que l soit vrai à la date t_l .

Proposition 3.2 (NESS-cause directe d'une conjonction de littéraux de fluent). *Étant donné une configuration causale $\chi = (\sigma, \kappa)$, des instants $t, t_l \in \mathbb{T}$, un événement $e \in E^\chi(t)$*

et une formule $\psi = l_1 \wedge \dots \wedge l_m$ de $\mathcal{P}$ vraie au temps t_ψ , (e, t) est une NESS-cause directe de (ψ, t_ψ) si et seulement si $\exists j \in \{1, \dots, m\}$ tel que :

- (c₁) $(S^x(t) \triangleright \{e\}) \models l_j$;
- (c₂) $\forall t', t < t' \leq t_\psi, S^x(t') \models l_j$;
- (c₃) $S^x(t) \not\models l_j$.

De la même façon pour une conjonction de littéraux de fluent, si un évènement e est une NESS-cause d'un des littéraux de la formule ψ , alors c'est une NESS-cause directe de cette formule.

Exemple 7 (Circuit électrique). (suite–) Reprenons le scénario durant lequel un agent a décidé d'éteindre la source A et de passer le commutateur à gauche, et cherchons les causes qui ont conduit à ce que la lampe soit éteinte dans l'état $S(2)$. Une NESS-cause directe de $\neg o_L$ est que l'évènement éteindreLampe s'est déclenché. Comme nous l'avons dit, il ne s'agit pas d'une information très intéressante pour comprendre ce qu'il s'est passé. Nous allons donc chercher les évènements qui ont conduit au déclenchement de l'évènement éteindreLampe, c'est-à-dire les causes effectives de $(\text{éteindreLampe}, t = 1)$. Pour cela, il faut regarder les préconditions de cet évènement et en déterminer les NESS-causes.

On a $\text{tri}(\text{éteindreLampe}) \equiv o_L \wedge ((\neg o_A \vee \neg g_S) \wedge (\neg o_B \vee \neg d_S))$. Comme $S(-1) \models \neg o_L$ (c₃), $(S(-1) \triangleright \{\text{ini}_{o_L}\}) \models o_L$ (c₁) et que $\forall t', -1 < t' \leq 1, S(t') \models o_L$ (c₂), alors $(\text{ini}_{o_L}, t = -1)$ est une NESS-cause de $(\text{tri}(\text{éteindreLampe}), t = 1)$ d'après la proposition précédente. Il s'agit donc d'une cause effective de $(\text{éteindreLampe}, t = 1)$. Plus simplement, le fait que la lampe a commencé en position allumée est une cause effective du fait que la lampe a pu être éteinte. On remarque également avec cet exemple l'intérêt d'introduire l'ensemble d'évènements $E(-1)$ pour modéliser le passé et ainsi garantir l'existence d'au moins une cause.

Avec ce raisonnement, on trouve aussi que $(\text{désactiver}_A, t = 0)$ et $(\text{positionGauche}, t = 0)$ sont des causes effectives de $(\text{éteindreLampe}, t = 1)$. Enfin, par transitivité, le fait que la source A a été désactivée à l'instant $t = 0$ et que le commutateur a été placé à gauche à ce même instant sont des NESS-causes du fait que la lampe est éteinte à l'instant $t = 2$.

3.3 Passage des AAF acycliques au langage d'action

Dans cette section, nous présentons la contribution principale de ce chapitre, à savoir une réécriture d'un graphe d'argumentation abstraite acyclique dans le langage d'action présenté dans la section précédente. Nous notons cette transformation $\mathcal{T}_{\text{AAF} \rightarrow \text{2AL}}$. L'objectif ici est double : modéliser l'ordre d'énonciation des arguments d'une part, et permettre la recherche des causes de l'acceptabilité d'un argument à certaines dates d'autre part. Pour cela, la section 3.3.1 présente la transposition de la notion de contexte au sens des langages d'action, comme défini dans la définition 3.2, au cadre des AAF. La section 3.3.2 fournit les définitions modifiées de la sémantique associée au langage d'action, la section 3.3.3 décrit brièvement l'implémentation ASP que nous avons réalisée. Enfin, la section 3.4 établit les propriétés de la transformation proposée.

Tout d'abord, contrairement aux AAF classiques, nous proposons de prendre en compte l'ordre d'énonciation des arguments. Pour cette raison, au lieu de seulement considérer un graphe d'argumentation abstraite $(\mathcal{A}, \mathcal{R})$, nous considérons un couple $(\Delta, \mathcal{R})$, où Δ

est un dialogue, c'est-à-dire une séquence d'arguments abstraits comme par exemple des énoncés en langage naturel. Formellement :

Définition 3.8 (Dialogue Δ). *Un dialogue Δ est défini par $\Delta \subseteq \mathcal{A} \times \mathbb{N}$, où chaque argument a est associé à son ordre d'énonciation o .*

Il est important de noter que nos travaux n'ont pas pour objectif de réguler un dialogue. Ainsi, à un instant donné, nous ne cherchons ni à savoir quel argument doit être énoncé, ni à imposer des règles sur les arguments qui peuvent être énoncés. À l'inverse, nous nous plaçons en tant que spectateur d'un dialogue qui se déroule devant nous et cherchons à le modéliser de façon à pouvoir raisonner dessus. Ainsi, le dialogue est une entrée du système avec laquelle nous n'interagissons pas. Par cette approche que nous prenons, nos travaux diffèrent par nature de ce que l'on peut retrouver dans la littérature sur les systèmes de dialogue argumentatif (Black *et al.*, 2021) ou bien sur les *dialogue games* (Caminada, 2017).

Exemple 8. À partir du graphe d'argumentation représenté sur la figure 3.2, on considère dans cette section le couple $(\Delta, \mathcal{R})$ avec $\Delta = \{(a, 1), (b, 2), (c, 3), (d, 4)\}$.

3.3.1 Instanciation du contexte κ

Dans cette section, nous allons définir successivement les différents éléments de l'octuplet κ défini dans la Définition 3.2.

Pour pouvoir passer d'un graphe d'argumentation au langage d'action décrit en section 3.2.1, il faut d'abord définir les fluents $\mathbb{F}$ c'est-à-dire les variables nécessaires pour décrire le monde, ici le graphe d'argumentation. Deux éléments doivent être pris en compte : les nœuds du graphe, c'est-à-dire les arguments, et les arcs, c'est-à-dire les relations d'attaque.

Pour décrire un argument x , nous introduisons deux fluents : $p_x \in \mathbb{F}$ qui indique si l'argument est présent ou non dans le graphe¹ et $a_x \in \mathbb{F}$ qui indique l'acceptabilité de l'argument.

Pour l'ensemble des relations d'attaque $\mathcal{R}$, nous utilisons les fluents $cA_{y,x} \in \mathbb{F}$ exprimant le fait que y peut attaquer x . Comme dans ce chapitre nous ne traitons que le cas des AAF acycliques, il n'y a pas de cycles sur ces fluents : il n'existe pas de $(x_1, \dots, x_n) \in \mathcal{A}$ tel que $(cA_{x_1,x_2}, \dots, cA_{x_{n-1},x_n}, cA_{x_n,x_1}) \in \mathbb{F}$. Nous appelons cette propriété l'acyclicité des fluents cA . Nous n'introduisons ici qu'une notion d'attaque potentielle et non celle d'attaque effective. Nous verrons dans la suite du manuscrit que ces deux notions, bien que différentes, voient leur distinction disparaître dans les équations. Nous choisissons de garder la notion d'attaque potentielle car c'est celle qui est sémantiquement la plus proche de la relation d'attaque en argumentation.

Maintenant que l'ensemble des fluents $\mathbb{F}$ a été introduit, nous définissons l'état initial du dialogue $S(0)$. Aucun argument n'ayant encore été énoncé, ils sont tous absents. De plus, nous faisons l'hypothèse qu'un argument non présent est nécessairement inacceptable. Il s'agit d'une hypothèse permettant d'assurer que la cause initiale de l'acceptabilité d'un argument est son énonciation. Enfin, toutes les relations d'attaque sont connues

1. Nous utilisons cette notation plutôt que $p(x)$ car nous voulons éviter le malentendu consistant à penser qu'il s'agit là d'une fonction dont le résultat donnerait la valeur de vérité associée au fluent. Ici, il s'agit bien d'une variable propositionnelle qui est créée pour chaque argument.

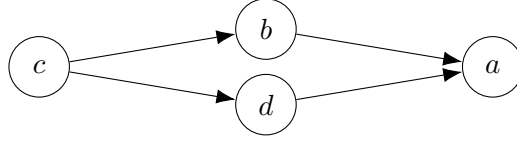


FIGURE 3.2 – Graphe d'argumentation associé à l'exemple 8.

dès le début du dialogue. Nous justifions ce choix quand nous présentons l'ensemble des actions possibles $\mathbb{A}$. Ainsi, nous avons $S(0) = \{\neg p_x, \neg a_x \mid x \in \mathcal{A}\} \cup \{cA_{y,x} \mid (y, x) \in \mathcal{R}\}$.

Exemple 8. (suite) – L'ensemble de fluents pour cet exemple est :

$$\mathbb{F} = \{p_a, p_b, p_c, p_d\} \cup \{a_a, a_b, a_c, a_d\} \cup \{cA_{b,a}, cA_{c,b}, cA_{c,d}, cA_{d,a}\}.$$

En ce qui concerne les événements $\mathbb{E}$, nous considérons uniquement l'action volontaire consistant à énoncer un argument x , notée $enunciate_x \in \mathbb{A}$. Pour cela, il faut que l'argument en question n'ait pas déjà été prononcé. Dans ce cas, x devient présent et acceptable par défaut. Ce choix est justifié car on évaluera son acceptabilité à l'état suivant avant qu'il n'ait un impact sur le reste du graphe. Formellement :

$$\begin{aligned} pre(enunciate_x) &\equiv \neg p_x \\ eff(enunciate_x) &\equiv p_x \wedge a_x \end{aligned}$$

Une autre action que nous aurions pu considérer ici est celle d'ajouter ou de retirer une relation d'attaque. Toutefois, dans cette thèse, nous restons sur le cadre de base des systèmes d'argumentation abstraite dans lequel la seule action possible est l'ajout d'un nouvel argument, c'est-à-dire $enunciate_x$.

Remarque – Aucun des événements décrits par la suite n'a pour effet de rendre un argument non présent. De fait si un argument venait à être répété, l'action ne se déclencherait pas, ses préconditions n'étant pas satisfaites. Une autre possibilité serait de permettre une telle répétition rendant l'argument accepté à nouveau acceptable. Cependant, dans le cas d'un graphe acyclique il n'y a qu'une seule extension complète qui correspond à l'ensemble admissible maximal au sens de l'inclusion (voir section 1.1.2) et donc l'état final des arguments est unique. Repasser le statut d'un argument d'inacceptable à acceptable n'a donc pas d'effet. Pour cette raison, nous avons choisi d'empêcher la répétition de se déclencher. Ce n'est plus vrai dans le cas des graphes cycliques dont nous discutons dans le chapitre 4.

La prise en compte de l'ordre d'énonciation des arguments implique de mettre à jour l'acceptabilité de tous les autres arguments présents après l'énonciation d'un nouvel argument et avant l'énonciation du suivant. Cela définit des états stables dans lesquels rien ne se passe si aucun argument n'est énoncé. Nous les appelons des *états argumentatifs*.

Définition 3.9 (État argumentatif). *Un état $S(t)$ est dit argumentatif si :*

- i) $\forall x, y, [S(t) \models a_x \wedge p_y \wedge cA_{y,x} \Rightarrow S(t) \models \neg a_y]$;
- ii) $\forall x, [S(t) \models p_x \wedge (\bigwedge_y \neg a_y \vee \neg cA_{y,x}) \Rightarrow S(t) \models a_x]$.

Nous avons choisi ce nom car un tel état correspond à une situation dans laquelle les règles de bases de l'argumentation sont respectées. En effet, la première condition indique que si un argument est acceptable alors tous ses attaquants sont inacceptables. La deuxième impose qu'un argument non attaqué, ou qui n'est attaqué que par des arguments non acceptables soit acceptable.

Exemple 8. (suite) – Considérons l'état $S(1)$ dans lequel uniquement l'argument a est présent et est donc acceptable c'est-à-dire $\forall x \in \mathcal{A} \setminus \{a\}, S(1) \models p_a \wedge a_a \wedge \neg p_x \wedge \neg a_x$.

D'après le dialogue Δ , l'argument suivant est l'argument b . On obtient donc l'état suivant $S(2)$ tel que $S(2) = (S(1) \triangleright \{enunciate_b\})$. En appliquant les effets de $enunciate_b$, on a $S(2) \models a_a \wedge p_b \wedge cA_{b,a}$. Ainsi, l'argument a est acceptable et l'argument b est présent et attaque l'argument a donc si l'état $S(2)$ est argumentatif alors on doit avoir que b n'est pas acceptable. Or $eff(enunciate_b) = p_b \wedge a_b$. Cet état n'est donc pas argumentatif. On se trouve dans un état intermédiaire dans lequel a et b sont acceptables alors que b attaque a , il faut donc mettre à jour l'acceptabilité de a et le rendre inacceptable.

Pour rappel, en argumentation abstraite, un argument n'est acceptable que s'il est non attaqué ou attaqué uniquement par des arguments inacceptables. De fait, il suffit que l'un des attaquants soit acceptable pour rendre l'argument attaqué inacceptable. Pour appliquer ces règles, nous souhaitons qu'après l'énonciation d'un argument, des mises à jour soient déclenchées automatiquement. Nous les représentons par deux événements exogènes : $makesUnacc_{y,x} \in \mathbb{U}$ et $makesAcc_x \in \mathbb{U}$.

Mise à jour de l'acceptabilité : Supposons que l'argument y venant d'être énoncé peut attaquer l'argument x , et que x et y sont acceptables. Alors, x étant attaqué par un argument acceptable y , x devient inacceptable. Formellement, l'évènement exogène $makesUnacc_{y,x}$ peut s'écrire :

$$\begin{aligned} tri(makesUnacc_{y,x}) &\equiv a_x \wedge a_y \wedge cA_{y,x} \\ eff(makesUnacc_{y,x}) &\equiv \neg a_x \end{aligned}$$

Cette écriture permet également de traiter les cas où un nouvel argument z rend un attaquant y de x à nouveau acceptable. Dans ce cas, x devient inacceptable.

Exemple 8. (suite) – Nous sommes donc dans un état $S(2)$ tel que $S(2) \models a_a \wedge a_b \wedge cA_{b,a}$. On a donc $S(2) \models tri(makesUnacc_{b,a})$ et donc $E(2) = \{makesUnacc_{b,a}\}$. On arrive alors dans un état $S(3)$ tel que $S(3) \models p_a \wedge \neg a_a \wedge p_b \wedge a_b$ qui est un état argumentatif. On peut donc énoncer l'argument suivant c .

On a $E(3) = \{enunciate_c\}$ et $S(4) \models a_b \wedge a_c \wedge cA_{c,b} = tri(makesUnacc_{c,b})$. Comme précédemment, $E(4) = \{makesUnacc_{c,b}\}$ et donc $S(5) \models \neg a_a \wedge \neg a_b \wedge a_c \wedge cA_{b,a} \wedge cA_{c,b}$. En particulier, $S(5) \models p_a \wedge (\bigwedge_y \neg a_y \vee \neg cA_{y,a}) \wedge \neg a_a$. Il ne s'agit donc pas d'un état argumentatif car la condition ii) de la définition 3.9 n'est pas vérifiée.

Du point de vue de l'argumentation, cela vient du fait que l'argument a n'est pas acceptable alors qu'il n'est attaqué que par des arguments inacceptables. Il faut un autre évènement exogène pour mettre à jour l'acceptabilité de a .

Mise à jour de l'inacceptabilité : Supposons que l'argument x est inacceptable et qu'un argument z vient d'être prononcé. Celui-ci n'a pas de lien direct avec l'argument x mais a

pu modifier l'acceptabilité de certains attaquants de x . On vérifie donc, a présent, si tous les arguments pouvant attaquer x sont acceptables ou non. Si aucun d'entre eux n'est effectivement acceptable, alors x le redevient. Dans le langage d'action, cela se traduit par l'évènement exogène $makesAcc_x$ tel que :

$$\begin{aligned} tri(makesAcc_x) &\equiv p_x \wedge \neg a_x \wedge \left(\bigwedge_y \neg cA_{y,x} \vee \neg a_y \right) \\ eff(makesAcc_x) &\equiv a_x \end{aligned}$$

Exemple 8. (suite) – L'état actuel du dialogue est donné par $S(5)$ qui n'est pas argumentatif. En particulier, $S(5) \models p_a \wedge \neg a_a \wedge \left(\bigwedge_y \neg cA_{y,a} \vee \neg a_y \right) = tri(makesAcc_a)$ et donc $E(5) = \{makesAcc_a\}$. Le nouvel état obtenu $S(6)$ est un état argumentatif car a est redevenu acceptable.

Relation d'ordre sur les évènements : Lorsqu'un argument x est énoncé, il faut vérifier qu'il n'est pas rendu inacceptable par un argument y déjà présent avant qu'il ne rende inacceptables d'autres arguments, z par exemple. Les règles où x est le second argument sont donc déclenchées avant celles où il est premier. Cela se traduit par la règle de priorité ci-dessous :

$$makesUnacc_{y,x} \succ_{\mathbb{E}} makesUnacc_{x,z}$$

Exemple 8. (suite) – $S(6)$ est un état argumentatif, on peut donc énoncer le prochain argument d . On a $E(6) = \{enunciate_d\}$ et donc $S(7) \models (a_a \wedge a_d \wedge cA_{d,a}) \wedge (a_d \wedge a_c \wedge cA_{c,d})$ c'est-à-dire que $S(7) \models tri(makesUnacc_{d,a}) \wedge tri(makesUnacc_{c,d})$.

Imaginons que la relation d'ordre précédente n'existe pas, alors les deux évènements exogènes se déclenchent, c'est-à-dire que $E'(7) = \{makesUnacc_{c,d}, makesUnacc_{d,a}\}$. On atteint un état $S'(8)$ tel que $S'(8) \models p_a \wedge \neg a_a \wedge \left(\bigwedge_y \neg cA_{y,a} \vee \neg a_y \right)$ car d a été rendu inacceptable par c . On obtient donc $S'(9) = S'(8) \triangleright \{makesAcc_a\}$.

En utilisant la règle précédente on a $makesUnacc_{c,d} \succ_{\mathbb{E}} makesUnacc_{d,a}$, ce qui implique que $E(7) = \{makesUnacc_{c,d}\}$. On obtient ainsi directement un état $S(8) = S'(9)$.

Notons qu'ajouter un argument dans le graphe ne peut avoir d'influence directe sur les autres arguments qu'en les rendant inacceptables. Pour cette raison, il n'est pas nécessaire d'établir une règle de priorité de la forme $makesUnacc_{y,x} \succ_{\mathbb{E}} makesAcc_z$ car cette situation est déjà couverte par la règle précédente.

Remarque – Dans la transformation proposée $\mathcal{T}_{AFa2AL}$, on ne fait pas de distinction entre la notion d'attaque potentielle et celle d'attaque réelle car la différence entre ces dernières disparaît dans les équations. En effet, considérons un fluent $attack_{y,x} \in \mathbb{F}$ traduisant le fait que l'argument y attaque effectivement l'argument x . Définissons l'évènement exogène $isAttacking_{y,x} \in \mathbb{U}$ comme :

$$\begin{aligned} tri(isAttacking_{y,x}) &\equiv p_x \wedge p_y \wedge cA_{y,x} \\ eff(isAttacking_{y,x}) &\equiv attack_{y,x} \end{aligned}$$

D'après cette définition, y attaque x si les deux arguments sont présents et y peut attaquer x . Cependant, pour que cette attaque soit prise en compte, il faut toujours que

l'attaquant y soit acceptable. On obtient des conditions de la forme $a_y \wedge attack_{y,x}$, c'est-à-dire $a_y \wedge p_y \wedge cA_{y,x}$. Or un argument ne peut pas être acceptable sans être présent donc $a_y \wedge p_y \equiv a_y$. Ainsi, avec ce nouveau fluent on aurait : $tri(makesUnacc_{y,x}) = a_y \wedge a_x \wedge attack_{y,x} = a_y \wedge a_x \wedge cA_{y,x}$. On retrouve la même précondition que sans l'introduction de *isAttacking* et de *attack*. De fait, nous avons choisi de n'utiliser que *cA*.

Dans cette section, nous avons défini le contexte κ adapté à un système d'argumentation abstraite. En particulier, nous avons défini les ensembles suivants :

- les fluents $\mathbb{F} = \{p_x, a_x \mid x \in A\} \cup \{cA_{y,x} \mid (y, x) \in R\}$;
- les événements $\mathbb{E} = \{enunciate_x \mid x \in A\} \cup \{makesAcc_x, makesUnacc_{y,x} \mid x, y \in A\}$ avec *pre*, *tri*, *eff* et $\succ_{\mathbb{E}}$ définis ci-dessus.

3.3.2 Modification de la sémantique

Après avoir défini le contexte κ pour le cadre argumentatif, il faut modifier les définitions associées à la sémantique du langage d'action. Cela va permettre en particulier d'obtenir des traces représentatives de la réalité. Pour cela, les arguments sont énoncés à partir d'états argumentatifs étape par étape dans l'ordre de l'interaction. Comme nous ne considérons que des graphes acycliques dans ce chapitre, il existe toujours un tel état après l'ajout d'un nouvel argument et il est donc toujours possible de continuer l'interaction.

La définition actuelle du scénario σ , telle que rappelée dans la définition 3.6, n'est pas adaptée à ce cas. En effet, à chaque action est associée une date à laquelle elle se produit. De fait, pour que l'énonciation d'un nouvel argument se produise uniquement pendant un état argumentatif il faudrait connaître préalablement le nombre d'étapes nécessaires pour revenir à un état argumentatif une fois un argument énoncé. Cela n'est évidemment pas souhaitable. Nous proposons, pour résoudre ce problème, de ne plus dater les actions mais seulement de les ordonner. Cela définit non plus un scénario mais ce que nous appelons une *séquence*.

Définition 3.10 (Séquence ς). *Une séquence, notée ς , est un ensemble fini d'actions ordonnées : $\varsigma \subseteq \mathbb{A} \times \mathbb{N}$.*

Définition 3.11 (Configuration argumentative χ). *La configuration argumentative, notée χ , est le couple (ς, κ) avec ς une séquence et κ un contexte.*

Exemple 8. (suite) – *Durant tout cet exemple, nous avons attendu un état argumentatif avant d'énoncer le prochain argument. Sans les modifications présentées dans cette section, ce n'est pas ce qui se serait normalement passé. En effet, sans ces dernières, les arguments auraient dû être énoncés à la date correspondant à leur valeur dans le dialogue. Par exemple, l'argument c a été énoncé à la date $t = 4$, c'est-à-dire que $enunciate_c \in E(4)$. Or dans le dialogue, c est associé à la date 3. Avec la définition de la sémantique donnée dans la section 3.2.1, on aurait eu $enunciate_c \in E(3)$. La différence est ici que la valeur associée à chaque argument dans le dialogue correspond à son ordre d'énonciation et non à sa date d'énonciation.*

L'unicité de l'exécution valide n'est plus obtenue grâce au scénario σ , mais à la séquence ς . Il faut donc modifier les définitions 3.5 et 3.7.

Les actions étant maintenant ordonnées et non datées, il faut tout d'abord modifier la définition d'une exécution valide de façon à ce que les actions ne se produisent que

lorsque plus aucun évènement exogène ne se déclenche. Pour cela nous ajoutons une nouvelle condition (2.d) à la définition 3.5, comme l'exprime la définition 3.12 ci-dessous. Ensuite, la condition 2.c est également modifiée pour ne plus s'appliquer qu'aux évènements exogènes. Enfin, une dernière condition, 2.e, empêche qu'un ensemble d'évènements soit vide. Pour les conditions 1, 2.a, 2.b, et 3 rien ne change : nous ne modifions rien vis-à-vis du déclenchement des évènements exogènes.

Définition 3.12 (Exécution valide dans le cadre des AAF). *Une exécution est une séquence : $E(-1), S(0), E(0), \dots, E(N), S(N+1)$. Elle est valide étant donné κ si elle vérifie, $\forall t \in \mathbb{T}$:*

1. $S(t) \subseteq \text{Lit}_{\mathbb{F}}$ est un état au sens de la définition 3.1.
2. $E(t) \subseteq \mathbb{E}$ vérifie :
 - (a) $\forall e \in E(t), S(t) \models \text{pre}(e)$;
 - (b) $\nexists (e, e') \in E(t)^2, e \succ_{\mathbb{E}} e'$;
 - (c) $\forall e \in \mathbb{U}$ tel que $S(t) \models \text{tri}(e), e \in E(t)$ ou $\exists e' \in E(t), e' \succ_{\mathbb{E}} e$;
 - (d) si $\exists e \in E(t) \cap \mathbb{A}$, alors $\forall e' \in \mathbb{U}, S(t) \not\models \text{tri}(e')$;
 - (e) $E(t) \neq \emptyset$.
3. $S(t+1) = S(t) \triangleright E(t)$.

Dans la définition 3.7, les traces ont été définies comme des extraits d'une exécution valide compte tenu de κ et de conditions supplémentaires liées à σ . Au lieu de définir directement des traces, la définition 3.13 ci-dessous correspond à une exécution valide étant donné la configuration $\chi = (\varsigma, \kappa)$. Les traces sont simplement des extraits de ces exécutions valides.

Définition 3.13 (Exécution valide étant donné χ). *Soit une configuration argumentative $\chi = (\varsigma, \kappa)$. Une exécution valide étant donné κ est valide étant donné χ si :*

1. $\forall t \in \mathbb{T}, E(t) \subset (\{a, \exists o \in \mathbb{N}, (a, o) \in \varsigma\} \cup \mathbb{U})$;
2. $\forall ((e, o), (e', o')) \in \varsigma^2$ tels que $o < o', \exists t, t'$ tel que $e \in E(t)$ et $e' \in E(t')$ et $t < t'$;
3. $\forall ((e, o), (e', o')) \in \varsigma^2$ tel que $o = o', \exists t$ tel que $(e, e') \in E(t)^2$.

Soit une exécution valide étant donné χ . Sa trace d'évènements τ_{χ}^e est sa séquence d'évènements $E(-1), E(0), \dots, E(N)$, et sa trace d'états τ_{χ}^s sa séquence d'états $S(0), S(1), \dots, S(N+1)$.

3.3.3 Implémentation en Answer Set Programming (ASP)

Pour obtenir les traces utilisées pour les différents exemples, nous avons adapté l'implémentation correcte et complète du langage d'action proposé par Sarmiento *et al.* (2023) en ASP. Pour les détails de cette implémentation, nous invitons le lecteur à se référer à cet article. Nous ne la décrivons ici que brièvement et détaillons les modifications principales que nous avons apportées.

Les programmes ASP $\pi_{con}(\kappa)$ and $\pi_{seq}(\varsigma)$ sont obtenus par la traduction respectivement du contexte κ et de la séquence ς . Le programme $\pi_{\mathbb{C}}$ est identique à celui proposé par Sarmiento *et al.* (2023) et permet de calculer la trace causale. Enfin, $\pi_{\mathbb{A}}$ est obtenu

par la traduction de la sémantique du langage d'action introduite dans la section 3.2.1 et modifiée dans la section 3.3.2.

Le changement le plus important concerne le moment à partir duquel une action peut être effectuée. Tout d'abord, nous passons d'un scénario à une séquence. Contrairement au programme original, dans lequel les actions sont exécutées à l'état déterminé par le scénario, dans notre programme une action ordonnée O ne peut être exécutée que si l'action précédente, c'est-à-dire ordonnée $O-1$ dans la séquence, a déjà été effectuée. Pour ce faire, une règle appelée `done` est ajoutée pour vérifier si une action a déjà été effectuée ou non. Par conséquent, la règle `triggered` doit également être modifiée pour tenir compte du fait que l'action précédente doit être terminée avant que la suivante puisse être lancée. De plus, une action peut être exécutée si et seulement si aucun événement exogène, appelé `auto`, n'est déclenché. Nous ne détaillons pas les autres notations dont les noms sont explicites, celles-ci étant définies par Sarmiento *et al.* (2023). Il en résulte les modifications suivantes :

```
done(O, T) :- happens(A, _, T1), performs(A, O), time(T), T1 < T.
triggered(A, GD, T) :- action(A, GD, Effect), performs(A, O), done(O-1, T),
                        holds(GD, T), time(T),
                        not triggered(U, _, T) : auto(U, _, _).
```

Le programme complet $\Pi(\chi) = \pi_{seq}(\varsigma) \cup \pi_{con}(\kappa) \cup \pi_A \cup \pi_C$ est disponible².

3.4 Étude des propriétés formelles de la transformation $\mathcal{T}_{AFa2AL}$

Dans cette section, nous étudions les propriétés formelles de la transformation $\mathcal{T}_{AFa2AL}$ que nous proposons. Tout d'abord, nous montrons que la notion de temporalité est bien prise en compte par cette transformation. Ensuite, nous établissons sa terminaison puis sa correction et sa complétude. Enfin, nous introduisons une proposition qui ouvre la voie à la discussion de la section 3.5. Toutes les preuves sont détaillées dans l'annexe B, et seuls leurs principes sont brièvement exposés ici.

3.4.1 Unicité des traces

Le premier résultat montre que, bien que les exécutions valides étant donné κ ne soient pas uniques, les exécutions valides étant donné χ le sont, ainsi que les traces correspondantes τ_χ^e et τ_χ^s .

Proposition 3.3 (Unicité des traces). *Soit une configuration argumentative $\chi = (\varsigma, \kappa)$, les traces τ_χ^e et τ_χ^s sont uniques.*

Principe de la preuve. S'il existait au moins deux exécutions valides étant donné χ alors, comme le contexte est le même, l'état initial serait également identique pour les deux exécutions. Donc, s'il existe une différence entre ces deux exécutions, elle apparaît pour la première fois au cours d'une transition d'état. Pour cette preuve, on suppose donc qu'il existe une première date t_0 telle que $E(t_0) \neq E'(t_0)$ et que, avant cette date les deux exécutions valides sont identiques. Or, l'état précédent étant le même entre les

2. https://gitlab.lip6.fr/sarmiento/temporality_and_causality_in_abstract_argumentation.git

deux exécutions, on arrive à une contradiction, peu importe la nature de l'évènement qui diffère :

- si c'est un évènement exogène : la condition 2.c de la définition 3.12 n'est pas respectée ;
- si c'est une action : soit la condition 2.e de la définition 3.12 soit la condition 3 de la définition 3.13 n'est pas respectée.

□

Dans la suite du chapitre, lorsqu'il est question d'évènements et d'états, il s'agit de ceux étant donné les traces uniques τ_χ^e et τ_χ^s . Ainsi, l'ensemble des évènements qui se sont effectivement produits à l'instant t est $E^\chi(t) = \tau_\chi^e(t)$. De même, l'état réel à l'instant t est $S^\chi(t) = \tau_\chi^s(t)$.

3.4.2 Terminaison de $\mathcal{T}_{AFa2AL}$

Nous établissons à présent la complétude et la correction de la transformation $\mathcal{T}_{AFa2AL}$ proposée. Pour cela, nous introduisons d'abord la notion de graphe associé à un état.

Définition 3.14 (Graphe associé à un état). *Soit $S^\chi(t)$ un état. On appelle graphe associé à cet état le graphe $AF' = (\mathcal{A}', \mathcal{R}')$, tel que $\mathcal{A}' = \{x \mid S^\chi(t) \models p_x\}$ et $\mathcal{R}' = \{(y, x) \mid S^\chi(t) \models cA_{y,x}\}$.*

Nous avons introduit dans la définition 3.9 un état argumentatif comme un état qui respecte les règles de base de l'argumentation. Or les règles de mise à jour que nous avons proposées servent à les faire appliquer. Pour cette raison, nous proposons une caractérisation d'un état argumentatif comme un état dans lequel rien ne se passe tant qu'une action volontaire n'est pas effectuée.

Lemme 3.1 (Caractérisation d'un état argumentatif). *Soit $S^\chi(t)$ un état. Les deux propositions suivantes sont équivalentes :*

- $\forall e \in \mathbb{U}, S^\chi(t) \not\models tri(e)$
- $S^\chi(t)$ est un état argumentatif d'après la définition 3.9.

Principe de la preuve. Les évènements exogènes ayant été construits dans ce but, il ne s'agit ici que d'une réécriture de la négation de ces deux évènements. En particulier, $S(t) \models \neg tri(makesAcc_x) \equiv (ii)$ de la définition 3.9 et $S(t) \models \neg tri(makesUnacc_{y,x}) \equiv (i)$ de la définition 3.9. □

Nous montrons maintenant qu'il est toujours possible d'atteindre un état argumentatif à partir d'un état argumentatif dans lequel un argument $x \in \mathcal{A}$ est énoncé.

Proposition 3.4 (Terminaison). *Soit $S^\chi(t)$ un état argumentatif et $x \in \mathcal{A}$ un argument. Si $enunciate_x \in E^\chi(t)$, alors $\exists t' \in \mathbb{T}, t < t'$ tel que $S^\chi(t')$ est un état argumentatif.*

Principe de la preuve. Le graphe d'argumentation étant acyclique et les nombres d'arguments ainsi que d'évènements exogènes étant finis, il y a un nombre fini maximum d'états intermédiaires. □

3.4.3 Correction et complétude de $\mathcal{T}_{AFa2AL}$

La proposition suivante permet de prouver qu'un argument acceptable dans l'état argumentatif est acceptable dans le graphe d'argumentation associé et vice-versa.

Proposition 3.5 (Correction de l'acceptabilité d'un argument). *Soit $S^x(t)$ un état argumentatif et $AF' = (A', \mathcal{R}')$ son graphe associé. Alors, $\forall x \in A'$, x est acceptable par A' si et seulement si $S^x(t) \models a_x$.*

Pour montrer ce résultat nous introduisons un lemme intermédiaire permettant de caractériser un argument non acceptable dans le langage d'action.

Lemme 3.2 (Caractérisation d'un argument non acceptable). *Soit $S^x(t)$ un état. Pour tout $x \in A$ on a : $S^x(t) \models p_x \wedge \neg a_x \Leftrightarrow S^x(t) \models \exists y, p_x \wedge a_y \wedge cA_{y,x}$.*

Une fois ce lemme établi, on peut prouver la proposition 3.5.

Principe de la preuve. On commence par montrer qu'être acceptable dans AF implique l'acceptabilité dans l'état associé. Pour cela, on raisonne par l'absurde. On construit une chaîne d'arguments qui attaque puis défend un argument x_0 supposé inacceptable. Comme le nombre d'arguments est fini et le graphe acyclique, on atteint un élément non attaqué. Or comme x_0 est supposé inacceptable, cet élément est un attaquant indirect de x_0 . Cela est contradictoire avec le graphe d'argumentation associé. Sinon, un événement exogène devrait se déclencher ce qui contredit l'hypothèse d'état argumentatif.

Pour l'implication réciproque, on applique l'implication précédente aux attaquants de x_0 qui sont inacceptables dans le langage d'action et que l'on suppose acceptables dans le graphe associé. Cela conduit à la contradiction que x_0 est non acceptable dans le langage d'action. $\square$

Nous avons établi qu'il existe une équivalence entre être acceptable dans un état argumentatif et être acceptable dans le graphe d'argumentation associé. La question restante est alors de savoir ce qu'il se passe une fois que tous les arguments d'un dialogue ont été énoncés. Avec le théorème suivant, nous établissons l'existence d'un état argumentatif dont le graphe associé est égal à l'AAF initial. Un tel état est appelé *état argumentatif final* et est défini comme un état argumentatif $S^x(t)$ tel que $\forall x \in A, \exists t' \in \mathbb{T}$ tel que $t' < t$ et $enunciate_x \in E^x(t')$.

Théorème 3.1 (Correction et complétude). *Soit un dialogue Δ , R un ensemble de relations d'attaque et $AF = (A, R)$ le graphe d'argumentation que (Δ, R) induisent. Étant donné une configuration argumentative χ , le graphe argumentatif AF' associé à l'état argumentatif final $S^x(t)$ vérifie : $AF' = AF$.*

Principe de la preuve. Comme $S(0)$ est un état argumentatif, on peut appliquer la proposition 3.4 pour déterminer l'existence d'un état argumentatif final. Ensuite d'après la proposition 3.5, on obtient un graphe d'argumentation associé avec les mêmes arguments et les mêmes relations d'attaque que le graphe initial, conduisant au résultat. $\square$

Les résultats précédents permettent de montrer la cohérence de l'état final du langage d'action avec l'argumentation. En particulier, le théorème 3.1 est essentiel car il établit la correction et la complétude de notre approche avec l'argumentation, et permet ainsi d'assurer qu'aucune information n'est perdue.

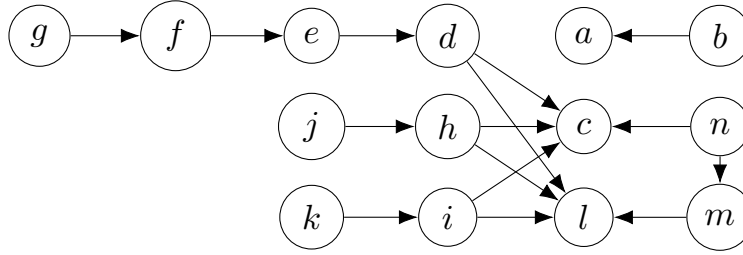


FIGURE 3.3 – Graphe d'argumentation associé à l'exemple 1 simplifié.

À l'inverse, intégrer la temporalité permet d'ajouter des informations supplémentaires grâce aux états intermédiaires, comme illustré dans la section suivante, et aux relations causales qui peuvent en être déduites. Ainsi, on a le résultat suivant.

Proposition 3.6. *Les relations causales sont dépendantes de la séquence ς .*

Principe de la preuve. En utilisant la version simplifiée de l'exemple 1 et sa version modifiée, on obtient des chaînes causales différentes. $\square$

Cette proposition montre que les relations causales sont dépendantes de l'ordre d'énonciation des arguments. Ainsi, même si, comme dans le cadre classique de l'argumentation, l'acceptabilité d'un argument dans l'état argumentatif final n'en dépend pas (cf théorème 3.1), il est tout de même essentiel de tenir compte de la temporalité lorsque l'on s'intéresse à des notions proches de la causalité notamment pour la recherche d'explications. Ce résultat peut être visualisé à partir de l'exemple 1 et sa version modifiée, illustrant l'enrichissement qu'apporte le passage à un langage d'action. L'utilisation de ces relations causales est discutée dans la section suivante.

3.5 Discussion sur la génération d'explications

En transformant un graphe d'argumentation abstraite acyclique dans le langage d'action tel que présenté dans la section 3.3, nous obtenons en sortie deux traces : la trace d'états décrivant les différents états du monde, donc les différentes étapes de l'argumentation au cours du temps, et la trace d'événements rapportant l'ensemble des événements s'étant produits au cours du dialogue, c'est-à-dire les arguments énoncés ainsi que les mises à jour d'acceptabilité qui en découlent. Grâce à ces traces, nous présentons dans cette section un processus en trois étapes pour rechercher des explications. En particulier, nous discutons deux types d'explication en argumentation d'après la classification faite par Čyras *et al.* (2021) (voir section 1.3.2) : tout d'abord nous proposons deux formes de représentation graphique du processus puis nous discutons de la génération d'explications causales. Tout au long de cette section, nous utilisons l'exemple 1 dans sa version simplifiée comme support de discussion ; son graphe est rappelé sur la figure 3.3.

3.5.1 Représentation graphique comme support de l'explication

Les systèmes de transition tels que les langages d'action modélisent le monde d'une façon adaptée pour une représentation visuelle. En effet, le principe de décrire le monde

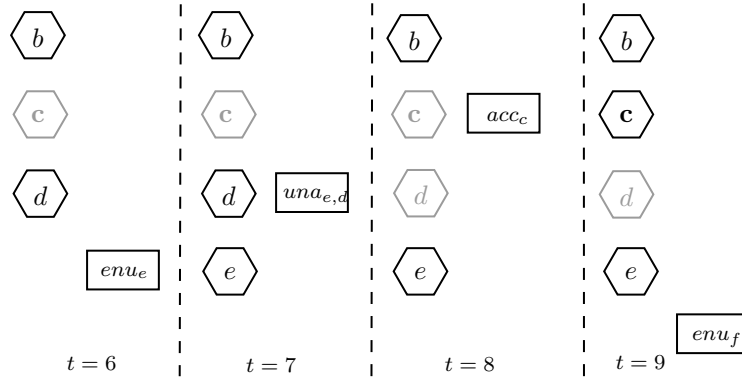


FIGURE 3.4 – Représentation graphique partielle associée à l’exemple 1. Les hexagones représentent des fluents et les rectangles des évènements.

par l’intermédiaire d’états dont l’évolution est entraînée par des évènements fournit une description sous forme de narration de l’interaction modélisée. Pour exploiter ce comportement, nous utilisons les deux traces qui enregistrent l’histoire du monde.

En particulier, nous proposons sur la figure 3.4 un affichage possible des traces d’évènements et d’états obtenues en appliquant notre transformation à l’exemple 1. Sur cette forme de frise chronologique du dialogue, les fluents sont représentés sous forme d’hexagones. Plus précisément, puisque l’acceptabilité des arguments est ce qui importe le plus dans le dialogue, nous ne représentons que les fluents a_x en utilisant uniquement les noms des arguments pour des raisons de lisibilité. D’autre part, nous ne faisons pas apparaître les fluents si l’argument n’est pas présent. Enfin, un argument non acceptable est représenté avec une teinte plus claire. Les évènements sont eux représentés par des rectangles, et leur nom est abrégé : $enunciate_x$, $makesAcc_x$ et $makesUnacc_{y,x}$ deviennent respectivement enu_x , acc_x et $una_{y,x}$.

Exemple 1 (Examen médical simplifié). (suite) – La figure 3.4 est une représentation partielle de la trace d’état obtenue pour l’exemple 1 en utilisant l’implémentation en ASP décrite dans la section 3.3.3.

Le premier état représenté correspond à $S(6)$, un état argumentatif au sens de la définition 3.9 et qui permet donc l’énonciation de l’argument suivant e , à savoir que la pathologie est visible sur une IRM d’après le guide des bonnes pratiques en radiologie. En effet, puisque tous les arguments précédant e ont déjà été énoncés et qu’aucune mise à jour d’acceptabilité n’est requise, l’action $enunciate_e$ peut être effectuée. L’occurrence de cet évènement produit la transition vers l’état suivant $S(7)$ où, comme le montre la figure 3.4, l’argument e est acceptable. Contrairement à $S(6)$, $S(7)$ n’est pas un état argumentatif : la condition (i) de la définition 3.9 n’est pas satisfaite car $(a_d \wedge cA_{e,d})$ et $a_e \in S(7)$. Par conséquent, l’argument suivant ne peut pas être énoncé. Comme les conditions de déclenchement de $makesUnacc_{e,d}$ sont satisfaites, cet évènement exogène est déclenché, ce qui conduit à une nouvelle transition d’état. Puisque l’argument d , c’est-à-dire qu’on ne peut pas voir la pathologie à l’IRM, n’est plus acceptable dans $S(8)$, la condition (i) de la définition 3.9 est maintenant satisfaite. Cependant, la condition (ii) n’est pas satisfaite par $S(8)$, ce qui empêche à nouveau l’énonciation de l’argument suivant. Au lieu de cela, $makesAcc_c$ est déclenché, ce qui conduit à l’état suivant $S(9)$. Ici, comme le montre la fi-

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>	<i>h,i</i>	<i>j</i>	<i>k</i>	<i>l</i>	<i>m</i>	<i>n</i>
<i>a</i>	•	○	○	○	○	○	○	○	○	○	○	○	○
<i>b</i>		•	•	•	•	•	•	•	•	•	•	•	•
<i>c</i>			•	○	•	○	•	○	○	•	•	•	○
<i>d</i>				•	○	•	○	○	○	○	○	○	○
<i>e</i>					•	○	•	•	•	•	•	•	•
<i>f</i>						•	○	○	○	○	○	○	○
<i>g</i>							•	•	•	•	•	•	•
<i>h</i>								•	○	○	○	○	○
<i>i</i>								•	•	○	○	○	○
<i>j</i>									•	•	•	•	•
<i>k</i>										•	•	•	•
<i>l</i>											•	○	•
<i>m</i>												•	○
<i>n</i>													•

FIGURE 3.5 – Représentation compacte du dialogue complet de l'exemple 1.

gure 3.4, l'argument *c* est acceptable. Comme ce nouvel état est argumentatif, l'argument suivant *f* peut être énoncé. Le dialogue se poursuit ainsi pas à pas jusqu'à l'état $S(31)$. A cet état, tous les arguments ont été énoncés et les mises à jour d'acceptabilité sont terminées, ce qui fait de $S(31)$ l'état argumentatif final. En particulier, les fluents a_a et a_c sont faux et le fluent a_l est vrai : tout comme pour un AAF, la décision finale est de faire une IRM aujourd'hui.

Cette représentation graphique partielle révèle des relations causales qui peuvent être déduites immédiatement : ce sont les NESS-causes directes. Elles peuvent être utilisées pour fournir des explications de base telles que le fait que l'énonciation de l'argument *e* au temps $t = 6$ fait que *e* est acceptable dans l'état suivant, c'est-à-dire au temps $t = 7$. Toutefois, si l'on demande pourquoi l'argument *c* est devenu acceptable à $t = 9$, une telle représentation a pour réponse que c'est parce que *makesAcceptable_c* a été déclenché. Cette relation causale indique que l'acceptabilité de *c* a été restaurée, ce qui signifie que tous ses attaquants ont été rendus inacceptables. On obtient donc ici des informations supplémentaires. Cependant, ces relations causales simples entre états ne sont pas suffisantes pour fournir des explications adaptées à un agent humain. En effet, expliquer que si l'argument *c* est acceptable c'est parce que son acceptabilité a été restaurée n'est pas satisfaisant. Il faudrait être capable de remonter à des causes antérieures, de préférence liées à des action et non des événements exogènes. Pour résoudre ce problème, un raisonnement causal plus complexe permet de déterminer des relations causales plus pertinentes à savoir les NESS-causes et les causes effectives. La représentation graphique illustrée sur la figure 3.4 reste toutefois utile comme support de l'explication pour aider à mieux comprendre le processus argumentatif et le déroulement de l'interaction.

Nous proposons une seconde représentation graphique, plus compacte sous forme de tableau, illustré par la figure 3.5. Ici, la première ligne indique l'argument qui est énoncé : *enunciate_x* est remplacé par *x* pour rendre le tableau plus lisible. Sur les lignes du dessous, nous pouvons suivre l'évolution de l'acceptabilité des arguments : • signifie que l'argument est acceptable, ○ que l'argument est inacceptable et une cellule grisée que l'argument n'a pas encore été énoncé. Contrairement à la représentation précédente,

ς_1	c	d	e	f	g	h,i	j	k	l	m	n
c	•	◦	•	◦	•	◦	◦	•	•	•	◦
ς_2	c	l	m	n	d	e	f	g	h,i	j	k
c	•	•	•	◦	◦	◦	◦	◦	◦	◦	◦

FIGURE 3.6 – Importance de l’ordre d’énonciation des arguments (lignes 1, 3) sur l’acceptabilité des arguments (lignes 2, 4).

seuls les états argumentatifs sont représentés. Il s’agit d’une représentation plus adaptée pour visualiser l’intégralité du dialogue ainsi que les effets de l’énonciation d’un argument. Par exemple, sur la 5^e colonne, on peut voir que l’énonciation de *d* a fait que *c* est devenu inacceptable. Cependant, contrairement à la représentation de la figure 3.4, elle ne fournit pas le détail du processus qui a conduit à ce changement.

Un autre avantage d’utiliser ce tableau est qu’il permet de visualiser facilement l’impact de l’ordre d’énonciation. Pour illustrer cela, reprenons l’exemple 1 :

Exemple 1 (Examen médical simplifié). *(version modifiée) – Considérons le même dialogue que dans l’exemple 1 simplifié, en commençant par l’énonciation des arguments *a, b, c*, mais en considérant cette fois-ci que le médecin demande ensuite directement s’il est possible de faire l’IRM aujourd’hui (*l*). Le radiologue répond qu’il ne peut le faire que dans deux jours au plus tôt (*m*). Le médecin précise alors qu’il s’agit d’une urgence (*n*). Les autres arguments sont ensuite énoncés dans le même ordre que dans l’exemple initial.*

*La figure 3.6 présente la visualisation compacte proposée pour l’évolution de l’acceptabilité de la variable de décision *c*, à partir de son énonciation. Même si l’état final du graphe d’argumentation est identique, c’est-à-dire que *c* est rejeté, cette visualisation permet d’observer facilement l’importance que l’ordre des actions a sur les étapes intermédiaires qui y conduisent : dans le nouveau scénario, *c* est inacceptable à partir de la 6^e action, c’est-à-dire l’énonciation de *n*, sans aucune modification jusqu’à la fin. Cela indique que *n* est l’argument clé entraînant le choix définitif.*

Avec cette visualisation, nous voyons clairement la pertinence de l’intégration de la temporalité dans le cadre de l’argumentation : bien que le résultat soit le même les différences entre les deux scénarios ne peuvent pas être saisies par des AAF classiques.

3.5.2 Construction de la chaîne causale

Au-delà des représentations graphiques du processus d’acceptation ou de rejet, la formalisation proposée des AAF dans le langage d’action fournit des outils nous permettant de transférer la notion de causalité effective, telle que rappelée dans la section 3.2.2, au cadre de l’argumentation abstraite. Cela va permettre la génération d’explications causales plus complètes et plus adaptées au destinataire de l’explication. En effet, Miller (2019) a souligné l’importance de l’extraction de chaînes causales pour la recherche d’explications.

En utilisant les trois relations causales décrites dans la section 3.2.2, nous proposons une représentation graphique plus riche, illustrée sur la figure 3.7. Elle considère le même principe qu’introduit sur la figure 3.4 en ajoutant trois types de flèches représentant les trois types de relations causales que nous avons introduit dans la section 3.2.2 :

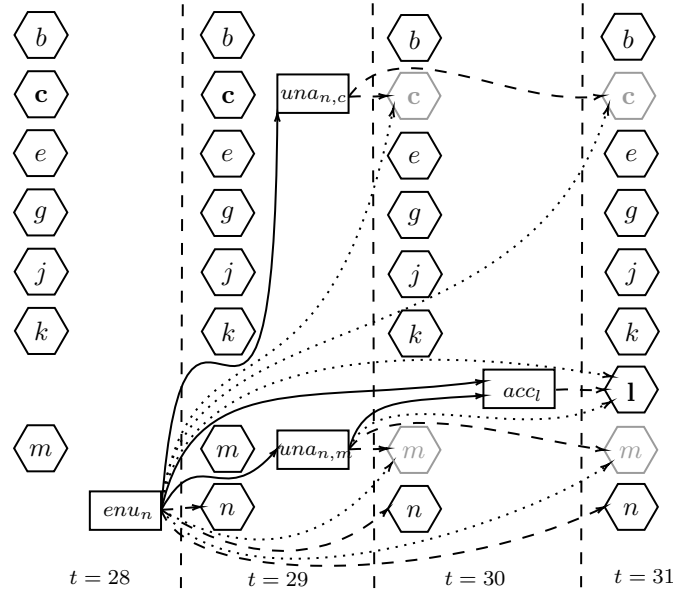


FIGURE 3.7 – Représentation graphique de l'exemple 1 incluant les relations causales suivantes : NESS-causes directes (—), NESS-causes (---), et causes effectives (—).

les NESS-causes directes (—), les NESS-causes n'étant pas des NESS-causes directes (---), et les causes effectives (—). Elles permettent de mieux comprendre le processus d'argumentation : par exemple, ces relations causales supplémentaires nous indiquent que l'énonciation de n est la cause du fait que l est acceptable et que c et m ne le sont pas, c'est-à-dire que le fait qu'il s'agisse d'une urgence a entraîné le fait de prévoir une IRM aujourd'hui plutôt que dans trois jours, bien qu'il n'y ait officiellement plus de créneau disponible. De plus, en examinant les relations de causalité effective, on peut comprendre, par exemple, que l'énonciation de n rend l acceptable en rendant m inacceptable, ce qui est beaucoup plus satisfaisant. Des relations plus complexes de ce type peuvent être obtenues en reconstruisant la chaîne à l'envers. Cela peut aider à expliquer le dialogue de manière simple lorsque le nombre d'arguments et de relations d'attaque deviennent important.

Exemple 1 (Examen médical simplifié). (suite) – Comme la représentation des traces d'événements et d'états est la même que dans la figure 3.4, nous ne commentons que les nouvelles relations causales obtenues dans la figure 3.7. Pour rappel, les arguments a , c et l sont les variables de décision. Comme l'argument a devient inacceptable très tôt dans le dialogue et le reste jusqu'à la fin, nous avons choisi de ne pas le représenter dans la figure. La figure 3.7 représente graphiquement les quatre derniers états des traces de l'exemple 1, correspondant à l'énonciation de l'argument n et aux mécanismes de mise à jour qui s'ensuivent. L'argument n , qui énonce l'urgence de l'examen, est celui qui clôt le débat.

On observe sur cette figure que son énonciation à la date 28 est une NESS-cause directe de son acceptabilité des dates 29 à 31, une relation que l'on note :

$$(enunciate_n, 28) dNc (a_n, 29 - 31).$$

De la même façon, on a :

$$\begin{aligned} & (\text{makesUnacc}_{n,c}, 29) \text{ dNc } (\neg a_c, 30 - 31); \\ & (\text{makesUnacc}_{n,m}, 29) \text{ dNc } (\neg a_m, 30 - 31); \\ & (\text{makesAcc}_l, 30) \text{ dNc } (a_l, 31). \end{aligned}$$

Comme le montrent ces exemples, la relation de NESS-cause directe s’intéresse aux effets réels du déclenchement d’un événement et constitue donc l’élément de base de la causalité. Pourtant, une fois de plus, cette relation n’est pas suffisante. Si nous voulons savoir pourquoi l’argument l est acceptable à la fin du dialogue (c’est-à-dire pourquoi la décision de passer une IRM le même jour est prise), il n’est pas satisfaisant de dire simplement que c’est à cause de l’événement $(\text{makesAcc}_l, 30)$, c’est-à-dire parce qu’il devient acceptable.

Pour savoir pourquoi cette dernière se produit, nous devons examiner les NESS-causes et les causes effectives afin de construire la chaîne causale qui y a conduit. Par transitivité, nous obtenons que $(\text{makesUnacc}_{n,m}, 29)$ est une cause effective du fait que makesAcc_l a été déclenché (noté $(\text{makesUnacc}_{n,m}, 29) \text{ cEff } (\text{makesAcc}_l, 30)$), et donc des effets que son déclenchement a pu avoir. En remontant encore plus loin et en recherchant les causes en raison desquelles l’occurrence $(\text{makesUnacc}_{n,m}, 29)$ a eu lieu, nous trouvons $(\text{enunciate}_n, 28) \text{ cEff } (\text{makesUnacc}_{n,m}, 29)$ et donc $(\text{enunciate}_n, 28)$ NESS-cause $(\neg a_m, 30 - 31)$. Par transitivité, on peut en déduire $(\text{enunciate}_n, 28)$ NESS-cause $(a_l, 31)$. Cette nouvelle relation permet de dire que le médecin qui énonce qu’il s’agit d’une urgence est l’une des causes de la décision finale, une réponse qui semble déjà plus satisfaisante et qui peut être incluse dans une explication. Le même raisonnement peut être appliqué pour trouver les causes de $(\neg a_c, 31)$, l’autre variable de décision.

3.5.3 Vers l’extraction d’explications adaptées à partir d’une chaîne causale

Ces chaînes causales, tout comme les représentations graphiques, peuvent ne pas représenter une explication en elles-mêmes. En effet, ces relations, aussi utiles soient-elles, ont tendance à former de longues chaînes causales comprenant des redondances ou des variables inutiles. Cela soulève la question de la manière appropriée de les utiliser pour déduire des explications adaptées. Dans cette section, nous discutons d’une méthode qui permette d’extraire, à partir de cet enrichissement des graphes argumentatifs, des explications sur l’acceptabilité ou l’inacceptabilité d’un ou plusieurs arguments. Comme présenté dans le chapitre 1, une première direction peut être trouvée dans les travaux qui étudient les liens entre les notions de causalité et d’explication. Ceux-ci sont résumés dans l’article de Miller (2019), qui établit plusieurs propriétés intéressantes qu’une explication doit satisfaire. Nous en abordons cinq : sa proximité avec la conséquence, sa prise en compte de la volition d’un agent, sa contrastivité, sa robustesse et sa longueur. La question est alors de savoir comment appliquer ces propriétés au nouveau cadre que nous proposons afin de fournir des explications adaptées à un utilisateur humain.

Pour appliquer les deux premiers principes, un langage d’action est particulièrement bien adapté. Tout d’abord, la proximité temporelle de l’explication avec la conséquence peut être évaluée et formalisée grâce à la formalisation du temps dans notre cadre en comparant leurs positions respectives dans les traces. Un autre intérêt est qu’une action délibérée est souvent préférée à une action qui ne l’est pas. Dans notre formalisme, cela

se traduit par le fait qu'une action est préférée à un événement exogène. Ainsi, face à deux explications identiques à l'exception d'un élément, celle dont les actions délibérées sont les plus proches temporellement de la conséquence à expliquer est préférée.

Enfin, une troisième caractéristique que doit respecter une explication concerne sa longueur : elle doit être courte. Il s'agit d'ailleurs d'une des raisons principales pour laquelle la chaîne causale ne constitue pas nécessairement une "bonne" explication et pour laquelle nous nous intéressons aux autres propriétés afin d'extraire un sous ensemble de taille réduit comme explication.

Le principe de robustesse a un caractère plus global car il s'applique à des explications candidates qui satisfont les propriétés précédentes. Une explication peut être considérée comme plus robuste qu'une autre si elle est une explication adaptée dans un plus grand nombre de scénarios. Il faut donc d'abord définir un ensemble de scénarios, c'est-à-dire un ensemble de séquence ς et comparer dans combien de ces dialogues différents chaque explication est valable.

Enfin, comme nous l'avons introduit dans le chapitre 1, une explication doit être contrastive. Il faut donc être capable de trouver les causes de x et $\neg y$ afin de pouvoir générer une explication causale contrastive adaptée. Comme l'a fait Miller pour définir une explication contrastive dans un modèle structurel causal de Halpern (Miller, 2021), nous devons introduire la définition formelle d'une cause contrastive. Intuitivement, les causes contrastives de a_l et $\neg a_c$ sont les éléments communs des deux chaînes causales, c'est-à-dire enu_n dans ce cas. Une fois la chaîne causale contrastive construite, on applique les principes précédents pour en extraire une explication contrastive.

Exemple 1 (Examen médical simplifié). *(suite) – En appliquant tous ces points, une explication pour faire une IRM aujourd'hui plutôt que dans deux jours pourrait être l'énonciation de n déclarant qu'il s'agit d'une urgence. En effet, il s'agit d'une explication courte (elle ne peut être plus courte qu'un fluent unique) et contrastive puisqu'elle appartient aux chaînes causales a_l et $\neg a_c$. De plus, il s'agit d'une action délibérée qui est temporellement proche de la conséquence. Enfin, dans le scénario modifié de l'exemple 1, enu_n est une explication au regard de tous les critères sans compter la robustesse. Par conséquent, enu_n est robuste pour au moins ces deux scénarios et il s'agit donc d'une explication satisfaisant les cinq propriétés.*

3.6 Conclusion

Dans ce chapitre, nous avons proposé une formalisation des systèmes d'argumentation abstraite acycliques dans le langage d'action de Sarmiento *et al.* (2022), en établissant ses propriétés formelles, en particulier sa terminaison, sa correction et sa complétude. Nous avons également montré que l'ajout de temporalité augmente l'expressivité d'un AAF, ce qui permet d'examiner l'effet de l'ordre d'énonciation des arguments. De plus, cette transformation permet d'exploiter la notion de causalité associée au langage d'action choisi, offrant la possibilité de fournir des informations riches sur les raisons de l'acceptation ou du rejet d'un argument. Nous avons également proposé deux types de représentations graphiques du processus d'argumentation qui peuvent être utilisées comme supports graphiques pour une explication. Enfin, nous avons discuté de la façon d'extraire des explications à partir des chaînes causales.

La suite de ce travail est d'étendre cette formalisation en supprimant l'hypothèse d'acyclicité. C'est ce que nous présentons dans le chapitre 4. Nous discutons d'une autre perspective dans le chapitre 6, à savoir de la formalisation et de l'évaluation empirique des différents critères mentionnés dans la section précédente.

Systèmes d'argumentation abstraite cycliques et langage d'action

Ce chapitre reprend les questions du chapitre précédent, à savoir la prise en compte de la temporalité en argumentation abstraite, mais s'affranchit de la contrainte d'acyclicité du graphe. Par conséquent, nous présentons une nouvelle formalisation d'un dialogue dans un langage d'action, adaptée à ce nouveau cadre. De plus, dans le cas d'un AAF cyclique, il n'existe plus une issue unique au dialogue, c'est-à-dire plus formellement une unique extension complète. De fait, en plus d'établir la correction et la terminaison de cette nouvelle transformation, nous discutons de deux définitions différentes pour la complétude. Enfin, nous proposons une modification basée sur une stratégie "dernier énoncé, dernier modifié" pour obtenir une transformation satisfaisant une des définitions de complétude. Il s'agit de la suite des travaux menés en collaboration avec Camilo Sarmiento. Le contenu de ce chapitre a fait l'objet d'une publication dans la conférence internationale PRIMA 2024 (Munro et al., 2024b).

Sommaire

4.1 Introduction	72
4.2 Passage des AAF au langage d'action	72
4.2.1 Instanciation du contexte	72
4.2.2 Modification de la sémantique du langage d'action	77
4.3 Étude formelle de la transformation proposée	78
4.3.1 Unicité des traces et cohérence des fluents	78
4.3.2 Terminaison et correction	78
4.4 Étude de la complétude	80
4.4.1 Deux définitions possibles pour la notion de complétude	80
4.4.2 Une nouvelle transformation $\mathcal{T}_{AF2AL}^{lf}$	80
4.4.3 Discussion sur les autres sémantiques	83
4.5 Application à l'exemple médical	84
4.6 Conclusion	85

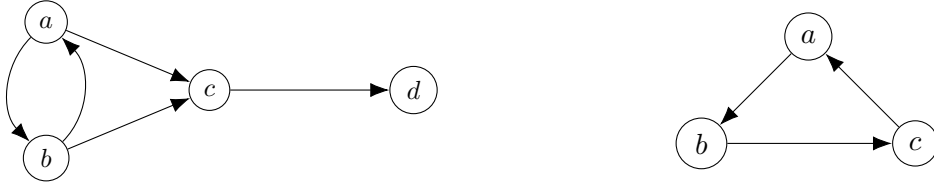


FIGURE 4.1 – Graphe d'argumentation associé à l'exemple 2 à gauche et à l'exemple 3 à droite.

4.1 Introduction

En argumentation abstraite, la présence de cycles dans le graphe peut entraîner la perte de l'unicité d'une extension complète unique qui coïncide avec les autres sémantiques présentées dans le chapitre 1. Or, la transformation $\mathcal{T}_{AFa2AL}$ définie dans le chapitre 3 ne renvoie qu'une issue unique, ce qui met à défaut la complétude de cette transformation. De plus, avec la modélisation par un langage d'action que nous avons proposée dans le chapitre précédent, ces cycles peuvent également créer des cycles de mises à jour empêchant d'atteindre un état argumentatif, nécessaire pour l'énonciation du prochain argument. Dans ce chapitre, nous proposons une modification de la transformation $\mathcal{T}_{AFa2AL}$, notée $\mathcal{T}_{AF2AL}$, pour permettre de répondre aux difficultés posées par la présence de cycles dans un système d'argumentation abstraite. La méthode de génération d'explications causales étant inchangée par rapport à celle du chapitre 3, nous n'en discutons pas dans ce chapitre.

Dans la section 4.2, nous proposons une nouvelle transformation permettant d'enrichir un AAF cyclique, ou non, avec l'ordre d'énonciation des arguments. Dans la section 4.3, nous établissons ses propriétés formelles : sa terminaison et sa correction. Nous discutons ensuite de la définition de complétude que nous souhaitons vérifier avec notre transformation et proposons une variante de cette dernière qui satisfait cette forme de complétude. Enfin, dans la section 4.5, nous appliquons les propositions de ce chapitre à l'exemple 1 dans sa version initiale, telle que décrite dans le chapitre 1.

4.2 Passage des AAF au langage d'action

Dans cette section, nous présentons la première contribution de ce chapitre : une nouvelle transformation dans le langage d'action présenté dans le chapitre 3 que nous notons $\mathcal{T}_{AF2AL}$. À nouveau, nous partons d'un couple $(\Delta, \mathcal{R})$ avec Δ un dialogue selon la définition 3.8 et $\mathcal{R}$ l'ensemble des relations d'attaque, que nous transformons en langage d'action.

4.2.1 Instanciation du contexte

Comme dans le chapitre précédent, nous décomposons la transformation en trois points : les fluents, les événements et les règles de priorité. De plus, nous considérons maintenant des graphes d'argumentation qui peuvent contenir des cycles. Nous utilisons donc les exemples 2 et 3, qui présentent des cycles, pour illustrer cette transformation. La figure 4.1 rappelle pour cela les graphes associés.

4.2.1.1 Description du monde : les fluents

Définissons d'abord les variables nécessaires pour décrire le monde, c'est-à-dire l'AAF. Ces variables correspondent aux fluents $\mathbb{F}$ dans le langage d'action. Comme introduit dans le chapitre 1, il y a deux éléments à considérer : les arguments et la relation d'attaque. Pour décrire un argument x , nous créons trois fluents. Tout d'abord, comme dans le chapitre 3, $p_x \in \mathbb{F}$ représente le fait que x est présent dans le graphe. Ensuite, contrairement au chapitre précédent, nous considérons ici trois valeurs d'acceptabilité différentes qui correspondent aux trois étiquetages présentés dans le chapitre 1 : IN, OUT et UNDEC. Ce choix est justifié par la présence potentielle de cycles dans le graphe d'argumentation, qui peuvent produire des situations indéterminées. Pour obtenir ces trois valeurs d'acceptabilité, nous proposons d'utiliser deux fluents : i_x (respectivement o_x) représente le fait que x est in (respectivement out). Leur combinaison permet d'obtenir les trois différentes valeurs d'acceptabilité :

- $i_x \wedge \neg o_x$ correspondant à l'état acceptable ou IN pour les étiquetages ;
- $\neg i_x \wedge o_x$ correspondant à l'état inacceptable ou OUT ;
- $\neg i_x \wedge \neg o_x$ correspondant à l'état indéterminé ou UNDEC.

La quatrième combinaison $i_x \wedge o_x$ est considérée comme impossible. Une autre option aurait pu être de créer directement trois fluents a_x, na_x et $undec_x$ correspondant aux trois étiquetages. Néanmoins, pour traiter correctement les cycles, nous voulons séparer autant que possible les mises à jour de l'état d'acceptabilité. Par exemple, lorsqu'un argument acceptable est attaqué par un autre argument acceptable ou indéterminé, il devient indéterminé. Dans notre formalisme, cela signifie simplement qu'il est attaqué par un argument qui n'est pas OUT. De plus, comme la quatrième combinaison est impossible, x est acceptable équivaut à i_x , et x est inacceptable équivaut à o_x . Par conséquent, la création de ces trois fluents n'est pas utile avec cette décomposition.

Le dernier élément nécessaire pour modéliser le dialogue est la relation d'attaque $\mathcal{R}$ entre les arguments : nous utilisons le fluent $cA_{y,x} \in \mathbb{F}$ pour modéliser le fait que l'argument y peut attaquer l'argument x . Tout comme dans le chapitre précédent, on ne garde ici que la notion d'attaque potentielle car la différence avec une attaque effective disparaît dans les équations.

Exemple 2. (suite) - En appliquant cette première étape à l'exemple 2 décrit par le graphe de gauche sur la figure 4.1, on obtient que $\mathbb{F} = \mathbb{F}_{\mathcal{A}} \cup \mathbb{F}_{\mathcal{R}}$ où :

- $\mathbb{F}_{\mathcal{A}} = \{p_a, i_a, o_a, p_b, i_b, o_b, p_c, i_c, o_c, p_d, i_d, o_d\}$
- $\mathbb{F}_{\mathcal{R}} = \{cA_{a,b}, cA_{b,a}, cA_{a,c}, cA_{b,c}, cA_{c,d}\}$

4.2.1.2 Évolution du monde : les événements

Comme dans le chapitre précédent, nous faisons le choix de ne considérer que l'énonciation d'un argument comme seule action volontaire, c'est-à-dire $\mathbb{A} = \{enunciate_x \mid x \in A\}$. Quand $enunciate_x$ est effectuée, x devient présent et acceptable par défaut.

Contrairement au chapitre 3, ici répéter un argument déjà présent a un intérêt. En effet, énoncer à nouveau x a pour effet de réinitialiser sa valeur d'acceptabilité pour que x soit acceptable. Il est ainsi possible de répéter un argument précédemment inacceptable ou indéterminé afin de reconsidérer son statut. Dans le cas d'un graphe acyclique, cela ne change rien car il n'y a qu'une seule issue finale au dialogue correspondant à

l'unique ensemble admissible maximal. Cela n'est plus vrai si des cycles sont présents. Pour cette raison, nous n'imposons plus que l'argument énoncé ne soit pas déjà présent.

Formellement :

$$pre(enunciate_x) \equiv \top; \quad eff(enunciate_x) \equiv p_x \wedge i_x \wedge \neg o_x$$

Exemple 2. (suite) Si nous poursuivons la transformation sur cet exemple, nous obtenons que $\mathbb{A} = \{enunciate_a, enunciate_b, enunciate_c, enunciate_d\}$.

Après l'énonciation d'un argument, les statuts d'acceptabilité de tous les arguments présents sont récursivement mis à jour selon des règles qui dépendent de la sémantique considérée Σ (voir section 1.1.2).

Une question importante est celle de la terminaison de ce processus récursif, c'est-à-dire l'atteinte d'un état dans lequel aucune règle de mise à jour de l'acceptabilité ne peut plus être déclenchée. Nous appelons un tel état un *état Σ -argumentatif* et sa définition dépend de la sémantique considérée. Dans cette section, nous nous concentrons sur la sémantique complète Σ_c , et discutons d'autres sémantiques avec les difficultés techniques qu'elles soulèvent dans la section 4.3. Nous exploitons la caractérisation de l'étiquetage complet établie par Caminada & Pigozzi (2011) : un étiquetage est complet si et seulement si pour un argument x , x est étiqueté IN si et seulement si tous ses attaquants sont étiquetés OUT, et il est étiqueté OUT si et seulement si il a au moins un attaquant qui est étiqueté IN. Ce principe nous conduit à la définition suivante :

Définition 4.1 (État Σ_c -argumentatif). *Un état $S(t)$ est un état Σ_c -argumentatif si :*

- i) $\forall x, [S(t) \models p_x \wedge i_x \wedge \neg o_x \Leftrightarrow S(t) \models p_x \wedge (\bigwedge_y (cA_{y,x} \wedge \neg i_y \wedge o_y) \vee \neg cA_{y,x})]$;
- ii) $\forall x, [S(t) \models p_x \wedge \neg i_x \wedge o_x \Leftrightarrow \exists y, S(t) \models p_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y]$.

Pour atteindre un état Σ_c -argumentatif après l'énonciation d'un argument, des mises à jour de l'acceptabilité sont déclenchées automatiquement. Pour rappel, en argumentation abstraite, un argument n'est acceptable que s'il n'est pas attaqué ou s'il n'est attaqué que par des arguments inacceptables. Il suffit également que l'un des attaquants soit acceptable pour que l'argument attaqué soit inacceptable. Enfin, un argument qui n'a pas d'attaquants acceptables et au moins un attaquant qui n'est pas inacceptable est indéterminé. Comme il peut y avoir des cycles dans le graphe d'argumentation, nous devons diviser ce mécanisme de mise à jour autant que possible. Pour cela, nous utilisons quatre types de mises à jour, représentés par quatre événements exogènes modélisant respectivement les transitions : d'acceptable à indéterminé ($\Delta_{y,x}^1$), d'indéterminé à inacceptable ($\Delta_{y,x}^2$), d'inacceptable à indéterminé (I_x^1) et d'indéterminé à acceptable (I_x^2), c'est-à-dire $\mathbb{U} = \{\Delta_{y,x}^1, \Delta_{y,x}^2, I_x^1, I_x^2\}$. Cette décomposition en deux étapes du passage direct d'acceptable à inacceptable et vice-versa est un élément crucial de la transformation que nous proposons. En effet, elle permet d'éviter de tourner en rond à cause des cycles et est donc un élément fondamental pour assurer la propriété de terminaison.

Remarque – Nous avons conscience que le fluent $cA_{y,x}$ n'est pas utile dans le point i) de la définition 4.1. Nous décidons toutefois de le mettre afin que les conditions d'un état Σ_c -argumentatif soient plus lisibles.

4.2.1.3 Passage d'acceptable à indéterminé ($\Delta_{y,x}^1$)

Nous étudions d'abord la règle qui conduit un argument acceptable à devenir indéterminé. Supposons qu'un argument y puisse attaquer l'argument x , et que y ne soit pas inacceptable, c'est-à-dire indéterminé ou acceptable et que x soit acceptable. Alors, x étant attaqué par un argument y qui n'est pas inacceptable, il ne peut plus être acceptable. Formellement, l'événement exogène $\Delta_{y,x}^1$ s'écrit comme suit :

$$\begin{aligned} tri(\Delta_{y,x}^1) &\equiv p_x \wedge i_x \wedge p_y \wedge cA_{y,x} \wedge \neg o_y; \\ eff(\Delta_{y,x}^1) &\equiv \neg i_x \end{aligned}$$

Dans la précondition, il suffit d'être *in* pour déclencher cet événement, car le fait d'être *in* équivaut à être acceptable. En effet, il est interdit d'être à la fois *in* et *out*. Par conséquent, la seule combinaison de fluents contenant i_x pour un argument x est $i_x \wedge \neg o_x$. De la même façon, être *out* est équivalent à être inacceptable. Par conséquent, comme x est attaqué par un argument y non inacceptable ($\neg o_y$), c'est-à-dire un argument acceptable ou indéterminé, cela conduit nécessairement x à devenir non acceptable ($\neg i_x$), c'est-à-dire que x est indéterminé ou inacceptable.

4.2.1.4 Passage d'indéterminé à inacceptable ($\Delta_{y,x}^2$)

À présent, si un argument acceptable y attaque un argument indéterminé x , alors x devient inacceptable. Cela conduit au second événement exogène $\Delta_{y,x}^2$:

$$\begin{aligned} tri(\Delta_{y,x}^2) &\equiv p_x \wedge \neg o_x \wedge p_y \wedge cA_{y,x} \wedge i_y; \\ eff(\Delta_{y,x}^2) &\equiv o_x \end{aligned}$$

4.2.1.5 Passage d'inacceptable à indéterminé (I_x^1)

Supposons maintenant que l'argument x soit inacceptable et qu'en raison de certaines circonstances, tous ses attaquants soient devenus non acceptables, c'est-à-dire *not in*. Alors, x ne peut plus être inacceptable et devient non inacceptable, c'est-à-dire *not out*. En particulier, x devient indéterminé. Cela est représenté par l'événement I_x^1 :

$$\begin{aligned} tri(I_x^1) &\equiv p_x \wedge o_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y) \vee \neg cA_{y,x} \right); \\ eff(I_x^1) &\equiv \neg o_x \end{aligned}$$

4.2.1.6 Passage d'indéterminé à acceptable (I_x^2)

Enfin, si l'argument x est indéterminé et n'est attaqué que par des arguments inacceptables, alors x devient acceptable. Cela se traduit dans le langage d'action par l'événement I_x^2 :

$$\begin{aligned} tri(I_x^2) &\equiv p_x \wedge \neg i_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge o_y) \vee \neg cA_{y,x} \right); \\ eff(I_x^2) &\equiv i_x \end{aligned}$$

Avec ces évènements exogènes, nous avons décomposé le processus consistant à rendre les arguments acceptables ou inacceptables en deux étapes en utilisant le statut indéterminé. Cela est nécessaire pour la terminaison lorsqu'il faut traiter des cycles afin d'éviter de tourner en rond. Par exemple, dans le cas d'un cycle à deux éléments, les deux arguments deviennent tout d'abord indéterminés, puis les mises à jour s'arrêtent, plutôt que de devenir simultanément acceptables, puis inacceptables, puis acceptables, et ainsi de suite. Dans le cas des graphes acycliques, il aurait été possible de fusionner les deux règles de chaque type. Un argument x ne peut alors plus être dans un état indéterminé. C'est ce qui est fait dans la transformation $\mathcal{T}_{AFa2AL}$.

Exemple 2. (suite) – L'énumération de tous les évènements exogènes pour cet exemple serait trop longue : il y a deux évènements I par argument et pour chaque argument il y a six évènements Δ , soit un total de 32 évènements exogènes. Nous donnons donc seulement l'exemple de l'argument a pour lequel nous avons $\mathbb{U} = \{I_a^1, I_a^2, \Delta_{b,a}^1, \Delta_{b,a}^2, \Delta_{c,a}^1, \Delta_{c,a}^2, \Delta_{d,a}^1, \Delta_{d,a}^2\}$.

4.2.1.7 Cycles et règles de priorité

La décomposition au moyen de l'état indéterminé nécessite la définition de règles de priorité tenant compte de deux éléments : empêcher l'état interdit pour l'acceptabilité et garantir la terminaison.

D'une part, les règles décomposées (I^1 et I^2 , Δ^1 et Δ^2) doivent être déclenchées dans le bon ordre. Cela permet d'éviter qu'un argument x soit à la fois *in* et *out*, ce qui est interdit. Pour cette raison, nous donnons la priorité aux règles numérotées 1 qui conduisent à l'état indéterminé avant de lever éventuellement l'indétermination. Cela revient à se concentrer d'abord sur les arguments acceptables et inacceptables avant de s'occuper des arguments indéterminés. Cela est exprimé par les deux règles : $R_1 := \forall x, I_x^1 \succ I_x^2$ et $R'_1 := \forall x, y, \Delta_{y,x}^1 \succ \Delta_{y,x}^2$.

Exemple 2. (suite) – Considérons la transformation sans les règles de priorité décrites ci-dessus. Supposons que nous nous trouvons dans un état $S(t_0)$ dans lequel seul a est présent et acceptable. À cet instant, l'argument b est énoncé, ce qui conduit à l'état suivant $S(t_0 + 1)$. Puisque $S(t_0 + 1) \models i_a \wedge \neg o_a \wedge i_b \wedge \neg o_b \wedge cA_{a,b} \wedge cA_{b,a}$, $\Delta_{a,b}^1, \Delta_{a,b}^2$ et $\Delta_{b,a}^1, \Delta_{b,a}^2$ sont déclenchées. On obtient que $S(t_0 + 2) \models \neg i_a \wedge o_a \wedge \neg i_b \wedge o_b$. Par conséquent, $I_a^1, I_a^2, I_b^1, I_b^2$ sont déclenchées, ce qui conduit à l'état $S(t_0 + 3) \models i_a \wedge \neg o_a \wedge i_b \wedge \neg o_b$. $\Delta_{a,b}^1, \Delta_{a,b}^2$ et $\Delta_{b,a}^1, \Delta_{b,a}^2$ sont à nouveau déclenchées puis $I_a^1, I_a^2, I_b^1, I_b^2$ encore une fois etc. Dans ce scénario, une boucle infinie apparaît. Les deux règles R_1 et R'_1 permettent de remédier à ce problème.

En effet, revenons à l'état $S(t_0 + 1)$. À nouveau, $S(t_0 + 1) \models i_a \wedge \neg o_a \wedge i_b \wedge \neg o_b$. $\Delta_{a,b}^1, \Delta_{a,b}^2$ et $\Delta_{b,a}^1, \Delta_{b,a}^2$ peuvent donc se déclencher. Cependant, avec la règle R'_1 , seuls $\Delta_{a,b}^1$ et $\Delta_{b,a}^1$ ont effectivement lieu. On obtient un état $S(t_0 + 2)$ tel que $S(t_0 + 2) \models \neg i_a \wedge \neg o_a \wedge \neg i_b \wedge \neg o_b$, c'est-à-dire que a et b sont indéterminés. Plus aucune mise à jour ne pouvant être déclenchée, nous avons atteint un état Σ_c -argumentatif.

D'autre part, nous voulons garantir la terminaison de notre transformation même en présence de cycles. Pour ce faire, nous utilisons les règles de priorité suivantes : $\forall x_1, y_1, x_2, y_2, R_2 := I_{x_1}^1 \succ \Delta_{y_2, x_2}^2$ et $R_3 := \Delta_{y_1, x_1}^1 \succ \Delta_{y_2, x_2}^2$. Bien que ces règles soient plus fortes que les précédentes, le principe qui les sous-tend est le même : se concentrer sur les arguments acceptables et inacceptables avant de traiter les arguments indétermi-

nés. Nous allons illustrer cela dans l'exemple suivant. Pour les détails techniques, nous invitons le lecteur à se référer à la preuve de terminaison en annexe C.

On peut noter que la dernière règle R_3 inclut la deuxième règle R'_1 présentée : R_3 est une version plus forte nécessaire pour traiter les cycles alors que R'_1 peut plutôt être considérée comme une règle argumentative. Par exemple, dans le cas des graphes d'argumentation acycliques, seule R'_1 est nécessaire.

Exemple 3. (suite) – Prenons maintenant l'exemple 3 dont l'AAF est représenté à droite de la figure 4.1. On se place dans le cadre suivant : seules les règles R_1 et R'_1 existent et l'état actuel du monde est $S(t_0) \models p_a \wedge p_b \wedge i_a \wedge \neg o_a \wedge \neg i_b \wedge o_b$. Cet état est obtenu en énonçant dans n'importe quel ordre les arguments a et b car, à ce stade, le sous-graphe induit par a et b est acyclique.

L'argument c est alors énoncé. On obtient l'état $S(t_1)$ tel que $S(t_1) \models p_c \wedge i_c \wedge \neg o_c \wedge p_a \wedge i_a \wedge \neg o_a \wedge cA_{c,a}$ c'est-à-dire $S(t_1) \models \text{tri}(\Delta_{c,a}^1) \wedge \text{tri}(\Delta_{c,a}^2)$. Avec la règle R'_1 , seule la règle 1 se déclenche et donc a devient indéterminé à l'état suivant $S(t_1 + 1)$.

Ensuite, on a simultanément $S(t_1 + 1) \models p_c \wedge p_a \wedge i_c \wedge \neg o_a \wedge cA_{c,a}$ c'est-à-dire $\text{tri}(\Delta_{c,a}^2)$ et $S(t_1 + 1) \models p_b \wedge o_b \wedge \left(\bigwedge_y (cA_{y,b} \wedge \neg i_y) \vee \neg cA_{y,b} \right)$ c'est-à-dire $\text{tri}(I_b^1)$. Sans la règle R_2 , les deux événements se déclenchent conduisant à l'état suivant $S(t_1 + 2)$ dans lequel a est inacceptable, b est indéterminé et c est acceptable.

Deux nouveaux événements exogènes se déclenchent alors : b devient acceptable (I_b^2) et c devient indéterminé ($\Delta_{b,c}^1$). Dans l'état suivant, comme a est inacceptable et c indéterminé alors a devient indéterminé (I_a^1). De plus, comme b est acceptable alors c devient inacceptable. Puis, a devient acceptable et b indéterminé, b inacceptable et c indéterminé, c acceptable et a indéterminé etc. Dans le cas d'un cycle de longueur impaire et sans la règle R_2 on peut donc tourner en rond.

Reprenons maintenant avec la règle R_2 à l'état $S(t_1 + 1)$. Seul l'évènement I_b^1 se déclenche. On arrive dans un état $S(t_2)$ où a et b sont indéterminés et c est acceptable. Deux événements peuvent donc se produire : a devient inacceptable ($\Delta_{c,a}^2$) et c devient indéterminé ($\Delta_{b,c}^1$). Avec la règle R_3 , on atteint un état où les trois arguments sont indéterminés, état où plus aucune règle de mise à jour ne se déclenche.

Sans cette règle, a est inacceptable et b et c indéterminés. Ensuite, b devient acceptable (I_b^2) et a devient indéterminé (I_a^1) à l'état suivant. On a donc b acceptable et a et c indéterminés. L'état ainsi obtenu est exactement le même que $S(t_2)$ où seul le nom des arguments change. On a donc à nouveau une boucle infinie qui apparaît sans la règle R_3 .

Dans cette section, nous avons introduit une nouvelle transformation $\mathcal{T}_{AF2AL}$ définie par son contexte κ adapté à un système d'argumentation abstraite cyclique. En particulier, nous avons défini les ensembles suivants :

- les fluents $\mathbb{F} = \{p_x, i_x, o_x \mid x \in \mathcal{A}\} \cup \{cA_{y,x} \mid (y, x) \in \mathcal{R}\}$;
- les événements $\mathbb{E} = \{\text{enunciate}_x \mid x \in \mathcal{A}\} \cup \{\Delta_{y,x}^1, \Delta_{y,x}^2, I_x^1, I_x^2 \mid x, y \in \mathcal{A}\}$ avec $\text{pre}, \text{tri}, \text{eff}$ et $\succ_{\mathbb{E}}$ définis ci-dessus.

4.2.2 Modification de la sémantique du langage d'action

Comme dans le chapitre 3, il faut modifier les définitions associées à la sémantique du langage d'action afin de l'adapter au cadre de l'argumentation abstraite. Contrairement au contexte κ qui décrit le fonctionnement des éléments constituant le langage

d'action, par exemple les fluents ou les événements, la sémantique décrit le système de transition en définissant par exemple quelles sont les transitions possibles à partir d'un état donné. De fait, la sémantique d'un langage d'action est en quelque sorte indépendante du contenu de ses états ainsi que des événements possibles. Modifier le contexte κ du chapitre 3 pour l'adapter au cas des systèmes d'argumentation abstraite cycliques ne modifie pas le fonctionnement d'un dialogue d'un point de vue global. Pour cette raison, nous n'avons pas besoin dans ce chapitre d'adapter les modifications de la sémantique que nous faisons dans le chapitre 3, c'est-à-dire que nous gardons dans ce chapitre les définitions 3.12 et 3.13 d'une exécution valide et des traces données dans la section 3.3.2.

4.3 Étude formelle de la transformation proposée

Dans cette section, nous commençons par établir et discuter certaines propriétés formelles de la transformation proposée $\mathcal{T}_{AF2AL}$ dans la section 4.2, notamment la correction et la terminaison de cette transformation. Les preuves détaillées sont données dans l'annexe C. Puis, nous discutons de deux formes de complétude. Nous proposons ensuite d'explorer une modification de la transformation dans le cadre d'une nouvelle stratégie conduisant à l'une de ces formes de complétude. Enfin, nous discutons des autres sémantiques que la sémantique complète.

4.3.1 Unicité des traces et cohérence des fluents

La propriété d'unicité des traces (proposition 3.3) n'étant pas dépendante de κ mais uniquement de la sémantique modifiée du langage d'action, elle reste valide pour cette nouvelle transformation. En particulier, cela implique que lorsqu'il est fait référence à des événements et à des états, il s'agit des événements et des états des uniques traces τ_χ^e et τ_χ^s , respectivement. Ainsi, l'ensemble de tous les événements qui se produisent effectivement au moment t est $E^\chi(t) = \tau_\chi^e(t)$. Selon le même principe, l'état effectif à l'instant t est $S^\chi(t) = \tau_\chi^s(t)$.

Lors de la définition des fluents, nous avons précisé que *in* et *out* étaient impossibles simultanément pour un argument. Avec cette transformation, il n'y a pas d'état dans la trace où un argument est à la fois *in* et *out*, ce qui montre la pertinence des règles proposées à cet égard.

Proposition 4.1 (Cohérence des fluents *i* et *o*). $\forall t, \forall x, S^\chi(t) \not\models i_x \wedge o_x$.

Il s'agit d'une proposition importante car elle établit une caractérisation du statut acceptable et inacceptable. En effet, si un argument est *in*, il est nécessairement *not out*, donc être *in* est équivalent à être acceptable, c'est-à-dire être IN, et il en va de même pour *out* et être inacceptable, c'est-à-dire OUT.

4.3.2 Terminaison et correction

Pour établir la terminaison et la correction de la transformation $\mathcal{T}_{AF2AL}$ que nous proposons, nous introduisons une caractérisation d'un état Σ_c -argumentatif comme un état où aucun événement exogène n'est déclenché : un tel état peut être considéré comme à l'équilibre jusqu'à ce qu'une action délibérée, c'est-à-dire l'énonciation d'un nouvel argument, se produise. Une autre façon de le dire est qu'un état argumentatif correspond

à une étape réelle du dialogue tandis que les états non argumentatifs sont des étapes intermédiaires utilisées pour modéliser le processus de mise à jour du dialogue.

Lemme 4.1 (Caractérisation d'un état Σ_c -argumentatif). *Soit $S^x(t)$ un état. Les deux propriétés suivantes sont équivalentes :*

- (i) $S^x(t)$ est un état Σ_c -argumentatif selon la définition 4.1.
- (ii) $\forall e \in \mathbb{U}, S^x(t) \not\models tri(e)$.

Nous établissons maintenant la terminaison et la correction de $\mathcal{T}_{AF2AL}$. Tout d'abord, comme un état Σ_c -argumentatif est un état où rien ne se passe à moins qu'un argument ne soit énoncé et que $S(0)$ est un état Σ_c -argumentatif, alors pour prouver la terminaison, nous prouvons qu'un tel état est toujours atteint en un nombre fini d'étapes. Cela est établi dans le théorème 4.1.

Théorème 4.1 (Terminaison). *Soit $S^x(t)$ un état Σ_c -argumentatif et $x \in \mathcal{A}$ un argument. Si $enunciate_x \in E^x(t)$, alors $\exists t' > t \in \mathbb{T}$ tel que $S^x(t')$ est un état Σ_c -argumentatif.*

Principe de la preuve. Pour prouver ce théorème, nous commençons par montrer qu'après avoir énoncé un argument à partir d'un état Σ_c -complet, les seuls événements exogènes qui peuvent se déclencher sont du type Δ^1 . Cela est permis grâce à la règle R'_1 . Ensuite, à partir de ce nouvel état, à cause de R_2 , les seuls événements exogènes qui peuvent se déclencher sont du type I^1 . Encore une fois si cela est possible par application de la règle R_3 , les seuls événements exogènes qui sont déclenchés sont Δ^1 . Comme ces deux événements ne sont pas contradictoires, cela dure un nombre fini de fois. Suivant le même principe, il existe une séquence finie d'événements exogènes Δ^2 puis I^2 jusqu'à ce que plus rien ne puisse se déclencher. En utilisant le lemme 4.1, nous pouvons conclure. $\square$

Ensuite, nous établissons la correction de $\mathcal{T}_{AF2AL}$. Pour cela, nous définissons une notion d'étiquetage associé $\mathcal{L}_t$ pour tout $t \in \mathbb{T}$:

Définition 4.2. *Soit $S^x(t)$ un état. L'étiquetage associé sur $Ar = \{x \in \mathcal{A} \mid S^x(t) \models p_x\}$ est la fonction $\mathcal{L}_t$ telle que :*

- (i) $in(\mathcal{L}_t) = \{x \in Ar \mid S^x(t) \models i_x \wedge \neg o_x\}$,
- (ii) $out(\mathcal{L}_t) = \{x \in Ar \mid S^x(t) \models \neg i_x \wedge o_x\}$,
- (iii) $undec(\mathcal{L}_t) = \{x \in Ar \mid S^x(t) \models \neg i_x \wedge \neg o_x\}$.

Comme la définition d'un état Σ_c -argumentatif a été construite à partir d'une caractérisation d'un étiquetage complet (Caminada, 2006), nous établissons la propriété de correction dans le théorème suivant :

Théorème 4.2 (Correction). *Soit $S^x(t)$ un état Σ_c -argumentatif et $\mathcal{L}_t$ son étiquetage associé.*

Alors $\mathcal{L}_t$ est un étiquetage complet du système d'argumentation abstraite $AF = (Ar, \mathcal{R})$ avec $Ar = \{x \in \mathcal{A} \mid S^x(t) \models p_x\}$ et $\mathcal{R} = \{(y, x) \mid S^x(t) \models p_x \wedge p_y \wedge cA_{y,x}\}$.

Le système d'argumentation abstraite AF défini dans le théorème 4.2 est appelé AAF associé à $S^x(t)$.

Principe de la preuve. La définition d'un état Σ_c -argumentatif ayant été construite à partir d'une caractérisation des étiquetages complets, on vérifie de façon assez directe que x est IN si et seulement si tous ses attaquants sont OUT et x est OUT si et seulement si il existe un argument y qui est IN. $\square$

Le théorème 4.1, en combinaison avec la proposition 3.3, établit qu'étant donné un ordre d'énonciation, il existe un unique état final et que cet état est un état Σ_c -argumentatif. Ensuite, en utilisant le théorème 4.2, l'étiquetage associé à cet état est unique et donne donc une extension complète unique pour le système d'argumentation associé, c'est-à-dire pour le dialogue en entier. Par conséquent, modéliser un dialogue avec le formalisme que nous proposons au lieu d'un système d'argumentation abstraite permet, et ce grâce à l'ordre d'énonciation, de produire une extension complète unique, c'est-à-dire une issue unique pour le dialogue. Il s'agit d'une différence fondamentale avec l'argumentation qui sélectionne les arguments acceptés parmi les extensions en utilisant des méthodes d'inférence sceptique et crédule par exemple (Baroni *et al.*, 2018).

4.4 Étude de la complétude

4.4.1 Deux définitions possibles pour la notion de complétude

Une dernière propriété formelle importante à étudier est la notion de complétude de la transformation. Une première définition de cette notion est, étant donné un dialogue Δ et une sémantique argumentative Σ , de pouvoir trouver toutes les extensions de cette sémantique. Cela signifie qu'il n'y a pas de perte d'information lorsque l'on utilise le formalisme que nous proposons pour modéliser le dialogue au lieu d'un système d'argumentation abstraite. De plus, grâce aux traces, nous gagnons même de l'information car celles-ci permettent de comprendre comment un tel état a été atteint, et même d'en trouver les causes (voir section 3.5.2). Le problème de cette exhaustivité est qu'elle efface la plupart des avantages de la modélisation de l'ordre d'énonciation des arguments. En effet, si quel que soit l'ordre nous devons trouver toutes les extensions, cette forme de temporalité n'a d'importance que pour les traces. Nous voulons aller plus loin et utiliser cette temporalité pour différencier les extensions.

Nous proposons donc de considérer une autre forme de complétude : pour chaque extension d'une sémantique donnée, il existe un ordre d'énonciation tel que l'étiquetage associé à son état final est égal à cette extension : en utilisant cette définition, nous pouvons trouver toutes les extensions et au moins un ordre d'énonciation qui conduit à une telle extension. Ainsi, si un agent veut parvenir à une certaine décision, c'est-à-dire à une extension particulière, la recherche d'un ordre qui y mène peut être considérée comme une stratégie de débat. C'est ce type de complétude que nous recherchons.

4.4.2 Une nouvelle transformation $\mathcal{T}_{AF2AL}^{lf}$

Avec la transformation $\mathcal{T}_{AF2AL}$ présentée dans la section 4.2.1, la seconde forme de complétude n'est pas vérifiée. En effet, pour traiter les graphes avec des cycles de longueur impaire, les deux règles de priorité R_2 et R_3 conduisent à mettre le statut d'acceptabilité des arguments à indéterminés avant d'appliquer les autres règles. Pour cette

raison, dans le cas d'un cycle de longueur paire, le seul état final qui peut être atteint est celui où tous les arguments sont indéterminés.

Exemple 2. (suite) – Revenons sur cet exemple et imaginons que l'argument a a déjà été énoncé. Nous énonçons maintenant l'argument b . Avec la transformation initiale, $\Delta_{a,b}^1$ et $\Delta_{b,a}^1$ se déclenchent et on atteint donc un état stable où les deux arguments sont indéterminés. En répétant alors par exemple b , c'est à nouveau $\Delta_{a,b}^1$ qui se déclenche à cause de la règle R_3 et donc il est impossible d'atteindre un état final où par exemple a est inacceptable et b est acceptable, alors qu'il existe une extension complète correspondante.

Pour résoudre ce problème, nous explorons ici une autre stratégie que nous appelons la stratégie du "dernier énoncé, dernier mis à jour" dans laquelle un argument venant d'être énoncé est protégé par un "bouclier" et donc son statut d'acceptabilité ne peut être modifié tant que des mises à jour pour des arguments sans bouclier se produisent. Par exemple, considérons le cas où un argument x qui est attaqué et qui attaque également d'autres arguments est énoncé. Sans autre hypothèse, l'événement Δ^1 peut être déclenché et changer le statut d'acceptabilité de x en indéterminé. Avec la stratégie "dernier énoncé, dernier mis à jour", nous appliquons d'abord tous les effets des arguments énoncés, puis nous mettons à jour leur propre statut, ce qui déclenche finalement une autre chaîne d'événements exogènes. Ce faisant, tous les attaquants de x peuvent devenir inacceptables et l'état de x peut donc rester inchangé.

Exemple 2. (suite) – L'idée de la nouvelle transformation est donc de dire que si b vient d'être énoncé, nous allons d'abord regarder les effets de cette action sur les autres arguments en "protégeant" le statut de b . De fait, $\Delta_{b,a}^1$ se déclenche, puis à l'état suivant $\Delta_{b,a}^2$ se déclenche aussi. Ensuite, plus rien ne peut se produire et donc on enlève le bouclier de b et on regarde ce qu'il se passe. Dans cet exemple aucune action n'est déclenchée, et donc b est acceptable et a inacceptable. À noter ici que si a est alors répété, b devient inacceptable et a acceptable.

Pour formaliser cette stratégie, le contexte κ doit être modifié. Tout d'abord, nous avons besoin d'un nouveau fluent l_x qui est défini comme vrai lorsqu'un argument vient d'être énoncé, représentant le bouclier de l'argument x . Ce fluent signifie que cet argument doit être mis à jour en dernier. Ensuite, l'action $enunciate_x$ devient :

$$pre(enunciate_x) \equiv \top; \quad eff(enunciate_x) \equiv p_x \wedge l_x \wedge i_x \wedge \neg o_x.$$

Ensuite, pour que la mise à jour de l'acceptabilité d'un argument venant d'être énoncé se produise en dernier, nous créons deux instances de chaque événement exogène, par exemple $\Delta_{y,x}^1$ est remplacé par $\Delta_{y,x}^{1f}$ et $\Delta_{y,x}^{1l}$ tel que :

$$\begin{aligned} tri(\Delta_{y,x}^{1f}) &\equiv p_x \wedge \neg l_x \wedge i_x \wedge p_y \wedge cA_{y,x} \wedge \neg o_y; & eff(\Delta_{y,x}^{1f}) &\equiv \neg i_x \\ tri(\Delta_{y,x}^{1l}) &\equiv p_x \wedge l_x \wedge i_x \wedge p_y \wedge cA_{y,x} \wedge \neg o_y; & eff(\Delta_{y,x}^{1l}) &\equiv \neg i_x \end{aligned}$$

Nous créons donc deux événements identiques dans leur effet mais qui s'appliquent dans le cas f aux arguments déjà présents dans le graphe et dans le cas l , aux arguments venant d'être énoncés. Nous créons de même six événements exogènes à partir des trois autres événements exogènes, trois pour les arguments les plus anciens et trois pour les plus récents.

Une fois que toutes les mises à jour ont eu lieu, les arguments venant d'être énoncés doivent perdre leur protection et donc devenir $\neg l_x$. Pour cela, nous introduisons l'évènement exogène suivant :

$$tri(lastToFirst_x) = l_x; \text{eff}(lastToFirst_x) = \neg l_x$$

L'ensemble des évènements exogènes $lastToFirst$ est noté $\mathbb{U}^{l2f}$. Nous définissons :

$$\mathbb{U} = \mathbb{U}^f \cup \mathbb{U}^l \cup \mathbb{U}^{l2f}.$$

Enfin, la dernière étape consiste à mettre à jour les règles de priorité en fonction des nouveaux évènements exogènes. Chaque règle est dédoublée, et nous passons donc de quatre à huit règles de priorité. De plus, pour forcer le modèle à mettre à jour les derniers arguments lorsque rien d'autre ne peut se produire, les règles suivantes sont nécessaires : $\forall e \in \mathbb{U}^f, \forall e' \in \mathbb{U}^l, e \succ e'$. Enfin, nous ajoutons une dernière règle pour garantir que lorsque plus aucune mise à jour du degré d'acceptabilité des arguments, nouveaux comme anciens, n'est possible, alors tous les nouveaux arguments deviennent anciens : $\forall e \in \mathbb{U}^f \cup \mathbb{U}^l, \forall e' \in \mathbb{U}^{l2f}, e \succ e'$.

Avec cette nouvelle transformation nommée $\mathcal{T}_{AF2AL}^{lf}$, il est possible de trouver un ordre d'énonciation pour chaque extension de la sémantique complète dans le cas des cycles de longueur paire et de longueur impaire, ce qui résout le problème de la complétude de $\mathcal{T}_{AF2AL}$.

Théorème 4.3 (Complétude de la transformation $\mathcal{T}_{AF2AL}^{lf}$). *Soit AF un AAF et $\mathcal{L}_c$ un étiquetage complet sur AF .*

Alors il existe une séquence ς telle que l'étiquetage $\mathcal{L}_t$ associé à l'état final argumentatif vérifie $\mathcal{L}_c = \mathcal{L}_t$.

Principe de la preuve. Étant donné un étiquetage complet, on construit un ordre d'énonciation qui permet d'atteindre cet étiquetage. Pour cela on utilise le principe d'arguments appelés *légalement UNDEC* : un argument est étiqueté UNDEC si et seulement si aucun de ses attaquants n'est étiqueté IN et qu'au moins un de ses attaquants n'est pas étiqueté OUT. Grâce à une caractérisation des étiquetages complets basée sur cette notion (Caminada, 2017), on montre qu'en énonçant d'abord tous les arguments OUT puis tous les arguments IN et enfin tous les arguments UNDEC, on retrouve bien un état final argumentatif dont l'étiquetage associé est le même que l'étiquetage recherché. $\square$

Une autre propriété intéressante de cette modification est que la répétition d'un argument a un effet encore plus important : non seulement cela remet le statut de l'argument à acceptable, mais cela oblige également à réévaluer l'ensemble du graphe en considérant d'abord cet argument comme acceptable.

Si l'on considère maintenant les autres propriétés formelles, la correction est toujours vérifiée. En effet, la caractérisation d'un état Σ_c -argumentatif avec les évènements exogènes est également valable. La seule différence est que maintenant les arguments précédemment énoncés sont traités en premier, puis une nouvelle boucle de mise à jour se produit sur l'ensemble du graphe. Il en va de même pour la terminaison. L'idée de la preuve ne change pas. La seule différence est qu'au lieu de décomposer le processus de mise à jour en deux étapes, il y en a maintenant quatre.

4.4.3 Discussion sur les autres sémantiques

Au-delà de la sémantique complète étudiée dans la section 4.2.1, nous discutons ici d'autres sémantiques et des questions techniques qu'elles soulèvent.

Tout d'abord, comme il n'existe qu'une seule extension de base, la construction d'une transformation correcte pour cette sémantique aboutirait à la première forme de complétude, c'est-à-dire que quel que soit l'ordre d'énonciation, l'état argumentatif final restera le même : l'extension de base. Ce n'est pas le comportement que nous désirons.

Ensuite, la sémantique préférée peut être considérée comme une "sémantique globale" dans la mesure où il s'agit d'une extension complète maximale. Par conséquent, elle nécessite de déterminer toutes les extensions complètes ou admissibles dans une étape préliminaire. Il s'agit d'une propriété globale qui ne correspond pas à notre approche. En effet, comme indiqué dans la section 4.4.1, notre objectif est de générer une extension en fonction d'un ordre d'énonciation. Ainsi, sans caractérisation « locale », c'est-à-dire indépendamment des autres sémantiques, il est difficile de garantir que l'étiquetage associé à l'énoncé final est un étiquetage préféré plutôt qu'un étiquetage complet.

En ce qui concerne la sémantique stable, il est possible de proposer une définition d'un état Σ_s -argumentatif basée sur la propriété qu'un étiquetage $\mathcal{L}$ est stable si et seulement si il est complet et $undec(\mathcal{L}) = \emptyset$.

Définition 4.3 (État Σ_s -argumentatif). *Un état $S(t)$ est un état Σ_s -argumentatif si :*

- i) $\forall x, [S(t) \models p_x \wedge i_x \wedge \neg o_x \Leftrightarrow S(t) \models p_x \wedge (\bigwedge_y (cA_{y,x} \wedge \neg i_y \wedge o_y) \vee \neg cA_{y,x})]$;
- ii) $\forall x, [S(t) \models p_x \wedge \neg i_x \wedge o_x \Leftrightarrow \exists y, S(t) \models p_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y]$;
- iii) $\forall x, [S(t) \models p_x \wedge ((i_x \wedge \neg o_x) \vee (\neg i_x \wedge o_x))]$.

La seule différence entre un état Σ_c -argumentatif et un état Σ_s -argumentatif est la condition **iii)**. Cette dernière est ajoutée en référence à la caractérisation d'une fonction d'étiquetage stable proposée par Modgil & Caminada (2009) : une fonction d'étiquetage $\mathcal{L}$ est stable si et seulement si elle est complète et que $undec(\mathcal{L}) = \emptyset$. Nous avons montré dans le théorème 4.2 qu'à un état Σ_c -argumentatif, on peut associer une fonction d'étiquetage complète. Par conséquent, nous pensons qu'en rajoutant la condition **iii)** qui impose que tous les arguments d'un état Σ_s -argumentatif soient IN ou OUT, nous pouvons obtenir un théorème similaire. Il ne manque plus qu'à définir des événements exogènes pour, étant donné un état Σ_c -argumentatif, rendre les arguments encore UNDEC soit IN soit OUT. Toutefois, la difficulté avec la sémantique stable et cette définition est qu'il n'y a aucune garantie de l'existence d'au moins une extension stable pour un graphe d'argumentation quelconque. Si nous suivons la même méthodologie que celle utilisée dans la section 4.2, alors la terminaison n'est plus garantie. En effet, supposons qu'il soit possible de prouver qu'un état Σ_s -argumentatif est équivalent à un état où aucun événement exogène n'est déclenché et qu'il est correct. Imaginons maintenant le scénario suivant : étant donné un état Σ_s -argumentatif où une extension stable existe, un nouvel argument est énoncé. Cependant, il n'y a pas d'extension stable dans le nouveau graphe d'argumentation associé. Alors, à moins d'ajouter une condition d'arrêt, les événements exogènes continueront à être déclenchés pour chaque état sans s'arrêter puisqu'il est impossible d'atteindre cet état Σ_s -argumentatif. Une étude approfondie de cette sémantique est laissée pour un travail futur.

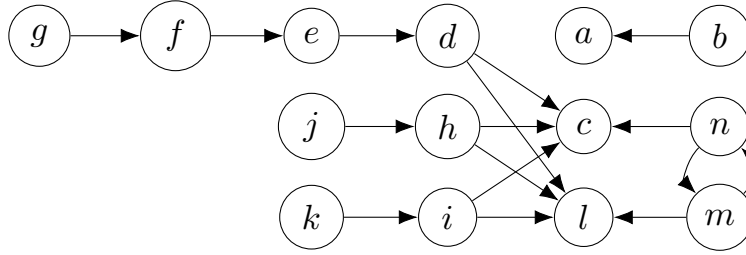


FIGURE 4.2 – Graphe d'argumentation associé à l'exemple 1.

4.5 Application à l'exemple médical

Dans cette section, nous revenons sur l'exemple 1 pour illustrer les différentes contributions de ce chapitre. Nous considérons en particulier la version non simplifiée de l'exemple 1 dont on rappelle le graphe d'argumentation sur la figure 4.5.

D'un point de vue argumentatif et en considérant la sémantique complète, il y a trois scénarios possibles correspondant aux trois extensions complètes : faire une IRM dans trois jours ($\{e, g, j, k, c, m\}$), en faire une d'urgence aujourd'hui ($\{e, g, j, k, l, n\}$) ou bien ne rien faire du tout ($\{e, g, j, k\}$). Dans le troisième cas, l'urgence (n) et l'absence de disponibilité (m) se neutralisent et les deux arguments sont UNDEC tout comme les variables de décisions l et c .

Avec la première transformation $\mathcal{T}_{AF2AL}$ que nous avons proposée et comme nous l'avons illustré avec l'exemple 2, nous obtenons uniquement le troisième scénario, à savoir le statu quo ce qui n'est évidemment pas satisfaisant. Cependant, en appliquant la stratégie "dernier énoncé, dernier mis à jour" c'est-à-dire la transformation $\mathcal{T}_{AF2AL}^{lf}$, la décision finale est de faire l'IRM aujourd'hui (l est acceptable). Si cela n'est vraiment pas possible malgré l'urgence, alors le radiologue répète qu'il n'a vraiment pas de disponibilité (m). Cela rend cet argument à nouveau acceptable et donc, avec la stratégie que nous proposons, le fait que ce soit une urgence (n) devient inacceptable. Finalement, la décision finale devient de faire une IRM dans trois jours (en tout cas en restant avec ce spécialiste). Chacun des interlocuteurs peut alors à tour de rôle répéter que c'est une urgence ou qu'il n'y a pas de créneaux disponibles s'engageant ainsi dans un dialogue très constructif ou bien jusqu'à ce qu'une autre possibilité soit envisagée (aller voir un autre service de radiologie) ou qu'un autre argument soit énoncé, par exemple le fait qu'il faut éviter les radiations ionisantes pour un bébé mais en cas d'urgence un scanner est quand même envisageable.

Concernant la génération d'explications causales, la méthode est identique à celle du chapitre 3. En effet, le processus d'extraction des traces causales prend en entrée uniquement les traces d'événements et d'états. On peut donc créer le même type d'illustration graphique du dialogue pour supporter l'explication mais également, une fois le processus d'extraction des bonnes explications à partir de la chaîne causale formalisée, générer des bonnes explications. Sur cet exemple, on obtient la même explication que dans la section 3.5.3 à savoir que c'est parce que c'est une urgence, c'est-à-dire que l'argument n a été énoncé. Cela vient du fait que dans le cas du cycle de longueur 2 entre n et m non attaqué par d'autres arguments, notre stratégie revient à casser le cycle en fonction de l'ordre d'énonciation.

4.6 Conclusion

Dans ce chapitre nous avons proposé une généralisation des travaux présentés dans le chapitre 3. En effet, nous avons introduit une formalisation des systèmes d'argumentation abstraite, cycliques ou non, qui prend en compte l'ordre d'énonciation des arguments. Nous avons également établi et prouvé des propriétés formelles pour la sémantique complète. L'avantage de cette transformation est double : elle nous permet d'une part de discriminer entre les extensions d'une sémantique en fonction de l'ordre des arguments, et d'autre part elle augmente la quantité d'informations en sauvegardant l'historique de tout le dialogue sous la forme de deux traces. Ces travaux ainsi que ceux présentés dans le chapitre précédent sont une première étape dans l'exploration de l'utilisation des langages d'action pour la génération d'explications causales pour un modèle d'argumentation abstraite. En effet, la notion de causalité introduite dans le chapitre 3 associée au langage d'action que nous utilisons permet de donner des informations riches sur le statut d'acceptabilité des arguments et des justifications sur ce dernier.

La prochaine étape consiste à formaliser les propriétés proposées par Miller (2019) pour une bonne explication dans le langage d'action. Cela permettrait de proposer une méthode de génération d'explications ordonnées et adaptées au destinataire. Nous souhaitons également étudier la pertinence vis-à-vis d'un utilisateur humain de notre stratégie "dernier énoncé, dernier mis à jour". Enfin, nous souhaitons étendre les travaux que nous avons présentés dans les précédents chapitres à un cadre d'argumentation abstraite plus général. Ainsi, dans le chapitre suivant nous discutons de la génération d'explications causales pour des systèmes d'argumentation bipolaires pondérés.

Systèmes d'argumentation bipolaire pondérée : sémantique agrégative et explications causales

Dans ce chapitre, nous nous intéressons à une double généralisation des systèmes d'argumentation abstraite appelée système d'argumentation abstraite bipolaire pondérée. Il s'agit d'un AAF classique auquel est ajoutée d'une part une relation de support qui, associée à la relation d'attaque, crée la bipolarité, d'autre part une force intrinsèque aux arguments. Dans ce formalisme, une méthode classique pour étudier l'acceptabilité des arguments consiste à utiliser une fonction d'acceptabilité appelée sémantique graduelle. Nous proposons dans ce chapitre un cas particulier de ces fonctions, que nous appelons sémantique agrégative graduelle, dans lequel les poids des supporters et des attaquants sont agrégés séparément, puis combinés avec la force intrinsèque de l'argument. Nous discutons également des propriétés de cette nouvelle famille de sémantiques dans le cadre de la génération d'explications causales adaptées au destinataire. Enfin nous discutons d'une réécriture de ces AAF étendus dans le formalisme des modèles structurels causaux.

Ces travaux ont été présentés dans la conférence française avec comité de relecture JIAF (Munro et al., 2024a).

Sommaire

5.1 Introduction	88
5.2 Rappels préliminaires	89
5.2.1 Définition d'un QBAF et exemple illustratif	89
5.2.2 Acceptabilité et sémantique graduelle	90
5.2.3 Axiomes classiques des sémantiques graduelles	92
5.3 Sémantique agrégative graduelle	96
5.3.1 Fonction d'agrégation et propriétés usuelles	96
5.3.2 Motivation et définition proposée	96
5.3.3 Liens et différences avec les sémantiques modulaires	98
5.4 Propriétés souhaitables des différentes fonctions d'agrégation	99

5.4.1	Les cas de φ_R et φ_S	100
5.4.2	Le cas de φ_f	103
5.5	Comparaison formelle avec les axiomes classiques des QBAF	105
5.5.1	Lien entre axiomes argumentatifs et propriétés des opérateurs d'agrégation	105
5.5.2	Lien entre axiomes argumentatifs et interactions des opérateurs d'agrégation	107
5.5.3	Discussion sur les sémantiques graduelles classiques	108
5.6	QBAF et modèles structurels causaux	110
5.6.1	Passage des QBAF aux modèles structurels causaux (MSC)	111
5.6.2	Que cherche-t-on réellement à expliquer?	113
5.7	Conclusion	115

5.1 Introduction

Les chapitres 2, 3 et 4 se sont concentrés sur la génération d'explications causales pour un système d'argumentation abstraite, c'est-à-dire constitué d'arguments tous identiques dont la seule différence réside dans l'information qu'ils contiennent et avec une unique relation binaire, la relation d'attaque qui indique une contradiction entre l'attaquant et l'argument attaqué. Pour soutenir un autre argument, la seule façon de faire est de le défendre en attaquant un de ses attaquants directs ou indirects.

Considérons par exemple la question fictive de la présence du jeu d'échecs comme discipline des Jeux Olympiques (JO) de Los Angeles en 2028 (*a*). Un argument qui s'y oppose, sans en juger la pertinence, pourrait être qu'il y avait déjà trop de disciplines différentes aux JO de Paris (*b*). Un argument en faveur de cette mesure serait de dire que les échecs respectent tous les critères imposés par le Comité International Olympique (CIO) (*c*). Il s'agit clairement d'un argument soutenant la présence des échecs aux JO de Los Angeles. Pourtant, cet argument ne défend pas *a* car il n'attaque aucun des attaquants de *a*. Une solution pourrait être de créer un argument supplémentaire *a'* selon lequel : Les échecs ne doivent pas être aux JO de Los Angeles, et de dire que *a* et *a'* s'attaquent mutuellement et que les arguments en faveur de *a* attaquent *a'*. Le problème de cette approche est que dans l'AAF résultat que nous venons de décrire *a* et *a'* sont tous les deux refusés alors que chacune des alternatives semble encore envisageable.

Une autre solution est d'ajouter une seconde relation binaire, appelée support, et qui permet de modéliser ce soutien direct. Un système d'argumentation comportant les deux relations, d'attaque et de support, est dit bipolaire (Cayrol & Lagasquie-Schiex, 2005b) ou *Bipolar Argumentation Framework* (BAF) en anglais. Cependant, dans l'exemple précédent, nous pourrions dire que le fait que les échecs vérifient tous les critères du CIO a plus de poids que dire qu'il y a déjà trop de disciplines à Paris. En effet, *c* est un argument rationnel dont la vérification auprès de l'organisme qui gère lui-même les JO est aisé alors que l'autre est un avis purement subjectif pas nécessairement partagé. Cette différence subjective n'est pas représentable avec un BAF. Il existe néanmoins une autre extension des AAF appelée système d'argumentation abstraite pondérée (Bistarelli & Santini, 2021) permettant d'associer à chaque argument une valeur, le plus souvent numérique entre 0 et 1, pour représenter sa "force intrinsèque". Nous discutons plus en détail de sa signification dans la suite.

Dans ce chapitre nous étudions la combinaison de ces deux extensions, appelée système d’argumentation bipolaire pondérée (Baroni *et al.*, 2019). Nous rappelons d’abord quelques notions sur ces systèmes ainsi que sur les fonctions d’agrégations, nécessaires pour la suite. Ensuite nous proposons une famille de sémantiques graduelles, appelées des sémantiques graduelles agrégatives, permettant de calculer l’acceptabilité des arguments. Puis nous discutons des propriétés de ces fonctions avant de les comparer avec celles déjà présentes dans la littérature. Nous revenons finalement sur la question des explications du statut d’acceptabilité des arguments en utilisant, comme dans le chapitre 2, les graphes causaux argumentatifs (GCA). Nous proposons en particulier une généralisation de la transformation $\mathcal{T}_{AFa2GCA}$ permettant de passer d’un AAF à un GCA, avant de conclure ce chapitre.

5.2 Rappels préliminaires

Dans cette section, nous rappelons les concepts de base sur les systèmes d’argumentation bipolaire pondérée (*Quantitative Bipolar Argumentation Framework*, QBAF) et sur les sémantiques associées. En particulier, nous détaillons la notion de sémantique graduelle (Cayrol & Lagasquie-Schiex, 2005a) ainsi que les axiomes qui ont été proposés pour les caractériser (Amgoud & Ben-Naim, 2018; Mossakowski & Neuhaus, 2018; Potyka, 2019).

5.2.1 Définition d’un QBAF et exemple illustratif

Définition 5.1 (Système d’argumentation bipolaire pondérée (Baroni *et al.*, 2019)). *Un système d’argumentation bipolaire pondérée ou Quantitative Bipolar Argumentation Framework (QBAF) est un quadruplet $A = (\mathcal{A}, \mathcal{R}, S, w)$ tel que :*

- *$(\mathcal{A}, \mathcal{R})$ est un AAF tel que défini dans la définition 1.1 du chapitre 1 ;*
- *S est une relation binaire sur $\mathcal{A}$ ($S \subseteq \mathcal{A} \times \mathcal{A}$) appelée support ;*
- *$w : \mathcal{A} \rightarrow I$ est une fonction qui associe une force intrinsèque à chaque argument dans un ensemble de valeurs préordonné I , par exemple $I = [0, 1]$.*

On note WAG l’ensemble des QBAF.

Dans tout le chapitre, nous ajoutons à cette définition l’hypothèse suivante : $\mathcal{R} \cap S = \emptyset$. Elle implique qu’il n’y a pas d’argument qui attaque et supporte un même argument. Nous pensons qu’il s’agit d’une hypothèse raisonnable car de tels arguments semblent incohérents, et nous supposons que ces incohérences ont été gérées en amont lors de la génération du QBAF.

Comme $\mathcal{R}$ et S sont des relations binaires définies sur un ensemble fini, un QBAF peut être représenté sous la forme d’un graphe orienté étiqueté dans lequel les nœuds représentent les arguments, les étiquettes leur force intrinsèque, et les arcs sont de deux types, représentant les relations d’attaque et de support.

Remarque – Dans la plupart des articles de la littérature sur les QBAFs, la valeur associée à un argument est appelée force intrinsèque (Amgoud & Ben-Naim, 2018; Baroni *et al.*, 2019; Kampik *et al.*, 2024). Toutefois, la sémantique associée à cette force intrinsèque d’un argument peut être variée. Par exemple, dans un cadre de modélisation de débats en ligne, Rago & Toni (2017) associent à cette valeur le nombre de réactions

positives à un argument. Certains travaux en argumentation abstraite réécrivent un réseau de neurones sous la forme d'un QBAF (Potyka, 2021). Dans ce cas, la force intrinsèque correspond au biais de chaque neurone. da Costa Pereira *et al.* (2011) notamment considèrent que la force intrinsèque d'un argument représente une notion de confiance a priori en une source. Dans la suite du chapitre, nous ne considérons pas de sémantique particulière pour rester dans un cadre de discussion le plus général possible.

Pour illustrer ce chapitre, nous introduisons l'exemple fictif esquissé dans l'introduction et détaillé ci-dessous portant sur l'intégration du jeu d'échecs aux Jeux Olympiques : il permet de discuter les notions de support et de poids des arguments de façon plus réaliste que l'exemple médical utilisé dans les chapitres précédents. Comme dans la plupart des articles de la littérature sur les QBAF (Delobelle & Villata, 2019; Čyras *et al.*, 2022; Yin *et al.*, 2023, 2024b), il conduit à un graphe acyclique avec une question centrale débattue, appelée le *sujet*. Celui-ci correspond à la notion de variable de décision.

Exemple 9 (Le jeu d'échecs aux JO). *On considère l'ensemble d'arguments A suivant autour de la présence du jeu d'échecs comme discipline aux JO 2028 :*

- a : les échecs doivent être une discipline aux JO de Los Angeles ;
- b : les parties peuvent être beaucoup trop longues ;
- c : les échecs ne sont pas un sport ;
- d : les échecs respectent les critères fixés par le CIO ;
- e : il y a déjà trop de disciplines différentes aux JO de Paris ;
- f : certaines épreuves commencent même avant la cérémonie d'ouverture ;
- g : aux jeux de Los Angeles il y aura déjà plus de disciplines qu'à Paris ;
- h : il existe des tournois officiels d'échecs en format rapide ;
- i : le jeu d'échecs est une activité compétitive avec des règles strictes ;
- j : le jeu d'échecs est une activité physique intense ;
- k : le jeu d'échecs est une activité nécessitant de l'habileté physique ;
- l : la Fédération Internationale Des Échecs (FIDE) a été reconnue par le CIO ;
- m : la FIDE impose des mesures anti-dopage strictes ;
- n : les échecs ont une bonne image dans le monde.

Nous choisissons de définir l'ensemble des relations d'attaque et de support comme :

- $\mathcal{R} = \{(b, a), (c, a), (e, a), (g, e), (h, b), (i, c), (j, c), (k, c), (l, c)\}$;
- $\mathcal{S} = \{(d, a), (f, e), (l, d), (m, d), (n, d)\}$.

Enfin, il reste à définir la force intrinsèque de chaque argument. Une valeur arbitraire entre 0 et 1 est associée à chacun d'entre eux pour illustrer notre propos dans la suite à l'exception de a . En effet, on prend $w(a) = 0.5$, c'est-à-dire au centre de l'intervalle, pour indiquer une absence de biais en faveur ou contre la décision. Cela dépend totalement de la sémantique associée à la notion de force intrinsèque : sans plus de précision on adopte cette convention pour le sujet du graphe d'argumentation uniquement.

Le graphe d'argumentation associé est représenté sur la figure 5.1.

5.2.2 Acceptabilité et sémantique graduelle

Pour déterminer le statut d'acceptabilité de chaque argument dans le contexte des QBAF, trois approches sont principalement utilisées, respectivement catégorielle, ordinaire et numérique :

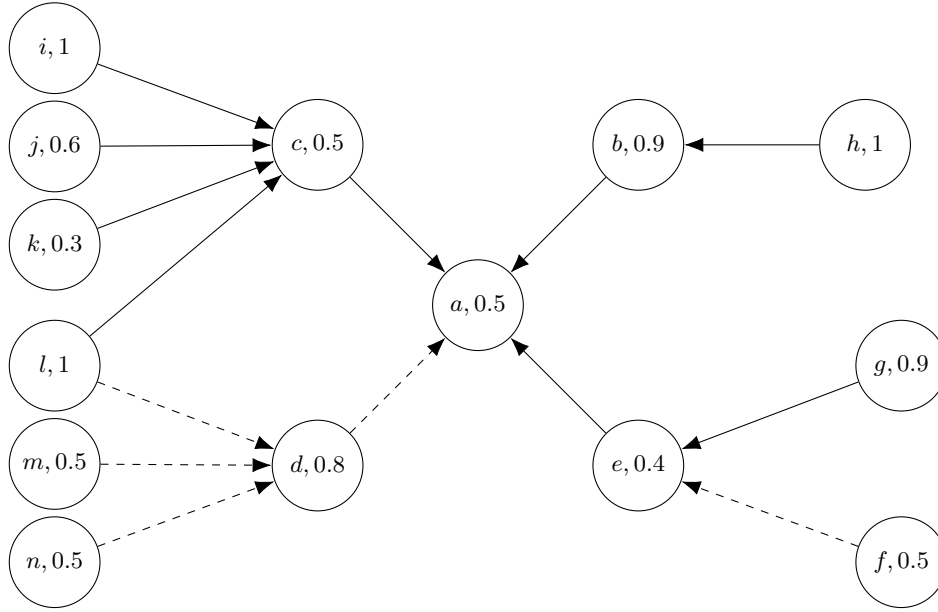


FIGURE 5.1 – Graphe d’argumentation associé à l’exemple 9. Les traits pleins représentent les attaques et les pointillés les supports. La valeur à côté du nom de l’argument correspond à sa force intrinsèque, appartenant à l’intervalle $[0, 1]$.

- les approches catégorielles par extension (Cayrol & Lagasque-Schiex (2005b) par exemple) déterminent des ensembles d’arguments collectivement acceptables selon certains critères ;
- les approches ordinales par classement (Bonzon *et al.* (2016) par exemple) renvoient un classement des arguments, des plus acceptables aux moins acceptables ;
- les approches numériques par évaluation (Cayrol & Lagasque-Schiex (2005a) par exemple) associent une valeur d’acceptabilité à chaque argument.

Dans ce chapitre, nous nous intéressons à ce troisième type d’approches appelées sémantiques graduelles.

Définition 5.2 (Sémantique graduelle). *Une sémantique graduelle S est une fonction définie sur WAG qui associe à $A \in WAG$ une fonction $Deg_A^S : \mathcal{A} \rightarrow I$, où $Deg_A^S(a)$ est le degré d’acceptabilité de l’argument a .*

Dans la suite, on utilise Deg_A^S pour parler à la fois du degré d’acceptabilité et de la sémantique S . De plus, pour alléger les notations dans les cas où il n’y a pas d’ambiguïté, nous notons $Deg_A(a)$ plutôt que $Deg_A^S(a)$.

En plus de cette définition, nous utilisons les notations usuelles ci-dessous dans la suite de ce chapitre :

- $Att_a = \{b \in \mathcal{A} \mid (b, a) \in \mathcal{R}\}$ est l’ensemble des arguments qui attaquent l’argument a , appelés les *attaquants* ;
- $sAtt_a = \{b \in Att_a \mid Deg_A(b) \neq 0\}$ est l’ensemble des attaquants de a de degré d’acceptabilité non nul ;
- $Supp_a = \{c \in \mathcal{A} \mid (c, a) \in \mathcal{S}\}$ est l’ensemble des arguments qui supportent l’argument a , appelés les *supporters* ;

- $\text{sSupp}_a = \{c \in \text{Supp}_a \mid \text{Deg}_A(c) \neq 0\}$ est l'ensemble des supporteurs de a de degré d'acceptabilité non nul ;
- Soit A, A' des QBAF tels que $\mathcal{A} \cap \mathcal{A}' = \emptyset$. $A'' = A \oplus A'$ est le QBAF défini par : $\mathcal{A}'' = \mathcal{A} \cup \mathcal{A}'$, $\mathcal{R}'' = \mathcal{R} \cup \mathcal{R}'$, $\mathcal{S}'' = \mathcal{S} \cup \mathcal{S}'$ et $w''(a) = w(a)$ si $a \in \mathcal{A}$ et $w''(a) = w'(a)$ si $a \in \mathcal{A}'$;
- On appelle isomorphisme de A vers A' une fonction f bijective qui conserve les relations et les étiquettes des graphes : $\forall a, b \in \mathcal{A}, w(a) = w'(f(a))$, $(b, a) \in \mathcal{R}$ ssi $(f(b), f(a)) \in \mathcal{R}'$ et $(b, a) \in \mathcal{S}$ ssi $(f(b), f(a)) \in \mathcal{S}'$, où w et w' sont les forces des arguments de A et A' respectivement.

Sémantiques graduelles couramment utilisées dans la littérature. Nous présentons ici trois sémantiques graduelles parmi les plus utilisées dans la littérature :

- DF-Quad (Rago *et al.*, 2016) :

$$\text{Deg}_A(a) = w(a) - w(a) \max(0, -s) + (1 - w(a)) \max(0, s)$$

$$\text{avec } s = \prod_{b \in \text{Att}_a} (1 - \text{Deg}_A(b)) - \prod_{c \in \text{Supp}_a} (1 - \text{Deg}_A(c))$$

- Euler-based (Ebs) (Amgoud & Ben-Naim, 2018) :

$$\text{Deg}_A(a) = 1 - \frac{1 - w(a)^2}{1 + w(a)e^s}$$

$$\text{avec } s = \sum_{c \in \text{Supp}_a} (\text{Deg}_A(c)) - \sum_{b \in \text{Att}_a} (\text{Deg}_A(b))$$

- Quadratic Energy (QE) (Potyka, 2018) pour des QBAFs acycliques :

$$\text{Deg}_A(a) = w(a) - w(a) \times \frac{\max(0, -s)^2}{1 + \max(0, -s)^2} + (1 - w(a)) \times \frac{\max(0, s)^2}{1 + \max(0, s)^2}$$

$$\text{avec } s = \sum_{c \in \text{Supp}_a} (\text{Deg}_A(c)) - \sum_{b \in \text{Att}_a} (\text{Deg}_A(b))$$

Les définitions de ces sémantiques sont récursives. En effet, le calcul de l'acceptabilité d'un argument a nécessite le degré d'acceptabilité des attaquants et des supporteurs de a . Par conséquent, dans le cas des graphes cycliques, la convergence n'est pas garantie.

Exemple 9 (Le jeu d'échecs aux JO). *(suite) – L'application de DF-Quad et Ebs sur l'exemple conduit respectivement à un degré d'acceptabilité $\text{Deg}_A^{\text{DF-Quad}}(a) = 0.88$ contre $\text{Deg}_A^{\text{Ebs}}(a) = 0.42$. Ainsi, avec un seuil de décision à 0.5 par exemple, les deux issues sont contradictoires : les échecs doivent devenir une discipline olympique selon DF-Quad et ils ne doivent pas l'être avec Ebs. Il est donc important d'établir des propriétés permettant de caractériser le comportement des sémantiques pour pouvoir les choisir.*

5.2.3 Axiomes classiques des sémantiques graduelles

La définition 5.2 précédente est très générale et n'impose aucune contrainte sur la forme que peut prendre la fonction Deg_A . Une approche axiomatique a été proposée pour

permettre de choisir une sémantique adaptée selon le comportement recherché (Amgoud & Ben-Naim, 2018; Mossakowski & Neuhaus, 2018; Potyka, 2019). En particulier, les douze axiomes énoncés par Amgoud & Ben-Naim (2018) utilisent une définition très générale des sémantiques graduelles et sont donc les plus utilisés. Nous les décrivons brièvement ci-dessous en reproduisant les définitions formelles et en en discutant une interprétation. Un récapitulatif est fourni en annexe D dans la table 1.

Soit $A, A' \in WAG$:

(A1) – Anonymat : Le degré d’acceptabilité d’un argument est invariant par isomorphisme.

Formellement, pour tout isomorphisme f de A vers A' , on a :

$$\forall a \in \mathcal{A}, \text{Deg}_A(a) = \text{Deg}_{A'}(f(a))$$

(A2) – Indépendance : Le degré d’acceptabilité d’un argument ne dépend que des nœuds auxquels il est connecté.

Formellement, si $\mathcal{A} \cap \mathcal{A}' = \emptyset$ alors :

$$\forall a \in \mathcal{A}, \text{Deg}_A(a) = \text{Deg}_{A \oplus A'}(a)$$

(A3) – Directionnalité : Le degré d’acceptabilité d’un argument ne dépend que de ses ancêtres, c’est-à-dire ses attaquants (directs ou indirects) et ses supporters (directs ou indirects).

Formellement, si $\mathcal{A} = \mathcal{A}', w = w', \mathcal{R} \subseteq \mathcal{R}'$ et $\mathcal{S} \subseteq \mathcal{S}'$ alors :

$$\forall a, b, x \in \mathcal{A}, \text{ si } \mathcal{R}' \cup \mathcal{S}' = \mathcal{R} \cup \mathcal{S} \cup \{(a, b)\} \text{ et qu'il n'y a pas de chemin entre } b \text{ et } x, \\ \text{ alors } \text{Deg}_A(x) = \text{Deg}_{A'}(x)$$

(A4) – Équivalence : Le degré d’acceptabilité d’un argument ne dépend que de sa force intrinsèque et du degré d’acceptabilité de ses attaquants et de ses supporters.

Formellement, soit $a, b \in \mathcal{A}$ tels que :

- $w(a) = w(b)$
- $\exists f : \text{Att}_a \rightarrow \text{Att}_b$ bijective telle que $\forall x \in \text{Att}_a, \text{Deg}_A(x) = \text{Deg}_A(f(x))$
- $\exists f' : \text{Supp}_a \rightarrow \text{Supp}_b$ bijective telle que : $\forall y \in \text{Supp}_a, \text{Deg}_A(y) = \text{Deg}_A(f'(y))$

alors, $\text{Deg}_A(a) = \text{Deg}_A(b)$

(A5) – Stabilité : Le degré d’acceptabilité d’un argument non attaqué et non supporté est égal à sa force intrinsèque.

Formellement, $\forall a \in \mathcal{A}$:

$$\text{Si } \text{Att}_a = \text{Supp}_a = \emptyset, \text{ alors } \text{Deg}_A(a) = w(a)$$

(A6) – Neutralité : Les arguments de degré d’acceptabilité nul n’ont pas d’impact sur le degré d’acceptabilité de leurs enfants.

Formellement, soit $a, b, x \in \mathcal{A}$ tels que :

- $w(a) = w(b)$
- $\text{Deg}_A(x) = 0$
- $\text{Att}_a \subseteq \text{Att}_b, \text{Supp}_a \subseteq \text{Supp}_b$
- $\text{Att}_b \cup \text{Supp}_b = \text{Att}_a \cup \text{Supp}_a \cup \{x\}$

alors $\text{Deg}_A(a) = \text{Deg}_A(b)$

(A7) – Monotonie : Un argument b de force intrinsèque égale à celle d'un autre a mais qui est au moins autant attaqué et au plus autant supporté a un degré d'acceptabilité au plus égal à celui de a .

Formellement soit $a, b \in \mathcal{A}$ tels que $w(a) = w(b)$, si $\text{Att}_a \subseteq \text{Att}_b$ et $\text{Supp}_b \subseteq \text{Supp}_a$:

- alors $\text{Deg}_A(a) \geq \text{Deg}_A(b)$ (Monotonie)
- si $\text{Deg}_A(a) > 0$ et $\text{sAtt}_a \subset \text{sAtt}_b$ ou $\text{Deg}_A(b) < 1$ et $\text{sSupp}_b \subset \text{sSupp}_a$, alors $\text{Deg}_A(a) > \text{Deg}_A(b)$ (Monotonie stricte)

(A8) – Renforcement : Un argument dont la "qualité" des attaquants augmente et dont la "qualité" des supporteurs diminue a un degré d'acceptabilité plus faible.

Formellement soit $a, b \in \mathcal{A}$, $C, C' \subseteq \mathcal{A}$ et $x, x', y, y' \in \mathcal{A} \setminus (C \cup C')$ tels que :

- $w(a) = w(b)$
- $\text{Att}_a = C \cup \{x\}$ et $\text{Att}_b = C \cup \{y\}$
- $\text{Supp}_a = C' \cup \{x'\}$ et $\text{Supp}_b = C' \cup \{y'\}$
- $\text{Deg}_A(x) \leq \text{Deg}_A(y)$ et $\text{Deg}_A(x') \geq \text{Deg}_A(y')$

alors :

- $\text{Deg}_A(a) \geq \text{Deg}_A(b)$ (Renforcement)
- Si $\text{Deg}_A(a) > 0$ et $\text{Deg}_A(x) < \text{Deg}_A(y)$ ou $\text{Deg}_A(b) < 1$ et $\text{Deg}_A(x') > \text{Deg}_A(y')$, alors $\text{Deg}_A(a) > \text{Deg}_A(b)$ (Renforcement strict)

(A9) – Résilience : Un argument dont la force intrinsèque n'est ni maximale ni minimale ne peut avoir un degré d'acceptabilité ni maximal ni minimal. Formellement $\forall a \in \mathcal{A}$:

$$\text{si } 0 < w(a) < 1 \text{ alors } 0 < \text{Deg}_A(a) < 1$$

(A10) – Principe de Franklin : Ajouter un attaquant et un supporteur de même degré d'acceptabilité à un argument diminue ou maintient constant le degré d'acceptabilité de cet argument.

Formellement, soit $a, b, x, y \in \mathcal{A}$ tels que :

- $w(a) = w(b)$
- $\text{Att}_a = \text{Att}_b \cup \{x\}$ et $\text{Supp}_a = \text{Supp}_b \cup \{y\}$
- $\text{Deg}_A(x) = \text{Deg}_A(y)$

alors :

- $\text{Deg}_A(a) \leq \text{Deg}_A(b)$ (Principe de Franklin)
- $\text{Deg}_A(a) = \text{Deg}_A(b)$ (Principe de Franklin strict)

(A11) – Affaiblissement : Si les attaquants d'un argument sont de "meilleure qualité" que ses supporteurs, alors le degré d'acceptabilité de cet argument est inférieur à sa force intrinsèque.

Formellement, soit $a \in \mathcal{A}$. Si $w(a) > 0$ et il existe une fonction injective f de Supp_a vers Att_a telle que :

- $\forall x \in \text{Supp}_a, \text{Deg}_A(x) \leq \text{Deg}_A(f(x))$
- $\text{sAtt}_a \setminus \{f(x) | x \in \text{Supp}_a\} \neq \emptyset$ ou $\exists x \in \text{Supp}_a$ tel que $\text{Deg}_A(x) < \text{Deg}_A(f(x))$

alors $\text{Deg}_A(a) < w(a)$.

(A12) – Consolidation : Si les supporteurs d'un argument sont de "meilleure qualité" que ses attaquants, alors le degré d'acceptabilité de cet argument est supérieur à sa force intrinsèque.

Formellement : soit $a \in \mathcal{A}$. Si $w(a) > 0$ et il existe une fonction injective f de Att_a vers Supp_a telle que :

- $\forall x \in \text{Att}_a, \text{Deg}_A(x) \leq \text{Deg}_A(f(x))$
 - $\text{sSupp}_a \setminus \{f(x) | x \in \text{Att}_a\} \neq \emptyset$ ou $\exists x \in \text{Att}_a$ tel que $\text{Deg}_A(x) < \text{Deg}_A(f(x))$
- alors $\text{Deg}_A(a) > w(a)$.

Un autre axiome, proposé par Mossakowski & Neuhaus (2018), porte le nom de modularité. Il définit une sémantique graduelle particulière, appelée *sémantique modulaire*, de la façon suivante :

Définition 5.3 (Sémantique modulaire (Mossakowski & Neuhaus, 2018)). *Soit Deg_A une sémantique graduelle et $a \in \mathcal{A}$ un argument.*

Deg_A est modulaire si $\text{Deg}_A(a)$ s'écrit sous la forme :

$$\text{Deg}_A(a) = i(\alpha(\mathbf{g}_a, \mathbf{s}), w(a))$$

avec :

- $\mathbf{g}_a$ le vecteur des relations de a (1 pour un support, -1 pour un attaquant et 0 sinon) ;
- $\mathbf{s}$ le vecteur des degrés d'acceptabilité de tous les arguments du graphe ;
- α une fonction d'agrégation ;
- i une fonction appelée fonction d'influence.

Cette définition permet de décomposer le calcul de l'acceptabilité d'un argument en deux étapes, séparant la prise en compte de la force intrinsèque par la fonction i , de la prise en compte des degrés d'acceptabilité des attaquants et des supporters, par la fonction α . Elle offre la possibilité de contraindre chacune des étapes différemment. Les trois sémantiques graduelles présentées dans la section précédente peuvent s'écrire sous cette forme (voir Table 5.1).

Il faut noter que, au-delà du cas particulier de ces trois sémantiques, la définition d'une sémantique modulaire est une définition récursive. En effet, le calcul de l'acceptabilité d'un argument nécessite d'avoir calculé l'acceptabilité des attaquants et supporters de ce dernier. De fait, dans le cas d'un graphe d'argumentation cyclique, la convergence de ce calcul n'est pas garantie et dépend du choix de α et de i (Potyka, 2019).

	Fonction d'agrégation α	Fonction d'influence i
DF-Quad	$\alpha : [0, 1]^n \times [0, 1]^n \rightarrow [-1, 1]$ $\alpha : (\mathbf{g}_a, \mathbf{s}) \mapsto \prod_i^n g_{a,i} \times (s_i - 1)$	$i : [-1, 1] \times [0, 1] \rightarrow [0, 1]$ $i : (x, w) \mapsto w \times (1 - \max(0, -x)) + (1 - w) \max(0, x)$
Ebs	$\alpha : [0, 1]^n \times [0, 1]^n \rightarrow \mathbb{R}$ $\alpha : (\mathbf{g}_a, \mathbf{s}) \mapsto \sum_{i=1}^n g_{a,i} \times s_i$	$i : \mathbb{R} \times [0, 1] \rightarrow [0, 1]$ $i : (x, w) \mapsto 1 - \frac{1-w^2}{1+we^x}$
QE	$\alpha : [0, 1]^n \times [0, 1]^n \rightarrow \mathbb{R}$ $\alpha : (\mathbf{g}_a, \mathbf{s}) \mapsto \sum_{i=1}^n g_{a,i} \times s_i$	$i : \mathbb{R} \times [0, 1] \rightarrow [0, 1]$ $i : (x, w) \mapsto w \times \left(1 - \frac{\max(0, -x)^2}{1 + \max(0, -x)^2}\right) + (1 - w) \times \frac{\max(0, x)^2}{1 + \max(0, x)^2}$

TABLE 5.1 – Réécriture des sémantiques graduelles classiques en sémantique modulaire (Mossakowski & Neuhaus, 2018)

5.3 Sémantique agrégative graduelle

Dans cette section, nous rappelons tout d’abord la définition et quelques propriétés sur un type particulier de fonctions appelé des fonctions d’agrégation. Ensuite, nous proposons une nouvelle famille de sémantiques graduelles que nous appelons *sémantiques agrégatives graduelles* qui reprend les idées des sémantiques modulaires tout en allant plus loin sur la décomposition du processus de calcul du degré d’acceptabilité d’un argument. Nous discutons ensuite de ce lien ainsi que des différences entre ces deux types de sémantiques graduelles.

5.3.1 Fonction d’agrégation et propriétés usuelles

Une fonction d’agrégation est une fonction qui combine un ensemble de valeurs en une unique valeur et qui peut prendre de multiples formes, voir par exemple les travaux de Bobillo & Straccia (2013); Dubois & Prade (1985); Grabisch *et al.* (2009); Torra & Narukawa (2007); Yager & Rybalov (1998). Certaines fonctions, comme la moyenne, possèdent un nombre fixe d’arguments; d’autres, comme le maximum, en autorisent un nombre variable.

Afin de choisir une fonction adaptée à un contexte donné, une analyse axiomatique similaire à celle pour les sémantiques graduelles a été faite. Ainsi, de nombreuses propriétés permettant de caractériser leur comportement ont été proposées dans la littérature (voir Bloch (1996); Dubois & Prade (1985)). Certaines d’entre elles sont présentées dans la table 5.2. Nous revenons sur chacune d’entre elles en détails dans la section 5.4.

Remarque – Le nom de certaines propriétés des fonctions d’agrégation est également utilisé pour parler de propriétés parfois très différentes en argumentation (Amgoud & Ben-Naim, 2018). Afin de pouvoir les distinguer, nous utilisons le mot postulat pour parler des propriétés définies pour les fonctions d’agrégation et nous gardons le terme axiome pour les propriétés issues de l’argumentation. On distinguera par exemple le postulat de monotonie qui donne le sens de variation d’une fonction d’agrégation de l’axiome de monotonie décrit dans la section 5.2.3.

5.3.2 Motivation et définition proposée

Décomposer le calcul du degré d’acceptabilité d’un argument en plusieurs étapes “élémentaires”, comme proposé par les sémantiques modulaires, offre plusieurs avantages. Tout d’abord, cela permet de spécifier le comportement attendu pour chaque étape du calcul, c’est-à-dire définir des axiomes pour la fonction d’influence et la fonction d’agrégation. Ensuite, cela permet de fournir différents niveaux d’explications : un premier en se concentrant sur la fonction d’influence, et un autre plus détaillé en remontant jusqu’au degré d’acceptabilité des attaquants et des supporteurs avec la fonction α . Cette méthode de calcul est aussi la plus intuitive comme l’illustre le fait que la grande majorité des sémantiques graduelles existantes dans la littérature peuvent se mettre sous la forme de sémantique modulaire avant que cette notion ne soit définie (Mossakowski & Neuhaus, 2018). Toutefois, nous pensons que décomposer le calcul avec une étape supplémentaire permet d’amplifier les avantages précédents. En effet, cela permet d’être

Nom	Propriété de φ
(P1) : Conditions aux limites	$\varphi(0, \dots, 0) = 0$ et $\varphi(1, \dots, 1) = 1$
(P2) : Monotonie (croissance)	si $x'_{n+1} \leq x_{n+1}$, $\varphi(\mathbf{x}_n, x'_{n+1}) \leq \varphi(\mathbf{x}_n, x_{n+1})$
(P3) : Continuité	$\lim_{\mathbf{x}'_n \rightarrow \mathbf{x}_n} \varphi(\mathbf{x}'_n) = \varphi(\mathbf{x}_n)$
(P4) : Commutativité	soit σ une permutation de $[1, n]$, $\varphi(\mathbf{x}_n) = \varphi(x_{\sigma(1)}, \dots, x_{\sigma(n)})$
(P5) : Idempotence	$\varphi(x, \dots, x) = x$
(P6) : Associativité	$\varphi(\varphi(\mathbf{x}_n), \varphi(\mathbf{y}_m)) = \varphi(\mathbf{x}_n, \mathbf{y}_m)$
(P7) : Affaiblissement	$\varphi(\mathbf{x}_n) \leq \min(\mathbf{x}_n)$
(P8) : Renforcement	$\varphi(\mathbf{x}_n) \geq \max(\mathbf{x}_n)$
(P9) : Composition	si $\varphi(\mathbf{x}_n) \leq \varphi(\mathbf{y}_m)$, $\varphi(\mathbf{x}_n, \mathbf{z}) \leq \varphi(\mathbf{y}_m, \mathbf{z})$ quelle que soit la position de $\mathbf{z}$
(P10) : Décomposition	si $\varphi(\mathbf{x}_n, \mathbf{z}) \leq \varphi(\mathbf{y}_m, \mathbf{z})$, $\varphi(\mathbf{x}_n) \leq \varphi(\mathbf{y}_m)$ quelle que soit la position de $\mathbf{z}$

TABLE 5.2 – Propriétés classiques des fonctions d'agrégation. Notations : les variables écrites en gras et indicées par n , comme $\mathbf{x}_n = (x_1, \dots, x_n)$, sont des éléments de $[0, 1]^n$, les autres variables sont des valeurs de $[0, 1]$.

encore plus spécifique lors du choix de chacune des fonctions et cela permet également d'obtenir un niveau supplémentaire d'explication.

Pour ces raisons, nous proposons une nouvelle définition de sémantique graduelle en argumentation que nous appelons *sémantique agrégative graduelle* pour souligner le rôle qu'y jouent les fonctions d'agrégation. Elle suit un principe similaire à celui des sémantiques modulaires, en considérant une modularité accrue : elle exploite la bipolarité des relations d'attaque et de support des QBAF en traitant séparément l'agrégation des poids des attaquants et celle des supporteurs, ce qui constitue une différence fondamentale avec ce qui est fait dans la littérature récente. En effet, cela permet une prise en compte indépendante et potentiellement asymétrique de ces deux composantes. Ce principe de séparation avait déjà été proposé par Cayrol & Lagasque-Schiex (2005a) dans les premiers travaux sur les sémantiques graduelles pour des systèmes d'argumentation bipolaire sans pondération. Cependant, les sémantiques graduelles utilisées par la suite ont fait disparaître cette étape. Enfin, après cette évaluation du poids global des attaquants et des supporteurs, ces deux valeurs sont agrégées avec le poids intrinsèque de l'argument concerné. Ce principe est formalisé dans la définition suivante.

Définition 5.4 (Sémantique agrégative graduelle). *Soit $A = (\mathcal{A}, \mathcal{R}, \mathcal{S}, w) \in \text{WAG}$, $a \in \mathcal{A}$ et I, J des ensembles préordonnés, par exemple $[0, 1]$.*

On définit le degré d'acceptabilité de a comme :

$$\text{Deg}_A(a) = \varphi_f(\varphi_{\mathcal{R}}(\{\text{Deg}_A(x) \mid x \in \text{Att}_a\}), \varphi_{\mathcal{S}}(\{\text{Deg}_A(y) \mid y \in \text{Supp}_a\}), w(a))$$

avec :

- $\varphi_{\mathcal{R}} : \bigcup_n I^n \rightarrow J$, une fonction d'agrégation calculant le poids global des attaquants d'un argument ;

- $\varphi_S : \bigcup_n I^n \rightarrow J$ une fonction d'agrégation calculant le poids global des supporteurs d'un argument;
- $\varphi_f : J^2 \times I \rightarrow I$ une fonction d'agrégation calculant le degré d'acceptabilité d'un argument en fonction du poids global de ses attaquants, de ses supporteurs et de sa force intrinsèque.

On pose de plus $\varphi_R(\emptyset) = 0$ et $\varphi_S(\emptyset) = 0$.

Afin de pouvoir faire référence au poids global des attaquants et des supporteurs, nous proposons les notations supplémentaires suivantes :

- $\pi_R(a) = \varphi_R(\{Deg_A(x) \mid x \in Att_a\})$;
- $\pi_S(a) = \varphi_S(\{Deg_A(y) \mid y \in Supp_a\})$.

Avec cette notation, le degré d'acceptabilité d'un argument a pour une sémantique agrégative graduelle s'écrit :

$$Deg_A(a) = \varphi_f(\pi_R(a), \pi_S(a), w(a)).$$

Dans la suite, on prend $I = J = [0, 1]$.

Remarque – Comme dans le cas modulaire, la définition de sémantique agrégative que nous proposons est récursive, aussi la question de la convergence du calcul de l'acceptabilité dans le cas des graphes cycliques se pose également. Nous ne la considérons pas dans cette thèse, en la laissant pour des travaux futurs, et en concentrant ici la discussion sur l'impact et la pertinence des propriétés des fonctions d'agrégation.

5.3.3 Liens et différences avec les sémantiques modulaires

Comme annoncé dans la section précédente, la définition de sémantique agrégative que nous proposons repose sur le même principe que les sémantiques modulaires, à savoir décomposer le calcul de l'acceptabilité d'un argument en différentes étapes afin d'une part pouvoir contraindre chacune d'entre elles indépendamment et d'autre part obtenir un processus offrant plusieurs niveaux d'explications. Toutefois, contrairement au cas des sémantiques modulaires, nous décomposons le calcul de la fonction α en deux étapes. Attaque et support étant deux relations indépendantes l'une de l'autre, nous souhaitons conserver cette indépendance. Bien qu'il soit possible, par l'intermédiaire de α , d'agréger les attaquants (respectivement supporteurs) de la même manière que φ_R (respectivement φ_S), ces deux valeurs sont ensuite combinées avant d'être agrégées avec w , empêchant un traitement asymétrique entre elles.

À l'inverse, nous pensons qu'il existe une réécriture des sémantiques modulaires en sémantique agrégative. Une telle réécriture est proposée dans la table 5.3 dans le cas particulier de certaines fonctions utilisées dans la littérature. Une première observation est que attaquants et supporteurs sont agrégés collectivement. En utilisant le vocabulaire du formalisme des sémantiques agrégatives, cette agrégation est simplement la différence entre le poids global des supporteurs et celui des attaquants. De plus, ces deux valeurs sont obtenues à partir de la même fonction, c'est-à-dire que $\varphi_R = \varphi_S$.

Nous ne trouvons pas de contre-exemple de sémantiques modulaires impossibles à écrire sous la forme de sémantique agrégative, toutefois, la preuve formelle de ce résultat est un travail en cours. En effet, la reformulation de la fonction α en deux fonctions φ_R

$\alpha = \varphi_S - \varphi_R$	
Somme	$\varphi_{R/S}(b_1, \dots, b_n) = \sum_i b_i$
Produit	$\varphi_{R/S}(b_1, \dots, b_n) = 1 - \prod_i (1 - b_i)$
Maximum	$\varphi_{R/S}(b_1, \dots, b_n) = \max_i(b_i)$
φ_f	
Linear(κ)	$\varphi_f(x, y, w) = w - \frac{w}{\kappa} \max(0, -s) + \frac{1-w}{\kappa} \max(0, s)$ avec $s = y - x$
Euler-based	$\varphi_f(x, y, w) = 1 - \frac{1-w^2}{1+we^s}$ avec $s = y - x$
p -Max(κ)	$\varphi_f(x, y, w) = w \times (1 + h(\frac{s}{\kappa}) - h(-\frac{s}{\kappa}))$ avec $h(x) = \frac{\max(0, x)^p}{1 + \max(0, x)^p}$ et $s = y - x$

TABLE 5.3 – Réécriture des sémantiques modulaires classiques dans le formalisme des sémantiques agrégatives. À l’exception de la somme où $I = \mathbb{R}$, les autres fonctions sont définies de $[0, 1]$ dans $[0, 1]$.

et φ_S nécessite de désentrelacer attaquants et des supporters ce qui est rendu complexe en raison de l’écriture sous forme vectorielle des sémantiques modulaires. D’autre part, comme les propriétés présentées par Mossakowski & Neuhaus (2018); Potyka (2019) ont été écrites spécifiquement pour les sémantiques modulaires dont le formalisme diffère du nôtre, elles nécessitent une telle réécriture. Par conséquent, elles ne sont pas non plus discutées dans ce chapitre.

Cette formalisation ouvre la voie vers de nouvelles sémantiques graduelles pour les QBAF, selon les fonctions d’agrégation choisies pour instancier la définition 5.4. La section suivante examine les propriétés classiques de ces fonctions dans le cadre du calcul du degré d’acceptabilité d’un argument. La section 5.5 examine, réciproquement, la correspondance des axiomes classiques des sémantiques argumentatives avec ces propriétés.

5.4 Discussion sur les propriétés souhaitables des différentes fonctions d’agrégation

Dans cette section, nous examinons différentes propriétés des fonctions d’agrégation dans le cadre de l’argumentation afin de guider l’instanciation de la définition 5.4, c’est-à-dire afin de permettre de choisir les fonctions d’agrégation les plus adaptées parmi la diversité de fonctions existantes. Nous discutons de la pertinence de chacune d’elles dans ce contexte argumentatif et nous cherchons en particulier à identifier, par l’intermédiaire d’exemples et en les mettant en regard avec “l’intuition”, si ces propriétés sont attendues ou si elles sont seulement souhaitées dans certains cas. Pour rappel, leur formalisation est regroupée dans la table 5.2.

Nous commençons par l’agrégation du degré d’acceptabilité des attaquants et de celle des supporters, c’est-à-dire les fonctions φ_R et φ_S . Nous passons ensuite à la fonction φ_f correspondant au calcul du degré d’acceptabilité d’un argument en fonction de sa force intrinsèque, du poids global de ses attaquants et du poids global de ses supporters.

5.4.1 Les cas de $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$

Cette section s'intéresse à la définition du poids global des attaquants et des supporters d'un argument et plus particulièrement aux propriétés souhaitables de $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$. Dans la suite, ces deux fonctions sont regroupées sous la notation commune φ_* avec $* \in \{\mathcal{R}, \mathcal{S}\}$. De plus, à part si la distinction entre attaquants et supporters est nécessaire, nous utilisons le cas des attaquants de façon systématique pour expliquer et discuter des postulats. On considère ici $AF = (\mathcal{A}, \mathcal{R}, \mathcal{S}, w)$ un QBAF et $a \in \mathcal{A}$, un argument.

(P1) : Conditions aux limites de φ_* – Ce postulat traite du comportement attendu pour la fonction aux bornes du domaine de définition, c'est-à-dire qu'il vise à définir les valeurs de $\varphi_*(0)$ et de $\varphi_*(1)$.

En argumentation, si un argument n'est attaqué que par des arguments d'acceptabilité égale à 0, il n'y a pas de raison a priori pour que le poids global des attaquants prenne une autre valeur que la valeur nulle. On attend donc que $\varphi_*(0) = 0$. De la même façon, on attend que $\varphi_*(1) = 1$.

Par ailleurs, le cas limite $\varphi_{\mathcal{R}}(\emptyset) = 0$ posé dans la définition 5.4 est justifié par le fait que pour un argument non attaqué, il est raisonnable de considérer que le poids des attaquants est nul.

(P2) : Croissance de φ_* – Transposé en argumentation, ce deuxième postulat traduit que si l'acceptabilité d'un des attaquants de a augmente alors le poids agrégé des attaquants de ce dernier augmente ou reste constant. Il s'agit du comportement généralement attendu en argumentation (Cayrol & Lagasquie-Schiex, 2005a).

(P3) : Continuité de φ_* – Imposer que $\varphi_{\mathcal{R}}$ satisfasse le postulat de continuité signifie que l'on attend qu'une faible modification de l'acceptabilité d'un des attaquants n'induisse qu'une légère modification sur le poids global des attaquants. Ce comportement peut en effet être considéré comme souhaitable.

Néanmoins, dans certains scénarios, on peut imaginer que des effets de seuil apparaissent : au-delà ou en deçà d'un certain poids global pour les attaquants, il peut y avoir par exemple saturation entraînant un saut brusque qui correspond à une discontinuité. En faisant l'hypothèse d'un nombre au plus dénombrable de tels seuils, un postulat souhaitable devient alors que les fonctions φ_* soient continues par morceaux. Nous donnons un exemple d'un tel phénomène dans la section suivante.

(P4) : Commutativité de φ_* – La pertinence du postulat de commutativité pour φ_* dépend du contexte considéré. En effet, si une fonction vérifie ce postulat alors l'ordre des variables de la fonction n'est pas important. C'est le cas par exemple lors de combinaisons de critères d'importance égale ou dans des procédures de vote. En argumentation abstraite et sans contrainte supplémentaire, cette propriété est même qualifiée de principe qui doit être vérifié par la sémantique graduelle (Cayrol & Lagasquie-Schiex, 2005a).

Cependant, si l'on souhaite tenir compte de la temporalité dans le calcul de l'acceptabilité d'un argument, les attaquants ne jouent plus un rôle symétrique. En effet, l'ordre d'énonciation joue un rôle crucial dans la construction d'un débat : par exemple un argument énoncé au tout début ou à la toute fin d'un débat a généralement plus d'impact que ceux au milieu (Miller & Campbell, 1959). Dans ce cas, les arguments ne peuvent pas être permutés et un poids pourrait être ajouté pour l'agrégation en fonction de la proximité avec le début ou la fin du débat. La propriété de commutativité n'est alors pas souhaitée. De plus, comme nous l'avons vu dans les chapitres 3 et 4, l'ordre d'énonciation

des arguments a également un impact sur les causes effectives du statut d'acceptabilité d'un argument. La non commutativité de φ_* peut ainsi fournir une piste intéressante pour permettre d'en tenir compte lors de la recherche de causes effectives.

Un autre cas où la commutativité peut ne pas être souhaitable est celui des *value-based argumentation frameworks* (Bench-Capon, 2002). Dans ce formalisme, une valeur est ajoutée aux arguments : un argument peut par exemple être d'ordre économique, un autre d'ordre social ou encore d'ordre environnemental. De plus, à chaque agent est associée une relation d'ordre sur ces valeurs. Par exemple, la valeur environnementale est aussi importante que la valeur sociale et plus importante que la valeur économique. Dans ce cas, lorsque le poids global des attaquants d'un argument a est calculé étant donné un agent, on souhaite que les attaquants dont les valeurs sont préférées aient une influence plus importante que les autres. Par conséquent, on ne peut pas permuter les variables de la fonction φ_* , c'est-à-dire que cette dernière ne doit pas être commutative.

(P6) : Associativité de φ_* – Il s'agit principalement d'une caractéristique structurale dont l'intérêt est de pouvoir agréger les données déjà présentes puis ensuite de mettre à jour le résultat avec de nouvelles données qui arrivent. En argumentation, dans le cas où l'ordre d'énonciation des arguments n'est pas pris en compte, toutes les informations sont déjà présentes dans le graphe au moment de l'agrégation. Dans des scénarios d'application en temps réel pour lesquels la vitesse de calcul est importante, utiliser l'associativité de φ_* peut permettre un gain de temps. Cela reste néanmoins négligeable dans le cas de graphes acycliques dont la convergence est de manière évidente garantie en maximum N_{arg} opérations avec N_{arg} le nombre total d'arguments présents dans le graphe (Potyka, 2019). De plus, s'agissant d'une propriété très restrictive (Grabisch *et al.*, 2009), cette propriété n'est en général pas requise.

(P5) : Idempotence de φ_* – La propriété d'idempotence est généralement souhaitable dans le cas où les sources des valeurs à agréger ne sont pas indépendantes : il y a alors redondance de l'information, elle est donc résumée en une valeur. À l'inverse, si les sources sont indépendantes, alors des informations redondantes peuvent se renforcer ou s'affaiblir. Nous discutons de cette propriété au point suivant.

En argumentation, la notion de redondance de l'information dépend de la nature des arguments et du contexte modélisé. Un exemple d'une telle situation est un débat au cours duquel un argument est répété par plusieurs personnes différentes. Dans le cas où l'on souhaite tenir compte de cette répétition pour souligner la pertinence ou bien l'influence de cet argument sur les orateurs, l'idempotence n'est pas souhaitable. En effet, si elle est imposée, un argument attaqué par un seul argument d'acceptabilité x et un autre attaqué par plusieurs arguments de même acceptabilité x auraient alors le même poids global pour ses attaquants.

De plus, si φ_* est associative et idempotente, la répétition d'un argument n'a plus d'impact. En effet, $\varphi_*(x, x, y) = \varphi_*(\varphi_*(x, x), y) = \varphi_*(x, y)$. Plus généralement, énoncer des arguments dont les degrés d'acceptabilités sont identiques n'a aucune utilité et ce peu importe la nature de ces derniers. De fait, la négation d'au moins un de ces deux postulats est nécessaire.

(P7-8) : Affaiblissement et renforcement de φ_* – Les postulats d'affaiblissement et de renforcement sont des propriétés permettant de caractériser le comportement d'une fonction d'agrégation en présence de valeurs *faibles* ou de valeurs *fortes*, où le sens de fortes et faibles varie selon le contexte. En argumentation, ce postulat traduit l'influence que vont avoir des arguments ayant un degré d'acceptabilité moins (respectivement plus)

élevé que les autres attaquants ou supporteurs d'un même argument. Formellement, le poids global des attaquants est majoré par le degré d'acceptabilité de l'attaquant le plus faible.

Une propriété liée à celle d'affaiblissement existe en argumentation dans le cas des arguments dont le degré d'acceptabilité est proche des bornes du domaine de définition. En effet, l'axiome de monotonie (A7) exprime le fait qu'un argument moins attaqué ou plus soutenu, toutes choses égales par ailleurs, a une acceptabilité plus forte. Il implique qu'ajouter un attaquant quel que soit son degré d'acceptabilité, ne peut qu'augmenter ou ne pas modifier le poids global des attaquants. Si de plus, l'axiome de neutralité (A6) est vérifié, le comportement de la sémantique à l'égard des arguments "inutiles" est l'ignorance, c'est-à-dire que pour un attaquant dont le degré d'acceptabilité est nul tout se passe comme si cet argument n'était pas présent. Nous pensons que ce comportement n'est pas toujours souhaitable et dépend notamment de l'attaquant supplémentaire : un argument "stupide" pourrait desservir son propre camp en faisant diminuer la confiance envers le camp des attaquants. Ainsi, une autre possibilité est que l'ajout d'un tel argument peut affaiblir l'ensemble des attaquants, c'est-à-dire diminuer la valeur du poids global de l'ensemble d'arguments. Ce cas est illustré avec l'exemple ci-dessous par le débat sur la forme de la planète Terre.

Exemple 10 (Plate ou ronde?). *Deux personnes discutent pour savoir si la Terre est ronde ou plate. L'une d'elles pense que la Terre est ronde (a) mais elle n'en est pour autant pas certaine. Son interlocuteur lui explique alors que lorsque l'on fabrique un plan horizontal et qu'on le pose sur le sol, celui-ci est parfaitement solide (b). L'argument lui semble à première vue raisonnable et donc sans autre argument, l'acceptabilité de a baisse. Mais, l'autre personne ajoute un deuxième argument : de toute façon si la Terre était ronde, les humains situés dans l'autre hémisphère seraient tous déjà tombés (c).*

Dans ce cas-là, l'argument c pourrait avoir deux conséquences possibles. Il pourrait d'une part être ignoré, c'est-à-dire que le poids global des attaquants et donc l'acceptabilité de a dans le deuxième cas reste inférieure ou égale à celle dans le premier cas. Le postulat d'affaiblissement n'est alors pas souhaitable. Cependant, l'argument étant si absurde, il pourrait aussi impacter négativement le camp des attaquants en le discréditant, ayant pour conséquence de faire baisser le poids global de ces derniers. Dans ce cas, le postulat d'affaiblissement est attendu.

De la même façon, des arguments "brillants" peuvent se renforcer mutuellement donnant un poids global encore plus important à leur ensemble. C'est ce que traduit le postulat de renforcement. Bien que ces deux propriétés ne soient pas compatibles entre elles, il existe des opérateurs comme par exemple les uninormes qui présentent un comportement hybride. Le domaine de définition est alors découpé en plusieurs zones selon lesquelles la fonction vérifie la propriété d'affaiblissement ou de renforcement ou aucune des deux.

Remarque – Dans le cas où l'opérateur est également monotone et idempotent alors les seules fonctions possibles sont les fonctions min et max. En effet, en notant $\mathbf{x}_n = (x_1, \dots, x_n)$, $x = \min(\mathbf{x}_n)$ et $\mathbf{x} = \{x, \dots, x\}$ on a : $x = \varphi(\mathbf{x})$ par idempotence. Or par monotonie, $\varphi(\mathbf{x}) \leq \varphi(\mathbf{x}_n)$. Enfin, par affaiblissement, $\varphi(\mathbf{x}_n) \leq x$. On a donc $x \leq \varphi(\mathbf{x}_n) \leq x$ c'est-à-dire $\varphi(\mathbf{x}_n) = x = \min_{i \in [1, n]} (x_i)$. De même façon, si φ est monotone et idempotente et vérifie le

postulat de renforcement alors $\varphi(\mathbf{x}_n) = \max(\mathbf{x}_n)$.

(P9) : Composition de φ_* – Aucun des postulats précédents n'impose de comportement dans le cas où un argument est ajouté. De fait, si par exemple un argument c attaquant à la fois un argument a et un argument b est ajouté, alors il se peut qu'il inverse le rapport de forces entre l'ensemble des attaquants de chaque argument. Dans le cas où tous les arguments ne sont pas déjà présents dans le graphe et que ces derniers sont ajoutés séquentiellement, ce n'est pas un comportement souhaitable. En effet, dans le scénario le plus extrême c pourrait faire augmenter le poids global des attaquants de a jusqu'à celui de b mais jamais le dépasser s'il était auparavant inférieur. Cet argument étant présent dans les deux ensembles d'attaquants et l'ensemble des attaquants de b étant initialement plus fort que celui de a , l'ensemble des attaquants de b doit rester au moins aussi fort que celui de a . Cette propriété est traduite par le postulat de composition, qui apparaît donc souhaitable.

(P10) : Décomposition de φ_* – De façon réciproque au postulat de composition, le postulat de décomposition implique que si un attaquant de deux arguments est retiré alors l'ordre entre le poids global des attaquants de chacun de ces deux arguments est conservé. Dans le cas où la temporalité est prise en compte, un tel scénario apparaît si l'on souhaite modéliser un mécanisme d'oubli par le retrait d'argument. Un argument ayant été énoncé "longtemps" auparavant peut être oublié au bout d'un certain temps s'il n'a pas été réénoncé. Dans ce cas, l'argument disparaît et est donc retiré des valeurs à agréger.

Un autre contexte dans lequel cette situation peut se présenter est celui où l'on considère 0 comme élément neutre des fonctions φ_* . Avec cette hypothèse, rendre un argument inacceptable, c'est-à-dire que son acceptabilité devient nulle, revient à retirer l'argument de l'agrégation. En effet, comme 0 est élément neutre de φ_* , alors on a $\varphi_*(\mathbf{x}_n, 0) = \varphi_*(\mathbf{x}_n)$. Dans ces deux situations, l'argument étant retiré des deux ensembles d'arguments, il est attendu que l'ordre entre la force globale de chacun des ensembles soit conservé, ce qui correspond à vérifier le postulat de décomposition.

Bilan – Les conditions aux limites, la croissance des fonctions ainsi que la composition et la décomposition sont les postulats qu'il nous semble souhaitable indépendamment du contexte. L'associativité n'étant pas une propriété nécessairement attendue n'est jamais souhaitable dans les cas où le postulat d'idempotence est vérifié. Les postulats de continuité, de commutativité, d'idempotence, d'affaiblissement et de renforcement dépendent vraiment des cas d'application de la sémantique agrégative.

5.4.2 Le cas de φ_f

Cette sous-section discute des postulats souhaitables pour la fonction d'agrégation φ_f qui calcule le degré d'acceptabilité d'un argument en fonction du poids global de ses attaquants, de ses supporters et de son poids intrinsèque. Contrairement à φ_* , φ_f admet exactement trois variables. Par conséquent, les postulats décrivant le comportement de la fonction en ajoutant ou retirant un argument, c'est-à-dire l'associativité, la composition et la décomposition, ne s'appliquent pas.

(P2) : Monotonie de φ_f – Chacune des variables de φ_f a une signification particulière, à l'inverse de φ_* . La première correspond au poids des attaquants d'un argument : plus elle est élevée, plus l'acceptabilité de l'argument est faible, appelant une fonction

décroissante. La deuxième et la troisième variables de la fonction correspondent au poids des supporteurs et à la force intrinsèque de l'argument. Plus l'une d'elles est élevée, plus l'acceptabilité est élevée, imposant des fonctions croissantes. Ainsi, il est attendu que φ_f soit décroissante selon sa première variable et croissante selon les deux dernières.

(P1) : Conditions aux limites de φ_f – En raison de la monotonie de φ_f , ses conditions aux limites diffèrent de celles présentées dans la table 5.2. Elles s'écrivent :

$$\varphi_f(0, 1, 1) = 1 \text{ et } \varphi_f(1, 0, 0) = 0.$$

Nous ajoutons également la condition : $\varphi_f(0, 0, w) = w$. Il s'agit d'une propriété classique en argumentation correspondant à l'axiome de stabilité **(A5)** : un argument qui n'est ni attaqué, ni soutenu a une acceptabilité égale à son poids intrinsèque.

(P3) : Continuité de φ_f – Comme pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$, imposer la continuité de φ_f dépend du choix ou non d'inclure des effets de seuil et de saturation dans la modélisation. Prenons par exemple le cas de différents témoignages confirmant un alibi d'un suspect potentiel dans une affaire judiciaire. Tant qu'un certain nombre de témoignages d'une certaine qualité n'ont pas été énoncés, c'est-à-dire tant que le poids global des supporteurs n'a pas atteint une certaine valeur, le juge ne considère pas le suspect comme complètement innocent. Par conséquent, les différentes attaques à l'égard du suspect sont prises en compte et viennent baisser le niveau d'innocence de la personne, c'est-à-dire le degré d'acceptabilité de l'argument. Cependant, au moment où suffisamment de témoignages ont été apportés, le suspect devient immédiatement innocent, et l'acceptabilité de l'argument passe alors à 1. Dans ce cas, la fonction φ_f n'est pas continue.

(P4) : Commutativité de φ_f – Les variables de φ_f n'ayant pas un rôle symétrique et ayant chacune leur propre signification, la commutativité n'est pas une propriété désirable. En effet, si permuter le poids des attaquants avec le poids des supporteurs n'a aucun effet sur l'acceptabilité d'un argument, cela signifie que faire une distinction entre attaquant et supporter n'apporte rien à la modélisation et la bipolarité n'aurait alors aucun intérêt.

(P5) : Idempotence de φ_f – La propriété d'idempotence appliquée à φ_f est un cas particulier d'une propriété plus générale correspondant à une forme de symétrie entre les attaquants et les supporteurs. En effet, considérer que $\varphi_f(x, x, x) = x$ peut être justifié par le fait que les attaquants et les supporteurs jouent un rôle symétrique et donc se compensent. Dans ce cas, cette propriété se généralise sous la forme :

$$\forall x, w \in J \times I, \varphi_f(x, x, w) = w.$$

(P7-8) : Affaiblissement et renforcement de φ_f – En raison de la monotonie de φ_f , le postulat d'affaiblissement devient : un poids global des attaquants "élevé" et des valeurs "faibles" pour le poids des supporteurs et le poids de l'argument s'affaiblissent pour donner une acceptabilité encore plus faible. De la même manière, un argument "fort", "faiblement" attaqué et "fortement" soutenu aura une valeur d'acceptabilité renforcée, c'est-à-dire encore plus forte. Avec un comportement hybride, φ_f étale les valeurs d'acceptabilité permettant d'obtenir plus de contraste entre arguments forts et faibles. Dans les cas où une décision est requise, ce comportement peut permettre de limiter le nombre de choix possibles.

Bilan – Les conditions aux limites ainsi que la décroissance de φ_f selon la première variable et sa croissance selon les deuxième et troisième variables sont des postulats attendus quel que soit le contexte. Le postulat de commutativité n'est lui jamais souhaitable. Enfin, la continuité, l'idempotence, l'affaiblissement et le renforcement sont des postulats qui dépendent fortement du contexte.

5.5 Comparaison formelle avec les axiomes classiques des QBAF

Dans cette section, on considère Deg_A une sémantique agrégative comme spécifiée dans la définition 5.4 de fonctions d'agrégation φ_R, φ_S et φ_f . Nous revenons sur les axiomes des QBAF présentés dans la section 5.2.3 et discutons de leur lien avec les propriétés de la section 5.4. Nous distinguons en particulier deux types d'axiomes : ceux qui ont un lien avec les postulats des fonctions d'agrégation pour lesquels nous établissons des correspondances, et ceux qui décrivent des propriétés impliquant les trois fonctions d'agrégation, imposant un choix conjoint de φ_R, φ_S et φ_f . Pour cette deuxième catégorie d'axiomes, nous en proposons une réécriture dans le formalisme des sémantiques agrégatives graduelles. Comme dans les chapitres précédents, les preuves complètes des propositions suivantes sont rédigées dans l'annexe D.1.

5.5.1 Lien entre axiomes argumentatifs et propriétés des opérateurs d'agrégation

La définition d'une sémantique agrégative impose une structure particulière pour calculer le degré d'acceptabilité d'un argument. En effet, seuls les degrés d'acceptabilité des attaquants directs et des supporteurs directs d'un argument ainsi que le poids de l'argument lui-même sont pris en compte pour ce calcul. Pour cette raison, certains axiomes de l'argumentation sont vérifiés sans hypothèse supplémentaire sur les fonctions d'agrégation.

Proposition 5.1. *Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Alors Deg_A vérifie les axiomes (A1), (A2), (A3) et (A5).*

Principe de la preuve. Pour (A1), la preuve est construite en démontrant la propriété pour les arguments non attaqués et non supportés. De tels éléments existent car le graphe est acyclique. Ensuite, on construit l'ensemble des arguments vérifiant ces propriétés en remontant le graphe jusqu'à obtenir tous les arguments. Cela est possible car il y a un nombre fini d'arguments et que le graphe est acyclique.

Pour (A2) et (A3), l'ensemble des attaquants et des supporteurs étant le même pour les deux arguments a et b étudiés, les trois valeurs finales à agréger sont identiques dans les QBAF et donc le résultat est identique.

Pour (A5), il s'agit d'une conséquence directe des conditions aux limites que nous imposons.

□

Proposition 5.2 (Équivalence). *Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Si φ_R et φ_S sont commutatives alors Deg_A vérifie l'axiome d'équivalence.*

Principe de la preuve. Comme $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ sont commutatives, alors avec f et f' des bijections sur l'ensemble des attaquants et des supporteurs de a et b , on a directement que $\pi_{\mathcal{R}}(a) = \pi_{\mathcal{R}}(f(a)) = \pi_{\mathcal{R}}(b)$. Cela est également vraie pour les supporteurs ce qui conduit à $\text{Deg}_A(a) = \text{Deg}_A(b)$. $\square$

L'axiome suivant est celui de neutralité (A6). Il s'intéresse à l'effet des attaquants et des supporteurs d'acceptabilité nulle. Ils sont qualifiés d'inutiles (Amgoud & Ben-Naim, 2018) et ne doivent donc pas avoir d'effet sur l'acceptabilité d'un argument : dans le cadre d'une sémantique agrégative, ils ne doivent pas modifier la valeur du poids des attaquants ou des supporteurs. Ce n'est pas le cas par défaut et nécessite d'imposer que 0 est un élément neutre pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$.

Proposition 5.3 (Neutralité). *Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Si la valeur 0 est un élément neutre pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ alors Deg_A vérifie l'axiome de neutralité.*

Principe de la preuve. La preuve est directe en examinant le cas où l'argument d'acceptabilité nul est un attaquant ou un supporter. $\square$

Les deux axiomes suivants sont la monotonie (A7) et le renforcement (A8). Ils concernent l'évolution du degré d'acceptabilité quand on ajoute des attaquants ou des supporteurs (axiome de monotonie) ou quand on modifie l'acceptabilité de l'un d'entre eux (axiome de renforcement).

Proposition 5.4 (Renforcement). *Soit $A \in WAG$, un QBAF acyclique et Deg_A une sémantique agrégative. Si $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ sont croissantes (respectivement strictement croissantes) et φ_f est décroissante selon x et croissante selon y (respectivement strictement monotone en x et en y) alors Deg_A vérifie l'axiome de renforcement (respectivement renforcement strict) au sens de l'argumentation.*

Principe de la preuve. La preuve est immédiate en écrivant le calcul. $\square$

Pour l'axiome de monotonie, une hypothèse supplémentaire est nécessaire : 0 doit être élément neutre. Cela vient du fait qu'il n'y a aucun postulat des fonctions d'agrégation qui explicite le comportement dans tous les cas quand un argument est ajouté. En effet, le postulat de composition (P9) concerne l'ajout d'un même ensemble d'arguments à deux ensembles. D'autre part, les postulats d'affaiblissement et de renforcement donnent une borne inférieure ou supérieure pour le résultat mais ne fournissent pas d'informations sur l'évolution par rapport à l'absence de cet argument. Or si 0 est élément neutre, on peut écrire $\forall z \in [0, 1], \varphi_{\mathcal{R}}(\mathbf{x}_n) = \varphi_{\mathcal{R}}(\mathbf{x}_n, 0) \leq \varphi_{\mathcal{R}}(\mathbf{x}_n, z)$.

Proposition 5.5 (Monotonie). *Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Si 0 est un élément neutre pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ et que les trois fonctions d'agrégation sont monotones (respectivement strictement monotones), alors Deg_A vérifie l'axiome de monotonie (respectivement de monotonie stricte).*

Principe de la preuve. Comme 0 est élément neutre et que φ_* est croissante, ajouter un attaquant (respectivement un supporter) ne peut pas diminuer le poids global des attaquants (respectivement des supporteurs). De plus si les fonctions sont strictement monotones alors cela ne peut qu'augmenter cette valeur. Avec cette propriété, la preuve est immédiate en écrivant simplement les inégalités entre les différentes fonctions. $\square$

Pour l'axiome de résilience (A9), s'agissant d'une propriété portant exclusivement sur φ_f , nous proposons plutôt une reformulation. Cet axiome donne un rôle particulier aux valeurs extrêmes des forces intrinsèques et finales d'un argument. En effet, les degrés d'acceptabilité maximum et minimum ne peuvent être atteints que par des arguments ayant déjà cette valeur comme force intrinsèque. Ainsi, un argument d'acceptabilité égale à 0.99 mais dont la force intrinsèque est différente de 1 ne pourra jamais être totalement accepté, quelle que soit la qualité de ses supporters. Il s'agit d'un principe optionnel dont l'intérêt dépend de la nature des arguments (Amgoud & Ben-Naim, 2018). Une reformulation de cet axiome pour une sémantique agrégative est assez immédiate : $\forall z \in]0, 1[, \forall x, y \in [0, 1]^2, 0 < \varphi_f(x, y, z) < 1$.

5.5.2 Lien entre axiomes argumentatifs et interactions des opérateurs d'agrégation

Les trois derniers axiomes n'ont pas de lien avec les postulats des fonctions d'agrégation. En particulier, contrairement à ces postulats, les axiomes (A10), (A11) et (A12) donnent des informations sur les interactions souhaitées entre ces trois fonctions. Par exemple, que se passe-t-il si un attaquant et un supporter de degrés d'acceptabilité égaux sont ajoutés simultanément ? De même, si les attaquants sont individuellement plus forts que les supporters, comment évolue le degré d'acceptabilité par rapport à son poids intrinsèque ? Les axiomes de Franklin, d'affaiblissement et de consolidation proposent de telles règles. Afin de guider au mieux le choix des trois fonctions d'agrégation, nous proposons de réécrire ces trois derniers axiomes dans le formalisme des sémantiques agrégatives.

Principe de Franklin (A10) – Il est décrit par Amgoud & Ben-Naim (2018) comme un principe indiquant la manière dont les attaquants et les supporters d'un argument doivent être agrégés. En particulier, cet axiome régit les situations dans lesquelles un attaquant et un supporter de même acceptabilité sont ajoutés. Dans le cas du principe de Franklin au sens large, l'attaquant domine le supporter et donc l'acceptabilité globale diminue (principe de Franklin). Dans le cas strict, attaquants et supporters se compensent et donc la valeur d'acceptabilité ne change pas (principe de Franklin strict). Dans notre formalisme où l'asymétrie est permise, l'ajout d'un attaquant et d'un supporter de même force peut ne pas avoir le même impact sur le poids global des attaquants et des supporters. Néanmoins, quelles que soient ces nouvelles valeurs, si l'axiome de Franklin est vérifié nous avons une information sur le sens de variation du degré d'acceptabilité. En réécrivant ce principe avec notre formalisme et en reprenant les notations de la table 5.2 on obtient :

$$\varphi_f(\varphi_{\mathcal{R}}(\mathbf{x}_n, x), \varphi_{\mathcal{S}}(\mathbf{y}_m, x), w) \leq \varphi_f(\varphi_{\mathcal{R}}(\mathbf{x}_n), \varphi_{\mathcal{S}}(\mathbf{y}_m), w)$$

et égalité dans le cas strict. On peut vouloir échanger le rôle de l'attaque et du support dans la propriété précédente, c'est-à-dire que le support domine sur l'attaque. Nous appelons cet axiome le Principe de Franklin inverse.

Affaiblissement (A11) – Informellement, l'axiome d'affaiblissement s'intéresse au cas où les attaquants "dominent" les supporters. Dans cette situation, le degré d'acceptabilité doit être inférieur strictement au poids de l'argument. En réécrivant formellement cet axiome dans notre formalisme on a :

Soit $a \in \mathcal{A}$ tel que $w(a) > 0$, et il existe f une fonction injective de Supp_a vers Att_a

telle que $\forall x \in \text{Supp}_a, \text{Deg}_A(x) \leq \text{Deg}_A(f(x))$. En notant $\text{Supp}_a = \mathbf{x}_m$ et $\text{Att}_a = \mathbf{y}_n$, si $\exists x_i$ tel que $\text{Deg}_A(x_i) < \text{Deg}_A(f(x_i))$ ou $\exists y_j$ tel que $f^{-1}(y_j) = \emptyset$ et $\text{Deg}_A(y_j) > 0$ alors on a :

$$\varphi_f(\varphi_{\mathcal{R}}(\{\text{Deg}_A(y_i) \mid y_i \in \mathbf{y}_n\}), \varphi_{\mathcal{S}}(\{\text{Deg}_A(x_i) \mid y_i \in \mathbf{x}_m\}), w(a)) < w(a).$$

Il s'agit bien d'une propriété couplant les trois fonctions d'agrégation. En effet, on pourrait tenter de reformuler cet axiome en disant que si les attaquants "dominent" les supporteurs alors le poids global des attaquants doit être supérieur à celui des supporteurs (ce faisant on lie déjà le choix de $\varphi_{\mathcal{R}}$ à celui de $\varphi_{\mathcal{S}}$). Ensuite, si le poids global des attaquants est supérieur à celui des supporteurs alors le degré d'acceptabilité de l'argument est inférieur à sa force intrinsèque. Cependant, cette reformulation n'est pas équivalente à la précédente. Considérons comme fonction d'agrégation pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ la fonction min, alors malgré l'existence d'une telle injection entre les deux ensembles d'arguments, si on a $\min(\mathbf{y}_j) < \min(\mathbf{x}_i)$, le poids global des attaquants sera plus faible que celui des supporteurs. Si par ailleurs on choisit pour φ_f une moyenne pondérée qui accorde plus de poids à l'attaque qu'au support, alors on peut avoir $\text{Deg}_A(a) < w(a)$. L'axiome d'affaiblissement est donc bien vérifié bien que le poids global des attaquants soit plus faible que celui des supporteurs.

Consolidation (A12) – De même, l'axiome de consolidation régit le cas où les supporteurs "dominent" les attaquants et impose que l'acceptabilité de l'argument soit supérieure strictement à son poids intrinsèque. Il se traduit dans le formalisme des sémantiques agrégatives de la façon suivante :

Soit $a \in \mathcal{A}$ tel que $w(a) > 0$, et il existe f une fonction injective de Att_a vers Supp_a telle que $\forall x \in \text{Att}_a, \text{Deg}_A(x) \leq \text{Deg}_A(f(x))$. En notant $\text{Supp}_a = \mathbf{x}_m$ et $\text{Att}_a = \mathbf{y}_n$, si $\exists x_i$ tel que $\text{Deg}_A(x_i) < \text{Deg}_A(f(x_i))$ ou $\exists y_j$ tel que $f^{-1}(y_j) = \emptyset$ et $\text{Deg}_A(y_j) > 0$ alors on a :

$$\varphi_f(\varphi_{\mathcal{R}}(\{\text{Deg}_A(y_i) \mid y_i \in \mathbf{y}_n\}), \varphi_{\mathcal{S}}(\{\text{Deg}_A(x_i) \mid y_i \in \mathbf{x}_m\}), w(a)) > w(a).$$

Ces trois axiomes permettent de connaître le comportement global de la sémantique graduelle par rapport aux ensembles d'attaquants et de supporteurs. Dans le cas d'une sémantique agrégative, ils imposent que les trois fonctions soient choisies conjointement et non indépendamment l'une de l'autre.

5.5.3 Discussion sur les sémantiques graduelles classiques

Dans cette section, nous reprenons les sémantiques graduelles les plus couramment utilisées dans la littérature dont la réécriture dans le formalisme des sémantiques agrégatives est donnée dans la table 5.3. Nous discutons des postulats vérifiés par leur implémentation dans le formalisme des sémantiques agrégatives. Ces résultats sont présentés dans la table 5.4.

Postulats vérifiés par $\varphi_{\mathcal{R}}/\varphi_{\mathcal{S}}$ – Une première remarque concerne le domaine de définition des fonctions utilisées : dans tout ce chapitre, nous avons considéré $I = J = [0, 1]$. Dans le cas de la fonction somme pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$, on a $I = \mathbb{R}$. Ce domaine de définition rend les conditions aux limites (P1) impossibles à satisfaire. Cette fonction est par exemple celle choisie pour les sémantiques *Euler-based* et *Quadratic Energy* définies dans la section 5.2.2. Concernant les autres postulats pour le calcul du poids global des attaquants et des supporteurs, la seule différence porte sur l'idempotence (P5). En effet,

$\varphi_{\mathcal{R}}/\varphi_{\mathcal{S}}$	(P1)	(P2)	(P3)	(P4)	(P5)	(P6)	(P7)	(P8)	(P9)	(P10)
Somme	×	✓	✓	✓	×	✓	×	✓	✓	✓
Produit	✓	✓	✓	✓	×	✓	×	✓	✓	✓
Maximum	✓	✓	✓	✓	✓	✓	×	✓	✓	✓
φ_f										
Linear(κ)	✓	✓	✓	×	✓		×	×		
Euler-based	✓	✓	✓	×	✓		×	×		
p -Max(κ)	✓	✓	✓	×	✓		×	×		

TABLE 5.4 – Postulats vérifiés (✓) ou non vérifiés (×) par différentes fonctions d’agrégation utilisées dans les sémantiques graduelles de la littérature.

la fonction `max` est idempotente contrairement aux fonctions somme et produit. Cela permet selon le contexte de tenir compte ou non de la répétition d’un argument.

Cependant, au regard de la discussion que nous avons proposée dans la section 5.4, certains postulats sont manquants. Tout d’abord, aucune de ces fonctions est non commutative (P4). Or il s’agit d’une propriété importante dans le cas où l’on souhaite tenir compte d’un ordre sur les arguments. Il existe plusieurs fonctions d’agrégation non commutative dans la littérature comme par exemple les intégrales de Choquet dont les capacités ne sont pas symétriques (Choquet, 1954). Concernant l’affaiblissement, les t -normes sont une famille d’opérateurs vérifiant cette propriété (Menger, 1942). Par exemple, la fonction `min` ou encore la t -norme de Łukasiewicz définie par $\varphi_*(\mathbf{x}_n) = \max(\sum_{i=1}^n x_i - n, 0)$ appartiennent à cette famille. Enfin, dans la littérature on observe une gestion essentiellement symétrique des attaquants et des supporters pour le calcul d’un poids global des relations d’un argument.

Postulats vérifiés par φ_f – On n’observe aucune diversité sur les propriétés vérifiées parmi les trois fonctions principalement utilisées dans la littérature. En particulier, pour les trois postulats pour lesquels il y a un choix à faire, ces trois fonctions sont idempotentes (et vérifient $\varphi_f(x, x, w) = w$) et ne satisfont ni l’affaiblissement ni le renforcement. Pour l’idempotence (P5), cela vient du fait qu’une étape préliminaire calculant la différence entre le poids des attaquants et des supporters est réalisée et que 0 est élément neutre de la fonction d’influence i correspondante.

Exemple 9 (Le jeu d’échecs aux JO). *(suite)* – Afin d’illustrer le genre d’asymétrie que l’on peut obtenir avec le formalisme des sémantiques agrégatives, considérons la formalisation de deux agents aux caractères très différents : Alice, très versatile, c’est-à-dire changeant assez facilement d’opinions sur un sujet et Bob, têtue et qui change difficilement d’avis.

Nous proposons de considérer qu’une première composante de la versatilité est l’affaiblissement : la présence d’un supporter un peu faible suffit à affaiblir l’ensemble des

supporters. Nous prenons donc $\varphi_S^{Alice} = \min$. Inversement, Bob n'entend que les arguments le supportant qui sont forts, et on pose donc $\varphi_S^{Bob} = \max$.

D'autre part, à chaque fois que Alice entend un argument supplémentaire contre elle, le poids global des attaquants augmente. Nous cherchons ici une fonction non idempotente et vérifiant le renforcement. Prenons par exemple le produit : $\varphi_R^{Alice} = 1 - \prod_i (1 - x_i)$.

Au contraire, pour Bob si un attaquant faible est énoncé, il diminue la force globale des attaquants, c'est-à-dire que φ_R^{Bob} vérifie le postulat d'affaiblissement, par exemple $\varphi_R^{Bob} = \min$.

Enfin, il reste à choisir la fonction φ_f . Comme nous en avons discuté dans la section 5.5, si les axiomes (A10), (A11) et (A12) sont importants, il faut choisir conjointement les trois fonctions. Dans le cas présent, le choix des fonctions φ_* couplé avec la monotonie de φ_f suffit pour que $\text{Deg}_{A,Alice}$ vérifie le principe de Franklin. De même, $\text{Deg}_{A,Bob}$ vérifie le principe de Franklin en inversant le rôle de l'attaque et du support. Ces deux comportements correspondent bien à l'asymétrie que nous cherchons à obtenir sur le traitement des attaquants et des supporters.

Concernant les axiomes d'affaiblissement et de consolidation, le fait que pour Bob (respectivement Alice) le calcul du poids des attaquants (respectivement des supporters) vérifie le postulat d'affaiblissement (respectivement de renforcement) implique que si φ_f vérifie le postulat de monotonie ainsi que la formule $\varphi_f(x, x, w) = w$ alors φ_f vérifie l'axiome de consolidation (respectivement affaiblissement). On peut donc prendre une des trois fonctions de la littérature dans cet exemple. Prenons la fonction Ebs :

- Pour Bob : $\text{Deg}_A(a) = 0.60$ donc les échecs doivent être ajoutés aux JO 2028 ;
- Pour Alice : $\text{Deg}_A(a) = 0.49$ donc les échecs ne doivent pas être ajoutés aux JO 2028.

Bien qu'avec un seuil à 0.5 la décision finale soit différente entre Alice et Bob, la différence n'est finalement pas si significative entre les valeurs observées. Cela s'explique par la présence d'un unique support pour l'argument a ce qui réduit l'impact de l'asymétrie entre attaquants et supporters car dans les deux scénarios, la force globale des supporters de a est égale au degré d'acceptabilité de d (0.84 pour Alice, contre 0.89 pour Bob). De même, b et e ne possèdent qu'un unique attaquant et un unique supporter pour e ce qui réduit une fois de plus l'impact de l'asymétrie.

Comme nous l'avons illustré dans la section 5.2.2, le choix de la sémantique graduelle peut conduire à des résultats différents. Cependant, dans le cas des sémantiques agrégatives, il est possible de contrôler ces différences en ayant des justifications de modélisation augmentant ainsi l'interprétabilité de la méthode. Cela ouvre la voie pour la génération d'explications causales compréhensibles et adaptées au destinataire.

5.6 Génération d'explications dans le cadre d'un QBAF : retour aux modèles structurels causaux

Dans les sections 5.3, 5.4 et 5.5, nous avons proposé une nouvelle famille de sémantiques graduelles. La particularité de cette famille de fonction est de décomposer le calcul du degré d'acceptabilité d'un argument en plusieurs étapes. L'objectif derrière cette proposition est d'obtenir un processus de calcul plus compréhensible pour un non expert.

Elle permet en effet de générer des explications à plusieurs niveaux : en se basant sur les poids globaux des attaquants et des supporteurs, puis en descendant d'un niveau si nécessaire en utilisant les degrés d'acceptabilité des attaquants et des supporteurs jusqu'à finalement atteindre la force intrinsèque des arguments. La prochaine étape est donc de proposer une méthode permettant de générer des explications causales pour des QBAF acycliques. Dans ce but, nous proposons d'utiliser le formalisme des graphes causaux argumentatifs (GCA) comme dans le chapitre 2.

Ainsi, dans cette section, nous revenons sur la transformation $\mathcal{T}_{AFa2GCA}$ proposée dans le chapitre 2 des AAF en GCA. Elle repose sur l'hypothèse que les variables sont des variables binaires, correspondant au statut d'acceptabilité des arguments. Dans le cas d'un AAF acyclique, une caractérisation avec deux états possibles, acceptable et inacceptable, suffit pour déterminer l'état final des arguments. Dans le cas des QBAF, le statut d'acceptabilité des arguments est numérique. D'autre part, un QBAF possède un élément de plus dans son graphe associé par rapport à un AAF qu'il faut inclure dans la transformation : les étiquettes sur les nœuds, qui représentent la force intrinsèque des arguments. Il est donc nécessaire de modifier la transformation pour l'adapter à ce nouveau cadre. Dans cette section, nous proposons tout d'abord une telle transformation. Ensuite, nous soulevons la question de l'objet sur lequel doit porter l'explication.

5.6.1 Passage des QBAF aux modèles structurels causaux (MSC)

Comme dans le cas des AAF, la définition d'un modèle structurel causal correspondant à un QBAF nécessite de définir trois éléments :

- $\mathcal{U}$ les variables exogènes dont les valeurs sont déterminées par des facteurs extérieurs au modèle ;
- $\mathcal{V}$ les variables endogènes dont les valeurs sont obtenues grâce aux équations structurelles ;
- $\mathcal{F}$ les équations structurelles.

Dans un AAF, les facteurs extérieurs correspondent à une instanciation du graphe d'argumentation, c'est-à-dire au "choix" des arguments présents. Pour cette raison, les variables exogènes de la transformation $\mathcal{T}_{AFa2GCA}$ définie dans le chapitre 2, ont une valeur binaire qui indique pour chaque argument sa présence ou son absence dans le graphe d'argumentation. Dans le cadre des QBAF, une information supplémentaire est nécessaire en cas de présence d'un argument : sa force intrinsèque. Nous proposons donc de modifier les valeurs prises par les variables exogènes définies dans la transformation $\mathcal{T}_{AFa2GCA}$ en passant d'une valeur binaire à une valeur numérique représentant la force intrinsèque de l'argument. En prenant comme domaine de définition l'intervalle $[0, 1]$ par exemple, la valeur 0 correspond à un argument non présent. Une telle convention pose toutefois un problème dans le cas de fonctions vérifiant le postulat d'affaiblissement. En effet, un argument absent aurait alors pour effet de diminuer le poids global des attaquants ou des supporteurs auxquels il appartient. Pour éviter ce problème, nous introduisons une valeur hors échelle $\perp$, telle que $w(a) = \perp$ si et seulement si a n'est pas présent et nous adaptons les conditions aux limites des fonctions φ_R , φ_S et φ_f de la façon suivante :

- $\varphi_f(x, y, \perp) = \perp$
- $\varphi_*(\perp) = 0$ avec $*$ $\in \{\mathcal{R}, \mathcal{S}\}$: si le seul attaquant (respectivement supporter) est absent, le poids des attaquants (respectivement supporteurs) est nul

- $\varphi_*(\mathbf{x}_n, \perp) = \varphi_*(\mathbf{x}_n)$ quelle que soit la position de $\perp$: un argument absent n'a pas d'impact sur le poids des attaquants et des supporteurs.

Remarque – L'ajout de cette valeur hors-échelle n'est pas nécessaire dans le formalisme des QBAF, il n'est utile que pour la transformation en modèle structurel causal. En effet, le calcul du degré d'acceptabilité d'un argument n'inclue que les arguments du graphe, c'est-à-dire ceux qui sont présents. Avec un modèle structurel causal, on souhaite étudier des scénarios contrefactuels et donc rendre absents des arguments qui ne l'étaient pas à la base. Or, les équations structurelles, dont le rôle est de calculer le degré d'acceptabilité des arguments, sont définies en même temps que le reste du modèle structurel causal et ne peuvent pas être modifiées en fonction du contexte. Par conséquent, il est nécessaire d'introduire une valeur dont la présence n'influence pas le résultat des équations structurelles.

Ainsi, nous proposons une nouvelle transformation, $\mathcal{T}_{QBAFa2MSC}$, permettant de passer d'un QBAF acyclique à un modèle structurel causal. En particulier, nous proposons de créer deux variables pour la force intrinsèque d'un argument, une variable exogène $\tilde{w}_a$ et une endogène w_a . ayant des rôles différents : la variable exogène permet de générer des scénarios contrefactuels et la variable endogène permet d'inclure la force de l'argument dans les causes potentielles et donc dans des explications.

Nous créons également trois variables endogènes pour représenter chacune des étapes d'agrégation de la sémantique agrégative. Elles permettent, lors de la recherche de causes effectives, de causes suffisantes et surtout d'explications, d'obtenir plusieurs niveaux de détails différents. En effet, un premier niveau d'explication possible est de fournir les valeurs des poids globaux des attaquants, des supporteurs et la force intrinsèque. À un second niveau, le destinataire de l'explication peut demander une explication sur les valeurs de ces poids. Dans ce cas, l'explication fournie porte sur les degrés d'acceptabilité d'un sous-ensemble d'attaquants ou de supporteurs. À un troisième niveau, l'*explainée* peut demander des explications sur ces degrés d'acceptabilité permettant de remonter à la force intrinsèque ainsi qu'au poids global des attaquants et des supporteurs de ce nouvel argument. Cette procédure peut se poursuivre jusqu'à atteindre des arguments non attaqués et non supportés pour lesquels l'explication du degré d'acceptabilité est leur force intrinsèque.

Définition 5.5 (Passage d'un QBAF à un modèle structurel causal – $\mathcal{T}_{QBAFa2MSC}$).

Soit $A = (\mathcal{A}, \mathcal{R}, \mathcal{S}, w)$ un QBAF et Deg_A une sémantique agrégative de fonctions d'agrégation φ_R, φ_S et φ_f .

On définit le triplet $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ par :

- $\mathcal{U} = \{\tilde{w}_a \mid a \in \mathcal{A}\}$;
- $\mathcal{V} = \mathcal{V}_w \cup \mathcal{V}_R \cup \mathcal{V}_S \cup \mathcal{V}_f$ avec :
 - ◊ $\mathcal{V}_w = \{w_a \mid a \in \mathcal{A}\}$,
 - ◊ $\mathcal{V}_R = \{\pi_R^a \mid a \in \mathcal{A}\}$,
 - ◊ $\mathcal{V}_S = \{\pi_S^a \mid a \in \mathcal{A}\}$,
 - ◊ $\mathcal{V}_f = \{\pi_f^a \mid a \in \mathcal{A}\}$;
- $\mathcal{F} = \{F_{w_a} \mid w_a \in \mathcal{V}_w\} \cup \{F_{\sigma_a} \mid \sigma_a \in \mathcal{V}_\sigma\}$ avec :
 - ◊ $\forall w_a \in \mathcal{V}_w, F_{w_a} = \tilde{w}_a$,
 - ◊ $\forall \pi_R^a \in \mathcal{V}_R, F_{\pi_R^a} = \varphi_R(\{\sigma_b \mid b \in \text{Att}_a\})$,
 - ◊ $\forall \pi_S^a \in \mathcal{V}_S, F_{\pi_S^a} = \varphi_S(\{\sigma_c \mid c \in \text{Supp}_a\})$,

$$\diamond \forall \pi_f^a \in \mathcal{V}_f, F_{\pi_f^a} = \varphi_f(\pi_{\mathcal{R}}^a, \pi_{\mathcal{S}}^a, w_a).$$

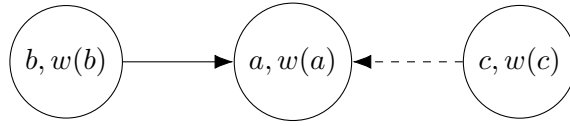
Proposition 5.6. Soit $A = (\mathcal{A}, \mathcal{R}, \mathcal{S}, w)$ un QBAF et Deg_A une sémantique agrégative de fonctions d'agrégation φ_R, φ_S et φ_f .

Soit $M = \mathcal{T}_{QBAFa2MSC}(A, \text{Deg}_A)$.

Alors M est un modèle structurel causal au sens de la définition 2.1.

Preuve. Par construction, on vérifie de manière immédiate la définition. $\square$

Exemple 10. Soit A le QBAF tel que $\mathcal{A} = \{a, b, c\}$, $\mathcal{R} = \{(b, a)\}$ et $\mathcal{S} = \{(c, a)\}$ représenté ci-dessous.



La figure 5.2 représente le graphe associé au modèle structurel causal $\mathcal{T}_{QBAFa2MSC}(A)$: à chaque argument on associe cinq variables, une exogène et quatre endogènes.

5.6.2 Que cherche-t-on réellement à expliquer ?

La seule différence avec ce que nous proposons dans le chapitre 2 est le modèle structurel causal adopté. Pour cette raison, en appliquant la définition présentée dans la section 2.2.4, il est possible de générer des explications causales sur le statut d'acceptabilité d'un argument en mettant en œuvre la même méthode que discutée dans la section 2.3.3.

Toutefois, les définitions de causes effectives, causes suffisantes et explications (voir définitions 2.3, 2.4 et 2.5) portent sur la valeur précise d'un ensemble de variables du graphe. Or, contrairement au cas binaire, expliquer pourquoi un argument a est acceptable avec un degré de 0.42 par exemple n'est selon nous pas intéressant. En effet, quel est le sens associé à une valeur de 0.42 par rapport par exemple à une valeur de 0.43 ? De plus si les trois fonctions d'agrégation sont continues, une petite variation de n'importe laquelle des variables, que ce soit une force intrinsèque ou bien un degré d'acceptabilité d'un attaquant ou d'un supporter, entraîne un degré d'acceptabilité différent et constitue donc une cause effective de $\text{Deg}_A(a) = 0.42$ (voir condition (AC2) de la définition 2.3).

Plusieurs solutions sont alors possibles :

- Rechercher des explications contrastives, c'est-à-dire pourquoi $\pi_f^a = u$ et $\pi_f^a \neq u'$? C'est ce qui est proposé par Yin *et al.* (2024a) sous le nom de contrefactuels forts en recherchant comment modifier les forces intrinsèques des arguments du graphe pour obtenir le degré d'acceptabilité désiré, où u' est fourni par l'explainée, permettant de plus personnaliser l'explication.
- Expliquer non pas une valeur mais un intervalle de valeurs c'est-à-dire pourquoi $\pi_f^a \in \mathcal{B}$ avec $\mathcal{B}$ un intervalle choisi par le destinataire de l'explication.
- Combiner les deux points précédents en expliquant pourquoi $\pi_f^a \in \mathcal{B}$ et $\pi_f^a \notin \mathcal{B}'$ avec $\mathcal{B}, \mathcal{B}'$ des intervalles choisis par l'explainée.

Dans le premier cas, une définition d'explication contrastive a été proposée par Miller (2021) pour des modèles structurels causaux. L'application de cette dernière ne demandant pas de changement particulier, nous ne la détaillons pas ici. Elle permet en plus de

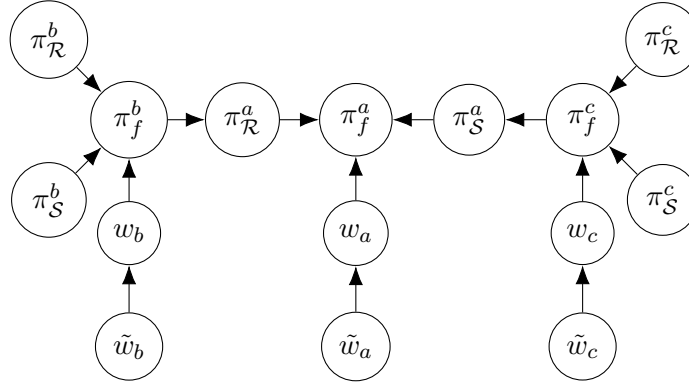


FIGURE 5.2 – Modèle structurel causal associé à l'exemple 10.

comparer, comme nous le faisons dans la section 2.3.3 pour la définition de Fan & Toni (2014), si la définition de Yin *et al.* (2024a) correspond à une explication causale contrastive. De plus, bien que l'écart entre deux valeurs dans une explication contrastive ait plus de sens pour un utilisateur qu'une valeur simple, de telles questions nous semblent peu réalistes. En effet, quelle est la différence entre savoir pourquoi π_f^a vaut 0.42 plutôt que 0.84 par rapport à savoir pourquoi π_f^a vaut 0.42 plutôt que 0.88 ?

Les deux types d'explications suivants permettent de partiellement remédier à ce problème. En effet, on peut par exemple en amont partitionner l'ensemble des valeurs d'acceptabilité en plusieurs sous-ensembles et associer une variable linguistique pour caractériser ces sous-ensembles voire demander à l'utilisateur de lui-même choisir les intervalles qui l'intéressent. De cette façon, on n'expliquerait non plus pourquoi $\pi_f^a = 0.42$ et $\pi_f^a \neq 0.84$ mais plutôt pourquoi a est moyennement acceptable plutôt que fortement acceptable. Imaginons que nous décidions de partitionner $[0, 1]$ en trois intervalles. Pour appliquer les définitions de Halpern (2016) comme détaillées dans le chapitre 2, il faut, pour chaque argument a , créer trois variables endogènes supplémentaires $\{X_a^{[0, s_1]}, X_a^{[s_1, s_2]}, X_a^{[s_2, 1]}\}$ dont les équations structurelles sont $F_{X_a^B} = \mathbb{1}_B(\pi_f^a)$. Avec cette modification, on peut expliquer pourquoi $\pi_f^a \in \mathcal{B}$ ou pourquoi $\pi_f^a \in \mathcal{B}$ et $\pi_f^a \notin \mathcal{B}'$.

Cependant, la définition de ces intervalles avec le choix des bornes en particulier pose à nouveau question. De même, si un argument a a un degré d'acceptabilité proche d'une des bornes de l'intervalle, expliquer son appartenance à ce dernier plutôt qu'à celui d'après n'aura pas le même impact qu'avec un argument situé "au milieu" de l'intervalle en question. Une solution possible pour ces deux problèmes pourrait être de considérer des variables linguistiques, par exemple peu acceptable, acceptable, fortement acceptable, etc., représentées par des sous-ensembles flous prédéfinis ou dont les définitions sont laissées au choix de l'utilisateur et ensuite utiliser la définition d'explication contrastive floue proposée par Bloch & Lesot (2022). Une première possibilité pourrait être d'avoir des calculs précis puis de fuzzifier le résultat pour avoir une présentation plus compréhensible pour l'utilisateur. Une autre possibilité, grâce au formalisme des sémantiques agrégatives que nous proposons, est que cette étape de fuzzification ait lieu dès la phase de calcul du degré d'acceptabilité d'un argument. En effet, il existe des opérateurs d'agrégation flous (Dubois & Prade, 1985) pouvant être utilisés pour les fonctions $\varphi_{\mathcal{R}}$, $\varphi_{\mathcal{S}}$ et φ_f , comme par exemple les t-normes dont nous discutons pour le postulat

d'affaiblissement. Dans ce cas, les poids globaux flous pour les attaquants et les supporters peuvent être calculés et ensuite être agrégés avec une force intrinsèque elle-même floue. Avoir une force intrinsèque floue permettrait, dans le cas où l'estimation de cette valeur est imprécise, de propager cette incertitude dans tout le processus de calcul du degré d'acceptabilité des arguments. De plus, les discussions menées sur les postulats de ces opérateurs restent valides même dans ce cas. Ces propositions constituent des pistes de recherche pour des travaux futurs.

5.7 Conclusion

Dans ce chapitre, nous avons proposé une formalisation d'un cas particulier de sémantique graduelle que nous appelons sémantique agrégative. Elle repose sur la décomposition du processus de calcul de l'acceptabilité d'un argument en trois étapes d'agrégation distinctes : calcul du poids global des attaquants, calcul du poids global des supporters et agrégation de ces deux poids avec la force intrinsèque de l'argument. Grâce à la diversité des fonctions d'agrégation qui existent dans la littérature, il est possible de construire de nouvelles sémantiques graduelles. De plus, la discussion que nous avons menée sur les postulats et axiomes désirables pour ces fonctions permet de guider le choix des différentes fonctions. Cela permet d'obtenir une méthode de calcul pour le degré d'acceptabilité d'un argument plus compréhensible pour un utilisateur. Nous avons également proposé une extension des travaux du chapitre 2, en proposant une réécriture des QBAF acycliques en modèles structurels causaux. Ainsi, en appliquant les définitions présentées dans le chapitre 2, il est possible de générer des explications causales sur l'acceptabilité des arguments du graphe. Enfin, la signification d'une valeur précise pour ce degré d'acceptabilité n'étant pas définie formellement, nous soulevons la question de ce que l'on souhaite expliquer avec ce cadre étendu et présentons différentes solutions possibles.

Une première perspective pour ce travail consiste à proposer une instanciation des sémantiques modulaires de Mossakowski & Neuhaus (2018) quand cela est possible, ainsi que des nouvelles sémantiques agrégatives afin de les comparer sur différents exemples. Une seconde perspective consiste à étudier la convergence de ces nouvelles sémantiques afin de déterminer des conditions sur les différentes fonctions d'agrégation permettant de garantir la convergence dans des cas cycliques.

6.1 Conclusions

Au cours de cette thèse, nous avons étudié la question de la génération d'explications causales pour l'argumentation abstraite afin d'enrichir le statut d'acceptabilité des arguments par des informations permettant au destinataire de comprendre pourquoi un argument est considéré acceptable. Nous avons abordé les trois cadres suivants : les systèmes d'argumentation abstraite (AAF) de Dung (1995), une extension que nous proposons qui prend en compte la temporalité, c'est-à-dire l'ordre d'énonciation des arguments, et le cas des systèmes d'argumentation bipolaire pondérée.

Pour la génération des explications, nous avons considéré un formalisme causal, considérant qu'une réponse à la question "pourquoi" correspondait à l'identification de causes : nous avons utilisé les modèles structurels causaux (Halpern, 2016) qui proposent une définition de la notion de causalité adaptée à des situations complexes, par exemple la surdétermination. Pour les exploiter dans le cadre de l'argumentation abstraite, nous avons proposé un cas particulier des modèles structurels causaux que nous appelons des graphes causaux argumentatifs (GCA). Nous avons établi une équivalence entre AAF acycliques et GCA, en construisant deux transformations. Grâce à ces transformations, nous avons montré que la définition d'explications en argumentation abstraite proposée par Fan & Toni (2014) constitue une explication causale non minimale au sens de Halpern. Cela implique une forme de redondance dans l'explication générée ce qui peut conduire à de longues explications même dans des cas simples, contredisant une des propriétés d'une bonne explication (Miller, 2019). Enfin, en transformant un AAF en GCA, nous profitons d'un cadre adapté à la génération d'explications causales, à savoir celui proposé par les modèles structurels causaux. En effet, grâce à la notion de contexte, nous pouvons adapter les explications à leur destinataire en restreignant l'ensemble des contextes considérés à ceux qui sont compatibles avec ses connaissances, permettant ainsi la génération d'explications personnalisées.

Nous nous sommes ensuite intéressés à la question de la temporalité, afin d'augmenter l'expressivité des AAF pour la représentation des dialogues, par la prise en compte de

l'ordre d'énonciation des arguments. Pour cela, nous avons proposé, en collaboration avec Camilo Sarmiento, une réécriture des AAF, acycliques dans un premier temps puis cycliques dans un second temps, dans un langage d'action incluant une définition de causalité effective. De cette façon, l'ordre d'énonciation des arguments est pris en compte dans le modèle et a une incidence sur les causes et donc les explications. Nous avons établi des propriétés formelles pour cette transformation : terminaison, correction et complétude. Nous avons également montré que l'ordre d'énonciation a un impact direct sur les chaînes causales, confirmant l'intérêt de sa prise en compte. Nous avons ensuite discuté du passage des chaînes causales, potentiellement longues et complexes, à des explications causales présentant des caractéristiques souhaitables ainsi que de deux représentations graphiques pouvant servir de support pour ces explications. Nous avons étendu ces propositions au cas des graphes cycliques, pour lesquels la terminaison, la correction et la complétude de la transformation ne sont plus garanties. Nous avons donc proposé une nouvelle transformation, inspirée par les fonctions d'étiquetage complet des AAF, dont nous avons établi la terminaison et la correction. De plus, la temporalité joue un rôle supplémentaire avec cette nouvelle réécriture : à un ordre d'énonciation donné correspond un unique état final, contrairement à l'argumentation abstraite selon laquelle il peut y avoir plusieurs extensions complètes. Nous pensons qu'il s'agit d'une propriété souhaitable dans le cas de la génération d'explications causales. En effet, la temporalité constitue une source d'informations riches qui permet de faire une différence entre des extensions qui ne sont équivalentes que si on omet l'ordre d'énonciation des arguments. De plus, le fait qu'il puisse y avoir plusieurs résultats possibles, potentiellement contradictoires, nécessite une connaissance du fonctionnement de l'argumentation abstraite et n'est donc pas adapté pour un public non expert. Enfin, nous avons étudié la propriété de complétude et proposé deux définitions : une première, qui peut être dite "forte", fait disparaître l'avantage de prendre en compte la temporalité, et la seconde, plus "faible", selon laquelle pour chaque extension complète, il existe un ordre d'énonciation permettant d'obtenir cette extension. Nous nous sommes focalisés sur cette deuxième forme, en accord avec la nécessité de modéliser l'apport de la temporalité. Pour la vérifier, nous avons proposé une modification de la transformation selon une stratégie que nous avons appelée "dernier énoncé, dernier mis à jour". Celle-ci repose sur l'hypothèse que l'on examine d'abord les conséquences d'un nouvel argument avant de considérer les conséquences des autres arguments sur celui-lui.

Enfin, nous avons étendu nos travaux à un cadre offrant une expressivité plus grande pour modéliser des situations complexes : nous nous sommes intéressés aux systèmes d'argumentation bipolaire pondérée (QBAF) qui ajoutent une relation binaire, le support, et offrent la possibilité d'ajouter un poids aux arguments. Une première différence majeure avec les cas précédents réside dans le recours à une fonction, appelée sémantique graduelle, qui associe à chaque argument un degré d'acceptabilité en fonction des autres arguments du graphe. On passe ainsi d'une acceptabilité binaire à une valeur numérique. Cette fonction vient remplacer l'ensemble de règles qui permet de déterminer si un argument est acceptable ou non dans le cas des AAF classiques. Nous avons proposé une famille de sémantiques graduelles, que nous avons appelées sémantiques graduelles agrégatives. Elles sont formées de trois fonctions d'agrégation distinctes dont nous discutons les propriétés selon le contexte. Les bénéfices apportés par cette nouvelle famille sont multiples. Tout d'abord, contrairement aux sémantiques graduelles de la littérature, les sémantiques agrégatives permettent de conserver la bipolarité grâce à

un calcul du poids des attaquants indépendant de celui des supporteurs. De plus, ce découpage en trois fonctions d'agrégation ouvre la voie vers de nouvelles sémantiques graduelles grâce à la grande diversité de telles fonctions étudiées dans la littérature (voir par exemple Grabisch *et al.* (2009)). Enfin, décomposer le processus en étapes intermédiaires permet d'augmenter la granularité des explications causales générées, c'est-à-dire d'ajouter différents niveaux pour les explications : un premier niveau d'explications utilise la force globale des attaquants et celle des supporteurs d'un argument ; un deuxième remonte au degré d'acceptabilité des attaquants et des supporteurs ; et un dernier parcourt la chaîne d'arguments pour remonter jusqu'à la force intrinsèque des arguments non attaqués et non supportés. Nous sommes ensuite revenus sur l'objectif initial, à savoir étendre les propositions précédentes sur la génération d'explications causales de l'acceptabilité des arguments aux QBAF. Nous avons par conséquent proposé une modification de la transformation en modèle structurel causal pour l'adapter aux QBAF. Finalement, le statut d'acceptabilité d'un argument étant défini dans ce cadre par un score, nous discutons également de la signification que peut prendre une explication. En particulier, nous soutenons le fait qu'expliquer une valeur précise d'acceptabilité d'un argument n'a pas vraiment de sens et qu'il est plus intéressant d'obtenir des explications sur l'appartenance à un intervalle de valeurs par exemple.

En résumé, nos contributions répondent aux questions posées dans l'introduction, à savoir la génération d'explications causales en argumentation abstraite pour différents cadres : celui des AAF classiques (**Q1**), celui des AAF enrichis avec l'ordre d'énonciation des arguments (**Q2**) et enfin celui des AAF étendus avec une relation binaire supplémentaire et un poids sur les arguments (**Q3**).

6.2 Perspectives

Les travaux que nous avons présentés dans cette thèse ouvrent des perspectives multiples que nous organisons autour de trois axes :

- **Enrichissement des transformations** : Dans la littérature, il existe de nombreuses extensions des AAF qui permettent d'enrichir le modèle de l'interaction. Étendre nos propositions à ces extensions permet d'obtenir un modèle plus expressif et ainsi améliorer la qualité des explications.
- **Évaluation, comparaison et enrichissement des explications causales pour l'argumentation abstraite** : Afin de pouvoir extraire des explications causales adaptées à l'utilisateur à partir des chaînes causales, une étape de formalisation des différentes propriétés souhaitables pour ces explications, proposées par exemple par Miller (2019), est nécessaire. Utiliser ces critères pourrait également permettre de comparer différentes explications, mesurer leur similarité ou bien évaluer leur complémentarité.
- **Mise en œuvre expérimentale** : Afin d'évaluer et comparer les différentes méthodes liées à la génération d'explications en argumentation (nos propositions et celles de la littérature), des expériences avec des utilisateurs sont nécessaires. Cela implique de se poser non seulement la question des évaluations et des tests pour les utilisateurs mais également de leur implémentation.

6.2.1 Enrichissement des transformations

Ajout de la temporalité dans les modèles structurels causaux – Lorsque nous nous sommes intéressés à la question de la génération d'explications causales incluant l'ordre d'énonciation des arguments, nous avons fait le choix d'utiliser un formalisme différent des modèles structurels causaux que nous avons exploités pour les AAF classiques. En effet, ces modèles n'intègrent pas naturellement la notion de temporalité. Elle peut être injectée artificiellement en modifiant les équations structurelles. C'est ce qui a été fait dans l'exemple du lancer de pierre sur la bouteille (voir exemple 4). En effet, nous avons fait l'hypothèse que si Suzy lance, sa pierre arrive toujours avant celle de Billy et c'est pourquoi Billy ne touche la bouteille que s'il lance une pierre et que Suzy n'en lance pas. Toutefois, que se passe-t-il si Billy lance d'abord, puis que Suzy lance?

Une première perspective est d'étendre les modèles structurels causaux à un cadre temporel pour permettre de continuer d'utiliser ce formalisme pour l'argumentation abstraite. Deux solutions naïves sont envisageables : considérer une suite de modèles structurels causaux, similaires à des états avec un langage d'action, ou ne plus considérer des variables endogènes et exogènes statiques mais dépendantes du temps. En particulier, l'utilisation d'une logique temporelle dans les équations structurelles peut permettre de modéliser les relations temporelles entre les variables. Par exemple, à l'aide de l'opérateur *Until*, on peut écrire que Billy touche la bouteille à la date t si jusqu'à la date t , ni lui ni Suzy ne l'ont touchée et qu'il a lancé une pierre. Ces deux modifications impliquent de plus d'adapter les définitions d'une cause effective, d'une cause suffisante et d'une explication pour ces nouveaux cadres et de vérifier leurs propriétés théoriques et pratiques.

Extensions aux systèmes d'argumentation incomplets – Nous avons considéré dans le chapitre 5 le cadre étendu des systèmes d'argumentation bipolaire pondérée, qui permettent d'augmenter l'expressivité du modèle pour représenter des situations complexes. Il existe de nombreuses autres extensions d'un AAF qu'il serait également intéressant d'étudier. En particulier, les systèmes d'argumentation incomplets (Coste-Marquis *et al.*, 2007; Mailly, 2022) permettent de modéliser une incertitude sur la présence ou non d'un argument ou d'une attaque, c'est-à-dire des nœuds et des arcs dans le graphe. La méthode de raisonnement la plus courante pour exploiter de tels systèmes est basée sur une notion de complétion. Elle consiste à créer un AAF pour chaque résolution possible des incertitudes puis à considérer qu'une propriété est vraie si elle est vraie dans au moins une complétion ou dans toutes les complétions. Ces deux formes d'incertitudes peuvent intervenir dans différents contextes. Dans un contexte multi-agents par exemple, la connaissance de chaque agent peut être modélisée par un AAF. Pour obtenir un point de vue global, il faut fusionner les différents AAF ce qui peut conduire à une incertitude sur la présence ou non d'une attaque entre des arguments (Coste-Marquis *et al.*, 2007). Une incertitude sur la présence d'un argument peut être observée dans un débat dans lequel on modélise son interlocuteur. En effet, on sait par exemple de façon certaine que l'argument a attaque l'argument b . Pour autant, on ne sait pas si l'interlocuteur connaît l'argument a ou non.

Inclure ces incertitudes dans un des cadres causaux que nous utilisons permettrait de gagner en expressivité dans la modélisation. La notion de complétion d'un AAF semble pouvoir être rapprochée de l'idée de la notion de contextes des modèles structurels cau-

saux, chacun représentant une résolution possible des incertitudes. Il faudrait ensuite adapter la définition d'une cause effective à un ensemble de contextes, correspondant aux complétions, et ainsi permettre d'appliquer les définitions qui s'ensuivent pour la génération d'explications causales.

6.2.2 Évaluation, comparaison et enrichissement des explications causales pour l'argumentation abstraite

Le langage d'action que nous avons introduit dans le chapitre 3 permet, à partir d'un AAF et d'un ordre d'énonciation des arguments, d'extraire la chaîne causale du statut d'acceptabilité de tout argument du graphe. Toutefois, une telle chaîne ne constitue pas nécessairement une explication. Nous nous demandons donc quels sous-ensembles de cette chaîne causale pourrait constituer une explication de bonne qualité.

La question de la qualité des explications est largement débattue, à la croisée de différentes disciplines incluant par exemple les sciences cognitives, les sciences de l'éducation et l'informatique dans le domaine de l'IA explicable (voir par exemple Miller 2019; Byrne 2019) Même si elle n'a pas encore abouti à une définition consensuelle, des propriétés à respecter ont été proposées. Elles peuvent ainsi constituer des critères numériques d'évaluation. Une fois ces différents critères évalués, l'objectif est de les combiner afin d'obtenir une relation d'ordre partiel sur les différentes explications possibles. Cela soulève donc deux questions différentes : comment évaluer ces critères et comment combiner ces évaluations.

Évaluation des critères de qualité d'une explication causale – Afin d'évaluer les différents critères de qualité, une étape de formalisation de ces derniers dans le formalisme utilisé pour générer les explications est nécessaire. Pour cela, nous pouvons nous inspirer des traductions qui ont été faites par exemple dans le cadre des explications contrefactuelles (Guidotti, 2024). Certains critères, comme la longueur d'une explication ou bien sa proximité temporelle avec la chose à expliquer, étant des critères subjectifs dépendant du destinataire de l'explication, une autre piste intéressante peut être de se baser sur des travaux en sciences cognitives qui modélisent les utilisateurs dans des modèles informatiques (Liefoghe & van Maanen, 2023).

Par exemple, pour qu'une explication soit bonne, les causes de la chose à expliquer doivent être temporellement proches de cette dernière. Dans un langage d'action, ce critère peut être mesuré comme le nombre d'états qui séparent la cause de sa conséquence. Cette distance peut représenter le score d'évaluation. Néanmoins, il existe de nombreux travaux sur les capacités limitées de la mémoire à court terme (Miller, 1956; Cowan, 2001), impliquant par exemple dans notre cas que des arguments trop éloignés pourraient être oubliés. Des modèles computationnels de cette notion d'oubli ont été proposés (Pavlik Jr & Anderson, 2005) et pourraient donc servir d'inspiration pour évaluer le critère de proximité temporelle.

Structuration de l'espace des explications – Afin de pouvoir comparer les différentes explications candidates et ainsi choisir les plus adaptées, une étape de structuration de l'espace des explications est nécessaire. Cela peut se faire en définissant différentes mesures de comparaison comme la similarité de deux explications afin d'évaluer leur com-

plémentarité, leur redondance ou leur contradiction. Une autre possibilité pourrait être d'agréger les évaluations de chacun des critères choisis à l'aide d'une fonction d'agrégation. Cela implique de discuter des différentes propriétés voulues pour un tel opérateur. Enfin, on peut également voir la question comme un problème de rangement en décision multicritère dans lequel l'objectif est de construire une relation d'ordre partiel sur l'espace des explications possibles en fonction d'un ensemble de critères imposés.

Explications sur la structure d'un AAF – Dans la thèse, nous nous sommes intéressés à la question de la génération d'explications causales pour le statut d'acceptabilité d'un argument, étant donné un AAF. Toutefois, l'explication peut ne pas être adaptée si l'*explainée* ne comprend pas la structure du graphe et notamment les relations binaires entre les arguments. En effet, si une explication fournie pour la non acceptabilité de l'argument a est l'énonciation de l'argument b car b attaque a , alors si cette attaque n'est pas comprise par l'*explainée*, l'explication ne sera pas compréhensible non plus. Une perspective est donc de pouvoir autoriser à questionner la structure du graphe d'argumentation en plus de l'acceptabilité des arguments. Dans le cas où le graphe d'argumentation est construit automatiquement, une direction possible pour répondre à cette question est à chercher au moment de la création de l'AAF grâce à des méthodes de fouille d'arguments (Lippi & Torroni, 2016) avec l'application de techniques développées dans le domaine de l'XAI.

6.2.3 Mise en œuvre expérimentale

Dans la section précédente, une des perspectives de recherche consiste à structurer l'espace des explications de façon à pouvoir définir un ensemble de meilleures explications. Pour confirmer ces définitions théoriques, le meilleur évaluateur de la qualité d'une explication est son destinataire lui-même, c'est-à-dire un humain. De fait, une des perspectives principales de cette thèse est la mise en place d'un cas d'application pratique, à partir d'exemples de débats réels complexes, afin d'évaluer les différentes méthodes proposées ainsi que celles de la littérature. Toutefois, mettre en place une telle expérimentation soulève de nombreuses questions sous-jacentes à plusieurs niveaux, parmi lesquels l'implémentation des différentes méthodes et l'évaluation de ces différentes méthodes.

Implémentation – L'implémentation des méthodes formelles que nous avons proposées pose de nombreuses questions. Tout d'abord, pour un modèle structurel causal, la vérification automatique qu'une cause candidate est bien une cause effective est un problème de complexité D_1^P (Halpern, 2015). La recherche de causes effectives, et par conséquent de causes suffisantes et d'explications, est donc au moins aussi difficile. Dans le cas des graphes acycliques, des travaux ont été proposés pour rechercher des causes effectives en formulant le problème comme un programme linéaire en nombres entiers (Ibrahim & Pretschner, 2020). Cela n'est cependant pas suffisant pour trouver des explications puisqu'il faut ensuite rechercher des causes suffisantes.

Concernant le langage d'action, nous avons proposé une implémentation en ASP, basée sur une implémentation déjà existante, pour construire les traces causales. Toutefois, nous n'avons utilisé ce programme que pour les exemples discutés dans cette thèse dont

la taille est réduite. De fait, avant de le mettre en œuvre pour un problème réel, une étude théorique de la complexité ainsi qu'une étude empirique du temps de calcul sont requises.

Évaluations – La mise en place d'un cas pratique permet de comparer et d'évaluer les différentes notions utilisées dans cette thèse. Tout d'abord, nous avons proposé deux formalismes causaux différents pour permettre de générer des explications causales. Le premier, les modèles structurels causaux, repose sur une définition de la causalité effective basée sur une approche contrefactuelle. Le deuxième, le test NESS, fait partie des approches par régularité. Il s'agit de deux approches très différentes par nature et qui peuvent donc fournir des causes effectives différentes étant donné un même problème. Un premier résultat intéressant serait donc comparer les causes effectives identifiées par chacun de ces formalismes.

D'autre part, dans le chapitre 3, nous avons proposé deux représentations graphiques du dialogue pouvant servir de support pour l'explication. En effet, des recherches récentes ont souligné l'importance de la façon de présenter des explications pour les rendre intelligibles (Liao *et al.*, 2020), donnant naissance au domaine du design d'interfaces explicables (eXplainable User Interface, XUI). Les représentations que nous avons proposées ont été développées indépendamment du domaine de l'XUI. Une perspective est de les évaluer à l'aide des critères de qualité formalisés dans ce domaine et de concevoir de nouvelles représentations graphiques du dialogue.

Un autre point nécessitant une évaluation est la stratégie que nous proposons dans le chapitre 4 afin de garantir une forme de complétude. Cette stratégie, que nous appelons "dernier énoncé, dernier mis à jour", est fondée sur l'hypothèse forte suivante : quand un argument est énoncé, ses effets sont évalués en premier avant de considérer les effets des autres arguments sur celui-ci. Grâce à cette stratégie, nous obtenons une transformation qui garantit l'existence d'un ordre d'énonciation conduisant à chaque extension complète. Une perspective consiste à vérifier cette hypothèse en proposant des dialogues avec des ordres d'énonciation différents afin de comparer les extensions auxquelles elles conduisent en appliquant ce principe aux arguments jugés acceptables par les participants. On pourra pour cela s'inspirer des travaux réalisés en argumentation abstraite pour évaluer expérimentalement les sémantiques argumentatives (Cramer & Guillaume, 2018) ou bien différents principes argumentatifs, comme par exemple certains axiomes (Rahwan *et al.*, 2010; Vesic *et al.*, 2022).

Enfin, nous proposons d'évaluer les explications générées par différentes méthodes de la littérature ainsi que celles que nous proposons. Un premier point intéressant est que les méthodes de la littérature, ainsi que celle basée sur les modèles structurels causaux ne tiennent pas compte de la temporalité. Par conséquent, un premier résultat attendu est la comparaison de nos contributions avec celles de la littérature. En particulier, en générant différentes explications causales en faisant varier les différents critères d'une bonne explication (Miller, 2019) qu'elles vérifient, nous souhaitons évaluer l'impact et l'évaluation de ces critères. Cela permettrait également de valider et de guider les travaux sur la structuration de l'espace des explications et en particulier les relations d'ordre partielles proposées.

Pour cela, une étape essentielle sera la conception du questionnaire d'évaluation pour chacun des points que nous souhaitons comparer entre les différentes approches. En

particulier, la question de l'évaluation de l'efficacité d'une explication est une question encore ouverte. Récemment, un consensus selon lequel il faut évaluer la compréhension objective et la satisfaction subjective des utilisateurs semble avoir été trouvé (Chromik *et al.*, 2021).

L'ensemble de ces études viendront ainsi compléter les différents résultats théoriques que nous avons présentés dans cette thèse afin d'aboutir à des applications concrètes de l'ensemble de nos propositions.

Références

- ALI S., ABUHMED T., EL-SAPPAGH S., MUHAMMAD K., ALONSO-MORAL J. M., CONFALONIERI R., GUIDOTTI R., DEL SER J., DÍAZ-RODRÍGUEZ N. & HERRERA F. (2023). Explainable artificial intelligence (XAI) : What we know and what is left to attain trustworthy artificial intelligence. *Information fusion*, **99**, 101805.
- AMGOUD L. & BEN-NAIM J. (2018). Weighted bipolar argumentation graphs : Axioms and semantics. In *27th Int. Joint Conf. on Artificial Intelligence-IJCAI*, p. 5194–5198.
- AMGOUD L. & CAYROL C. (2002). Inferring from inconsistency in preference-based argumentation frameworks. *Journal of Automated Reasoning*, **29**, 125–169.
- AMGOUD L., DIMOPOULOS Y. & MORAITIS P. (2007). A unified and general framework for argumentation-based negotiation. In *6th Int. joint Conf. on Autonomous Agents and MultiAgent Systems-AAMAS*, p. 1–8.
- AMGOUD L., DIMOPOULOS Y. & MORAITIS P. (2008). Making decisions through preference-based argumentation. In *Int. Conf. on Principles of Knowledge Representation and Reasoning-KR*, volume 8, p. 963–970.
- ANDREAS H. & GUENTHER M. (2021). Regularity and Inferential Theories of Causation. In *The Stanford Encyclopedia of Philosophy*. Stanford University.
- ANTAKI C. & LEUDAR I. (1992). Explaining in conversation : Towards an argument model. *European Journal of Social Psychology*, **22**(2), 181–194.
- ARISAKA R. & SATOH K. (2018). Abstract argumentation/persuasion/dynamics. In *PRIMA 2018 : 21st Int. Conf. on Principles and Practice of Multi-Agent Systems*, p. 331–343 : Springer.
- ARISTOTE (350 AEC). *Metaphysics*. The Internet Classics Archive.
- AUDEMARD G., BELLART S., BOUNIA L., KORICHE F., LAGNIEZ J.-M. & MARQUIS P. (2022). On the explanatory power of boolean decision trees. *Data & Knowledge Engineering*, **142**, 102088.
- BARONI P., CAMINADA M. & GIACOMIN M. (2011). An introduction to argumentation semantics. *The Knowledge Engineering Review*, **26**(4), 365–410.
- P. BARONI, D. GABBAY, M. GIACOMIN & L. VAN DER TORRE, Coordinateurs (2018). *Handbook of Formal Argumentation*. College Publications.
- BARONI P., RAGO A. & TONI F. (2019). From fine-grained properties to broad principles for gradual argumentation : A principled spectrum. *Int. Journal of Approximate Reasoning, IJAR*, **105**, 252–286.

- BARRINGER H., GABBAY D. & WOODS J. (2005). Temporal dynamics of support and attack networks : From argumentation to zoology. *Mechanizing Mathematical Reasoning : Essays in Honor of Jörg H. Siekmann on the Occasion of His 60th Birthday*, p. 59–98.
- BATUSOV V. & SOUTCHANSKI M. (2018). Situation calculus semantics for actual causality. In *Proceedings of the AAAI Conf. on Artificial Intelligence*, volume 32.
- BECKERS S. (2021). The counterfactual NESS definition of causation. In *Proceedings of the AAAI Conf. on artificial intelligence*, volume 35, p. 6210–6217.
- BENCH-CAPON T. (2002). Value based argumentation frameworks. *arXiv preprint cs/0207059*.
- BENFERHAT S., DUBOIS D., KACI S. & PRADE H. (2002). Bipolar representation and fusion of preferences on the possibilistic logic framework. In *Int. Conf. on Principles of Knowledge Representation and Reasoning-KR*, volume 2, p. 421–432.
- BENGEL L., BLÜMEL L., RIENSTRA T. & THIMM M. (2022). Argumentation-based causal and counterfactual reasoning. In *1st Int. Workshop on Argumentation for eXplainable AI*.
- BEX F., PETERS J. & TESTERINK B. (2016). AI for online criminal complaints : From natural dialogues to structured scenarios. In *Artificial intelligence for justice workshop@ECAI 2016*, p. 22.
- BISTARELLI S. & SANTINI F. (2021). Weighted argumentation. *Journal of Applied Logics*, **2631**(6), 1589.
- BLACK E., MAUDET N. & PARSONS S. (2021). Argumentation-based dialogue. In *Handbook of Formal Argumentation, Volume 2*. College Publications.
- BLOCH I. (1996). Information Combination Operators for Data Fusion : A Comparative Review with Classification. *IEEE Transactions on Systems, Man, and Cybernetics*, **26**(1), 52–67.
- BLOCH I. & LESOT M.-J. (2022). Towards a formulation of fuzzy contrastive explanations. In *2022 IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE)*, p. 1–8 : IEEE.
- BOBILLO F. & STRACCIA U. (2013). Aggregation operators for fuzzy ontologies. *Applied Soft Computing*, **13**(9), 3816–3830.
- BOCHMAN A. (2005). Propositional argumentation and causal reasoning. In *Int. Joint Conf. on Artificial Intelligence'05*.
- BOCHMAN A. (2021). *A logical theory of causality*. MIT Press.
- BONZON E., DELOBELLE J., KONIECZNY S. & MAUDET N. (2016). A comparative study of ranking-based semantics for abstract argumentation. In *Proceedings of the AAAI Conf. on Artificial Intelligence*, volume 30.
- BONZON E., MAUDET N. & MORETTI S. (2014). Coalitional games for abstract argumentation. In *Computational Models of Argument*, p. 161–172. IOS Press.
- BORG A. & BEX F. (2021). A basic framework for explanations in argumentation. *IEEE Intelligent Systems*, **36**(2), 25–35.
- BORG A. & BEX F. (2022). Contrastive explanations for argumentation-based conclusions. In *Int. Conf. on Autonomous Agents and Multiagent Systems-AAMAS22*, p. 1551–1553.
- BORG A. & BEX F. (2024). Minimality, necessity and sufficiency for argumentation and explanation. *Int. Journal of Approximate Reasoning - IJAR*, **168**, 109–143.
- BYRNE R. M. (2019). Counterfactuals in explainable artificial intelligence (xai) : Evidence from human reasoning. In *IJCAI*, p. 6276–6282 : California, CA.
- CACIOPPO J. T., GARDNER W. L. & BERNTSON G. G. (1997). Beyond bipolar conceptualizations and measures : The case of attitudes and evaluative space. *Personality and Social Psychology Review*, **1**(1), 3–25.
- CAMINADA M. (2006). On the issue of reinstatement in argumentation. In *European Conf. on Logics in Artificial Intelligence - JELIA06*, p. 111–123 : Springer.

- CAMINADA M. (2017). Argumentation semantics as formal discussion. *Journal of Applied Logics*, **4**(8), 2457–2492.
- CAMINADA M. & PIGOZZI G. (2011). On judgment aggregation in abstract argumentation. *Autonomous Agents and Multi-Agent Systems*, **22**, 64–102.
- CAYROL C. & LAGASQUIE-SCHIEX M.-C. (2005a). Gradual valuation for bipolar argumentation frameworks. In *European Conf. on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU)*, p. 366–377 : Springer.
- CAYROL C. & LAGASQUIE-SCHIEX M.-C. (2005b). On the acceptability of arguments in bipolar argumentation frameworks. In *European Conf. on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU)*, p. 378–389 : Springer.
- CHOQUET G. (1954). Theory of capacities. In *Annales de l'institut Fourier*, volume 5, p. 131–295.
- CHROMIK M., EIBAND M., BUCHNER F., KRÜGER A. & BUTZ A. (2021). I think I get your point, AI! the illusion of explanatory depth in explainable AI. In *Int. Conf. on Intelligent User Interfaces - IUI*, p. 307–317.
- COBO M. L., MARTÍNEZ D. C. & SIMARI G. R. (2010). On admissibility in timed abstract argumentation frameworks. In *European Conf. on Artificial Intelligence - ECAI*, p. 1007–1008 : IOS Press.
- COCARASCU O., ČYRAS K., RAGO A. & TONI F. (2018). Explaining with argumentation frameworks mined from data. In *1st Int. Workshop on Dialogue, Explanation and Argumentation in Human-Agent Interaction (DEXAHAI)@HAI2018*.
- COSTE-MARQUIS S., DEVRED C., KONIECZNY S., LAGASQUIE-SCHIEX M.-C. & MARQUIS P. (2007). On the merging of dung's argumentation systems. *Artificial Intelligence*, **171**(10-15), 730–753.
- COWAN N. (2001). The magical number 4 in short-term memory : A reconsideration of mental storage capacity. *Behavioral and brain sciences*, **24**(1), 87–114.
- CRAMER M. & GUILLAUME M. (2018). Empirical cognitive study on abstract argumentation semantics. In *Computational Models of Argument*, p. 413–424. IOS Press.
- ČYRAS K., BIRCH D., GUO Y., TONI F., DULAY R., TURVEY S., GREENBERG D. & HAPUARACHCHI T. (2019). Explanations by arbitrated argumentative dispute. *Expert Systems with Applications*, **127**, 141–156.
- ČYRAS K., KAMPIK T. & WENG Q. (2022). Dispute trees as explanations in quantitative (bipolar) argumentation. In *1st Int. Workshop on Argumentation for eXplainable AI - ArgXAI@COMMA2022*, volume 3209 : CEUR-WS.
- DA COSTA PEREIRA C., TETTAMANZI A. G. & VILLATA S. (2011). Changing one's mind : Erase or rewind? possibilistic belief revision with fuzzy argumentation based on trust. In *Int. Joint Conf. on Artificial Intelligence - IJCAI*.
- DE TARLE L. D. (2023). A gradual semantic to model opinion using bipolar argumentation graphs. p. 20–27.
- DELOBELLE J. & VILLATA S. (2019). Interpretability of gradual semantics in abstract argumentation. In *15th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty - ECSQARU*, p. 27–38 : Springer.
- DOUTRE S., MAFFRE F. & MCBURNEY P. (2017). A dynamic logic framework for abstract argumentation : adding and removing arguments. In *30th Int. Conf. on Industrial Engineering and Other Applications of Applied Intelligent Systems Advances in Artificial Intelligence : From Theory to Practice IEA/AIE 2017* : Springer.
- DOUTRE S. & MAILLY J.-G. (2018). Constraints and changes : A survey of abstract argumentation dynamics. *Argument & Computation*, **9**(3), 223–248.

- DUBOIS D. & PRADE H. (1985). A review of fuzzy set aggregation connectives. *Information Sciences*, **36**(1-2), 85–121.
- DUNG P. M. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial intelligence*, **77**(2), 321–357.
- DUNG P. M., MANCARELLA P. & TONI F. (2007). Computing ideal sceptical argumentation. *Artificial Intelligence*, **171**(10-15), 642–674.
- DUNNE P. E., HUNTER A., MCBURNEY P., PARSONS S. & WOOLDRIDGE M. (2011). Weighted argument systems : Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, **175**(2), 457–486.
- DUPIN DE SAINT-CYR F., BISQUERT P., CAYROL C. & LAGASQUIE-SCHIEUX M.-C. (2016). Argumentation update in YALLA (yet another logic language for argumentation). *Int. Journal of Approximate Reasoning - IJAR*, **75**, 57–92.
- FAN X. & TONI F. (2014). On computing explanations in abstract argumentation. In *European Conf. on Artificial Intelligence - ECAI*, volume 263, p. 1005–1006.
- FAN X. & TONI F. (2015a). On computing explanations in argumentation. In *Proceedings of the AAAI Conf. on Artificial Intelligence*, volume 29.
- FAN X. & TONI F. (2015b). On explanations for non-acceptable arguments. In *3rd Int. Workshop on Theory and Applications of Formal Argumentation*, p. 112–127 : Springer.
- FERREIRA J., DE SOUSA RIBEIRO M., GONÇALVES R. & LEITE J. (2022). Looking inside the black-box : Logic-based explanations for neural networks. In *Int. Conf. on Principles of Knowledge Representation and Reasoning - KR*, volume 19, p. 432–442.
- GAO Y., TONI F., WANG H. & XU F. (2016). Argumentation-based multi-agent decision making with privacy preserved. In *Int. Conf. on Autonomous Agents and Multiagent Systems*, p. 1153–1161.
- GELFOND M. & LIFSCHITZ V. (1998). Action languages. *Electronic Transaction on Artificial Intelligence*, **2**, 193–210.
- GRABISCH M., MARICHAL J.-L., MESIAR R. & PAP E. (2009). *Aggregation functions*, volume 127. Cambridge University Press.
- GUIDOTTI R. (2024). Counterfactual explanations and how to find them : literature review and benchmarking. *Data Mining and Knowledge Discovery*, **38**(5), 2770–2824.
- HALPERN J. Y. (2015). A modification of the Halpern-Pearl definition of causality. In *24th Int. Conf. on Artificial Intelligence - IJCAI*, p. 3022–3033 : AAAI Press.
- HALPERN J. Y. (2016). *Actual Causality*. MIT Press.
- HALPERN J. Y. & PEARL J. (2005). Causes and explanations : A structural-model approach. part I : Causes. *The British J. Philosophy of Science*.
- HEMPEL C. G. & OPPENHEIM P. (1948). Studies in the logic of explanation. *Philosophy of science*, **15**(2), 135–175.
- HOLLAND J. (1986). *Induction : Processes of inference, learning, and discovery*, volume 292. MIT Press.
- IBRAHIM A. & PRETSCHNER A. (2020). From checking to inference : Actual causality computations as optimization problems. In *Int. Symposium on Automated Technology for Verification and Analysis*, p. 343–359 : Springer.
- JANSSEN J., DE COCK M. & VERMEIR D. (2008). Fuzzy argumentation frameworks. In *Information processing and management of uncertainty in knowledge-based systems - IPMU*, p. 513–520.
- KAMPIK T., ČYRAS K. & ALARCÓN J. R. (2024). Change in quantitative bipolar argumentation : Sufficient, necessary, and counterfactual explanations - *ijar*. *Int. Journal of Approximate Reasoning*, **164**.

- KEIL F. C. (2006). Explanation and understanding. *Annual Review of Psychology*, **57**(1), 227–254.
- KRISHNA S., HAN T., GU A., WU S., JABBARI S. & LAKKARAJU H. (2024). The disagreement problem in explainable machine learning : A practitioner’s perspective. *Transactions on Machine Learning Research*.
- LEWIS D. (2004). Causation as influence. In *Causation and Counterfactuals*. The MIT Press.
- LI H., OREN N. & NORMAN T. J. (2011). Probabilistic argumentation frameworks. In *Int. workshop on theory and applications of formal argumentation*, p. 1–16 : Springer.
- LIAO B. & VAN DER TORRE L. (2020). Explanation semantics for abstract argumentation. In *Computational Models of Argument*, p. 271–282. IOS Press.
- LIAO Q. V., GRUEN D. & MILLER S. (2020). Questioning the ai : informing design practices for explainable AI user experiences. In *CHI Conf. on human factors in computing systems*, p. 1–15.
- LIEFOOGHE B. & VAN MAANEN L. (2023). Three levels at which the user’s cognition can be represented in artificial intelligence. *Frontiers in Artificial Intelligence*, **5**, 1092053.
- LIPPI M. & TORRONI P. (2016). Argumentation mining : State of the art and emerging trends. *ACM Transactions on Internet Technology*, **16**(2), 1–25.
- MACKIE J. L. (1980). *The cement of the universe : A study of causation*. Clarendon Press.
- MADUMAL P., MILLER T., SONENBERG L. & VETERE F. (2019). A grounded interaction protocol for explainable artificial intelligence. In *18th Int. Conf. on Autonomous Agents and MultiAgent Systems - AAMAS* : Int. Foundation for Autonomous Agents and Multiagent Systems.
- MAILLY J.-G. (2022). Yes, no, maybe, I don’t know : Complexity and application of abstract argumentation with incomplete knowledge. *Argument & Computation*, **13**(3), 291–324.
- MENGER K. (1942). Statistical metrics. *National Academy of Sciences of the United States of America*, **28**(12), 535–537.
- MERCIER H. & SPERBER D. (2011). Why do humans reason ? arguments for an argumentative theory. *Behavioral and brain sciences*, **34**(2), 57–74.
- MILLER G. A. (1956). The magical number seven, plus or minus two : Some limits on our capacity for processing information. *Psychological review*, **63**(2), 81.
- MILLER N. & CAMPBELL D. T. (1959). Recency and primacy in persuasion as a function of the timing of speeches and measurements. *The Journal of Abnormal and Social Psychology*, **59**(1).
- MILLER T. (2019). Explanation in artificial intelligence : Insights from the social sciences. *Artificial Intelligence*, **267**, 1–38.
- MILLER T. (2021). Contrastive explanation : A structural-model approach. *The Knowledge Engineering Review*, **36**, e14.
- MINH D., WANG H. X., LI Y. F. & NGUYEN T. N. (2022). Explainable artificial intelligence : a comprehensive review. *Artificial Intelligence Review*, p. 1–66.
- MODGIL S. & CAMINADA M. (2009). Proof theories and algorithms for abstract argumentation frameworks. *Argumentation in artificial intelligence*, p. 105–129.
- MOSSAKOWSKI T. & NEUHAUS F. (2018). Modular semantics and characteristics for bipolar weighted argumentation graphs. *arXiv preprint arXiv :1807.06685*.
- MUNRO Y., BLOCH I., CHETOUANI M., LESOT M.-J. & PELACHAUD C. (2022a). Argumentation and causal models in human-machine interaction : A round trip. In *8th Int. Workshop on Artificial Intelligence and Cognition*.
- MUNRO Y., BLOCH I., CHETOUANI M., LESOT M.-J. & PELACHAUD C. (2022b). De l’équivalence entre les modèles structurels causaux et les systèmes abstraits d’argumentation. In *Rencontres des Jeunes Chercheurs en Intelligence Artificielle*, p. 123–130 : Association française pour l’intelligence artificielle. RJCIA 2022.

- MUNRO Y., BLOCH I., CHETOUANI M., PELACHAUD C. & LESOT M.-J. (2024a). Sémantique agrégative graduelle pour les systèmes d'argumentation bipolaires pondérés. In *JIAF-JFPDA 2024 (Plate-Forme Intelligence Artificielle)*.
- MUNRO Y., SARMIENTO C., BLOCH I., BOURGNE G. & LESOT M.-J. (2023a). Dynamic argumentation and action languages : Towards explanations. In *The 4th Workshop on Explainable Logic-Based Knowledge Representation (XLoKR 2023@KR23)*.
- MUNRO Y., SARMIENTO C., BLOCH I., BOURGNE G. & LESOT M.-J. (2023b). Temporalité et causalité en argumentation abstraite. In *JIAF-JFPDA 2023 (Journées d'Intelligence Artificielle Fondamentale 2023)*.
- MUNRO Y., SARMIENTO C., BLOCH I., BOURGNE G. & LESOT M.-J. (2025). Integrating temporality and causality into acyclic argumentation frameworks using a transition system. *AI Communications*, in press.
- MUNRO Y., SARMIENTO C., BLOCH I., BOURGNE G., PELACHAUD C. & LESOT M.-J. (2024b). An action language-based formalisation of an abstract argumentation framework. In *The 25th Int. Conf. On Principles and Practice of Multi-Agent Systems (PRIMA 2024)*.
- PAVLIK JR P. I. & ANDERSON J. R. (2005). Practice and forgetting effects on vocabulary memory : An activation-based model of the spacing effect. *Cognitive science*, **29**(4), 559–586.
- PEARL J. (2000). Models, reasoning and inference. *Cambridge, UK : Cambridge University Press*, **19**(2), 3.
- POTYKA N. (2018). Continuous dynamical systems for weighted bipolar argumentation. In *16th Int. Conf. on Principles of Knowledge Representation and Reasoning - KR*.
- POTYKA N. (2019). Extending modular semantics for bipolar weighted argumentation. In *KI 2019 : Advances in Artificial Intelligence : 42nd German Conf. on AI*, p. 273–276 : Springer.
- POTYKA N. (2021). Interpreting neural networks as quantitative argumentation frameworks. In *AAAI Conf. on Artificial Intelligence*, volume 35, p. 6463–6470.
- RAGO A., BARONI P. & TONI F. (2022). Explaining Causal Models with Argumentation : the Case of Bi-variate Reinforcement. In *Int. Conf. on Principles of Knowledge Representation and Reasoning - KR*, p. 505–509.
- RAGO A., COCARASCU O. & TONI F. (2018). Argumentation-based recommendations : Fantastic explanations and how to find them. In *Int. Joint Conf. on Artificial Intelligence - IJCAI*, volume 18, p. 1949–1955.
- RAGO A. & TONI F. (2017). Quantitative argumentation debates with votes for opinion polling. In *20th Int. Conf. of Principles and Practice of Multi-Agent Systems - PRIMA2017*, p. 369–385 : Springer.
- RAGO A., TONI F., AURISICCHIO M. & BARONI P. (2016). Discontinuity-free decision support with quantitative argumentation debates. In *15th Int. Conf. on the Principles of Knowledge Representation and Reasoning - KR*.
- RAHWAN I., MADAKKATEL M. I., BONNEFON J.-F., AWAN R. N. & ABDALLAH S. (2010). Behavioral experiments for assessing the abstract argumentation semantics of reinstatement. *Cognitive Science*, **34**(8), 1483–1502.
- SARIBATUR Z. G., WALLNER J. P. & WOLTRAN S. (2020). Explaining non-acceptability in abstract argumentation. In *European Conference on Artificial Intelligence - ECAI*, p. 881–888. IOS Press.
- SARMIENTO C., BOURGNE G., INOUE K., CAVALLI D. & GANASCIA J.-G. (2023). Action languages based actual causality for computational ethics : a sound and complete implementation in ASP. *arXiv, CoRR*.
- SARMIENTO C., BOURGNE G., INOUE K. & GANASCIA J.-G. (2022). Action languages based actual causality in decision making contexts. In *Int. Conf. on Principles and Practice of Multi-Agent Systems - PRIMA*, volume LNCS 13753 : Springer.

- SARTOR G., ARASZKIEWICZ M., ATKINSON K., BEX F., VAN ENGERS T., FRANCESCONI E., PRAKKEN H., SILENO G., SCHILDER F., WYNER A. *et al.* (2022). Thirty years of artificial intelligence and law : the second decade. *Artificial Intelligence and Law*, **30**(4), 521–557.
- SASSOON I., KÖKCIYAN N., MODGIL S. & PARSONS S. (2021). Argumentation schemes for clinical decision support. *Argument & Computation*, **12**(3), 329–355.
- SRINIVASAN R. & CHANDER A. (2021). Explanation perspectives from the cognitive sciences—a survey. In *Int. Joint Conf on Artificial Intelligence - IJCAI*, p. 4812–4818.
- TORRA V. & NARUKAWA Y. (2007). *Modeling decisions : information fusion and aggregation operators*. Springer Science & Business Media.
- VASSILIADES A., BASSILIADES N. & PATKOS T. (2021). Argumentation and explainable artificial intelligence : a survey. *The Knowledge Engineering Review*, **36**, e5.
- ČYRAS K., RAGO A., ALBINI E., BARONI P. & TONI F. (2021). Argumentative XAI : A survey. In *Int. Joint Conf. on Artificial Intelligence - IJCAI'21*, p. 4392–4399.
- VESIC S., YUN B. & TEOVANOVIC P. (2022). Graphical representation enhances human compliance with principles for graded argumentation semantics. In *Int. Conf. on Autonomous Agents and MultiAgent Systems - AAMAS*.
- WRIGHT R. W. (1985). Causation in tort law. *California Law Review*, **73**(6), 1735–1828.
- WRIGHT R. W. (2011). The NESS account of natural causation : A response to criticisms. In *Perspectives on Causation*. Social Science Research Network.
- YAGER R. R. & RYBALOV A. (1998). Full reinforcement operators in aggregation techniques. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **28**(6), 757–769.
- YIN X., POTYKA N. & TONI F. (2023). Argument attribution explanations in quantitative bipolar argumentation frameworks.
- YIN X., POTYKA N. & TONI F. (2024a). Ce-qarg : Counterfactual explanations for quantitative bipolar argumentation frameworks (technical report). *arXiv preprint arXiv :2407.08497*.
- YIN X., POTYKA N. & TONI F. (2024b). Explaining arguments' strength : Unveiling the role of attacks and supports. In *Int. Joint Conf. on Artificial Intelligence - IJCAI*.
- ZENG Z., FAN X., MIAO C., LEUNG C., CHIN J. J. & ONG Y. S. (2018). Context-based and explainable decision making with argumentation. In *17th International Conference on Autonomous Agents and MultiAgent Systems*.

A Preuves du chapitre 2

Dans cette section, nous détaillons les preuves des différentes propositions du chapitre 2.

A.1 Proposition 2.2

Proposition (Explication compacte et modèle structurel). *Soit $AF = (\mathcal{A}, \mathcal{R})$ un système d'argumentation abstraite, dont le graphe est acyclique.*

Soit $a^ \in \mathcal{A}$ tel qu'il existe un ensemble admissible dont il est le sujet. Soit S une explication compacte de a^* .*

Soit $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ le graphe causal argumentatif issu de la transformation $\mathcal{T}_{AFa2GCA}$ (définition 2.7).

On pose :

- $\varphi = (X_{a^*} = 1)$,
- $\mathbf{X}_{\text{arg}} = S \setminus \{a^*\}$ et $\mathbf{X} = \{X_a \mid a \in \mathbf{X}_{\text{arg}}\}$,
- $\mathcal{K}$ l'ensemble des contextes considérés comme possibles par l'explaine. On fait l'hypothèse que le contexte par défaut $\mathbf{u}^*$ appartient à $\mathcal{K}$.

*Alors $\mathbf{X} = 1$ est une explication, au sens causal, non minimale, de φ relative à $\mathcal{K}$, c'est-à-dire $\mathbf{X} = 1$ vérifie **EX1** et **EX3** dans $\mathcal{K}$.*

Preuve de la proposition 2.2. Montrons que $\mathbf{X} = 1$ vérifie **EX1** et **EX3** dans $\mathcal{K}$.

(EX3) Montrons d'abord que $\mathbf{u}^*$ est inclus dans $\mathcal{K}_{(\mathbf{X}=1) \wedge \varphi}$.

Par hypothèse, $\mathbf{u}^*$ est inclus dans $\mathcal{K}$.

(i) Montrons par l'absurde que $\mathbf{u}^*$ est inclus dans $\mathcal{K}_{(\mathbf{X}=1)}$.

Supposons que $(M, \mathbf{u}^*) \models \neg(\mathbf{X} = 1)$. Alors, $\exists X_a \in \mathbf{X}$ tel que $X_a = 0$. Or $\text{Att}_a \neq \emptyset$, donc comme $F_{X_a} = (\bigwedge_{b \in \text{Att}_a} \neg X_b)$, $\exists b \in \text{Att}_{X_a}$ tel que $X_b = 1$.

Or S est admissible donc $\exists c \in \text{Att}_b \cap S$.

Si $\text{Att}_c = \emptyset$ alors $X_c = 1$ par définition de $\mathbf{u}^*$. C'est impossible car $X_b = 1$. On arrive donc à une contradiction.

Sinon, comme $X_b = 1$ alors $X_c = 0$. Or $Att_c \neq \emptyset$, donc comme $F_{X_c} = (\bigwedge_{d \in Att_c} \neg X_d)$,

$\exists d \in Att_{X_c}$ tel que $X_d = 1$. Comme S est admissible, $\exists e \in Att_d \cap S$.

Si $Att_e = \emptyset$ alors $X_e = 1$ par définition de u^* . C'est impossible car $X_d = 1$. On arrive donc à une contradiction.

Sinon, on peut encore une fois répéter ce raisonnement jusqu'à ce que $Att_e = \emptyset$ car le graphe est fini et acyclique.

Donc u^* est inclus dans $\mathcal{K}_{X=1}$.

(ii) Montrons maintenant que u^* est inclus dans $\mathcal{K}_\varphi$.

Comme S est admissible de sujet a^* et que le graphe est acyclique, alors $\forall b \in Att_{a^*}$, $\exists c \in X \cap Att_b$. Or $X = 1$, donc $X_c = 1$ et de fait $X_b = 0$. Ainsi, $\forall b \in Att_{a^*}$, $X_b = 0$ et donc $X_{a^*} = \bigwedge_{b \in Att_{a^*}} \neg 0 = 1$.

Donc u^* est inclus dans $\mathcal{K}_\varphi$.

Ainsi, u^* est inclus dans $\mathcal{K}_{(X=1) \wedge \varphi}$, et donc cet ensemble est non vide et **EX3** est satisfaite.

(**EX1**) Montrons maintenant que $X = 1$ est une cause suffisante dans $\mathcal{K}$, c'est-à-dire qu'il vérifie **SC1**, **SC2** et **SC3**, pour tout $u \in \mathcal{K}_{(X=1) \wedge \varphi}$.

Soit $u \in \mathcal{K}_{(X=1) \wedge \varphi}$:

SC4 – Il s'agit d'une condition de minimalité qui porte sur la cause suffisante mais qui dans le cas des explications est équivalente à **EX2** (Halpern, 2016). Pour cette raison, on ne démontre pas que $X = 1$ vérifie **SC4**.

SC1 – Cette condition est vérifiée par définition de u .

SC3 – Montrons par l'absurde que **SC3** est vérifiée.

Soit u' un contexte tel que $(M, u') \models [X = 1] \neg \varphi$.

Comme on a $\neg \varphi$ (c'est-à-dire $X_{a^*} = 0$), d'après $\mathcal{F}$ pour les arguments attaqués (a^* est un sujet de S et donc $Att_{a^*} \neq \emptyset$) $\exists X_b \in \mathcal{V}$, tel que $b \in Att_{a^*}$ et $X_b = 1$.

Or S est admissible donc $\exists c \in S$ tel que $\mathcal{R}(c, b)$. De plus, le graphe est acyclique donc $c \neq a^*$ et donc $c \in X$. Comme $X = 1$, on a en particulier $X_c = 1$ et donc $X_b = 0$ d'après F_{X_b} . On arrive à une contradiction.

SC2 – Enfin montrons que **SC2** est bien vérifiée en deux temps :

(i) On construit d'abord une cause effective de φ dans u :

Soit $b \in Att_{a^*}$, posons $Z_b = \bigcup_{c \in Att_b} \{X_c \mid (M, u) \models (X_c = 1)\}$. Comme $u \in \mathcal{K}_{(X=1) \wedge \varphi}$,

alors $X_{a^*} = 1$ et donc $X_b = 0$.

Or S est admissible donc en particulier, $\forall \alpha \in S, \forall \beta \in Att_\alpha, \exists \gamma \in S \cap Att_\beta$.

Comme $b \in Att_{a^*}$ et $a^* \in S$, $Att_b \neq \emptyset$. De plus, $X_b = 0$ donc $\exists X_c \in Att_b$ tel que $X_c = 1$. Ainsi, $X_c \in Z_b$ donc Z_b est non vide.

Posons ensuite $Z = \bigcup_{b \in Att_a} Z_b$. Z n'est pas le candidat pour être une cause effective. Toutefois, montrons qu'il vérifie **AC1** et **AC2** :

- **AC1** est vérifiée par construction de Z_b .
- Par construction de Z , si on impose $Z = 0$, alors on a $\forall b \in Att_{a^*}, \forall c \in Att_b, X_c = 0$.
Or $F_{X_b} = \bigwedge_{c \in Att_b} \neg X_c = \bigwedge_{b \in Att_a} \neg 0 = 1$.
On a donc bien $(M, u) \models [Z = 0] \neg \varphi$ et donc **AC2** est vérifiée avec $W = \emptyset$.

Notons Z^m un sous-ensemble minimal de Z tel que $(Z^m = 1)$ vérifie **AC1** et **AC2**.

Il est bien défini et est non vide car $(Z = 1)$ vérifie **AC1** et **AC2**.

De plus, Z^m vérifie **AC3** par définition.

On a donc construit Z^m tel que $(Z^m = 1)$ vérifie **AC1**, **AC2** et **AC3** c'est-à-dire que $(Z^m = 1)$ est une cause effective de φ .

(ii) On montre ensuite que cet ensemble contient bien au moins un élément de X . Pour cela, prouvons maintenant que l'on peut construire une cause effective $(Z' = z')$ de φ telle que $Z' \cap X \neq \emptyset$.

Si $Z^m \cap X \neq \emptyset$ alors $Z' = Z^m$ convient.

Sinon, c'est-à-dire si $Z^m \cap X = \emptyset$, soit $b \in \text{Att}_a^*$:

- $\exists X_c \in Z^m$ tel que $c \in \text{Att}_b$, $(M, u) \models (X_c = 1)$ et $X_c \notin X$.
- Comme S est admissible et le graphe est acyclique, $\exists X_{c'} \in X$ tel que $c' \in \text{Att}_b$.
De plus, $Z^m \cap X = \emptyset$, donc $X_{c'} \notin Z^m$.

Enfin, comme $u \in \mathcal{K}$ on a $(M, u) \models (X_{c'} = 1)$.

Posons $Z^{m'} = (Z^m \setminus \{X_c\}) \cup \{X_{c'}\}$. $Z^{m'}$ vérifie aussi **AC1** et **AC2**. Comme Z^m est minimal par construction, alors si $Z^{m'}$ ne l'est pas, $\exists Z' \subseteq Z^{m'}$ tel que $Z' \not\subseteq Z^m$. Or $Z^{m'} \setminus Z^m = \{X_{c'}\}$ donc $X_{c'} \in Z'$ et on a donc $Z' \cap X \neq \emptyset$. On a donc construit un ensemble Z' vérifiant **AC1** et **AC2**, minimal pour l'inclusion (**AC3**) et tel que $Z' \cap X \neq \emptyset$ on a donc vérifié **SC2**.

On a montré que $X = 1$ vérifie bien **SC1**, **SC2** et **SC3**, pour tout $u \in \mathcal{K}_{(X=1) \wedge \varphi}$ c'est-à-dire que $X = 1$ vérifie bien **EX1**.

Comme on a montré que $X = 1$ vérifie **EX1** et **EX3**, alors $X = 1$ est une explication (au sens causal) non minimale de φ . $\square$

A.2 Proposition 2.4

Proposition (Équivalence entre AAF et GCA). *Soit $AF = (\mathcal{A}, \mathcal{R})$ un système d'argumentation abstrait acyclique. Alors :*

$$AF = \mathcal{T}_{GCA2AFa}(\mathcal{T}_{AFa2GCA}(AF)).$$

Preuve de la Proposition 2.4. Soit $AF = (\mathcal{A}, \mathcal{R})$ un système abstrait d'argumentation, $M = (\mathcal{U}, \mathcal{V}, \mathcal{F})$ le graphe causal argumentatif associé à AF et $AF' = (\mathcal{A}', \mathcal{R}')$ le système d'argumentation associé à M .

Montrons que $AF = AF'$ c'est-à-dire $\mathcal{A} = \mathcal{A}'$ et $\mathcal{R} = \mathcal{R}'$.

- On a par construction $\mathcal{A} = \mathcal{A}'$.
- Montrons que $\mathcal{R} = \mathcal{R}'$ par double inclusion.

1. Soit $(a, b) \in \mathcal{A}^2$ tel que $\mathcal{R}(b, a)$.

On a par définition $X_a = \tilde{X}_a \wedge \neg X_b \wedge (\bigwedge_{c \in \text{Att}_a \wedge c \neq b} \neg X_c)$.

En particulier, si $X_b = 1$ alors on a :

$$X_a = \tilde{X}_a \wedge 0 \wedge (\bigwedge_{c \in \text{Att}_a \wedge c \neq b} \neg X_c) = 0.$$

Ainsi, pour tout contexte u , on a $(M, u) \models [X_b = 1, Y = 0](X_a = 0)$ avec $Y = \mathcal{V} \setminus \{X_a, X_b\}$, donc $(b, a) \in \mathcal{R}'$ et $\mathcal{R} \subseteq \mathcal{R}'$.

2. Soit $(a', b') \in \mathcal{A}'^2$ tel que $b' \in \text{Att}_{a'}^{\mathcal{R}'}$.

Soit $\mathbf{Y} = \mathcal{V} \setminus \{X_{a'}, X_{b'}\}$. On a par définition

$$(M, \mathbf{u}) \models [X_{b'} = 1, \mathbf{Y} = \mathbf{0}](X_{a'} = 0)$$

Or $\mathcal{A} = \mathcal{A}'$, donc $\forall \alpha \in \mathcal{A}, X_\alpha = X_{\alpha'}$. En particulier $(M, \mathbf{u}) \models [X_{b'} = 1, \mathbf{Y} = \mathbf{0}](X_a = 0)$ et donc $\text{Att}_a^{\mathcal{R}} \neq \emptyset$.

De plus, $F_{X_a} = \bigwedge_{z \in \text{Att}_a^{\mathcal{R}}} \neg X_z$. Si $b' \notin \text{Att}_a^{\mathcal{R}}$ alors avec $\mathbf{Y} = \mathbf{0}$ on a $F_{X_{a'}} = \bigwedge_{\beta \in \text{Att}_a^{\mathcal{R}}} \neg 0 = 1$,

ce qui contredit l'hypothèse.

On a donc $b' \in \text{Att}_a^{\mathcal{R}}$ et $\mathcal{R}' \subseteq \mathcal{R}$.

On a donc montré par double inclusion que $\mathcal{R} = \mathcal{R}'$. □

B Preuves du chapitre 3

Dans cette section, nous détaillons les preuves du chapitre 3. Celles-ci sont données en anglais, par conséquent nous énonçons les lemmes, propriétés et théorèmes en anglais, en conservant leur nom en français.

B.1 Proposition 3.3

Proposition (Unicité des traces). *Given an argumentative setting $\chi = (\varsigma, \kappa)$, the traces τ_χ^e and τ_χ^s are unique.*

Preuve de la Proposition 3.3. Let us prove by contradiction the unicity of valid executions given χ . Let $\chi = (\varsigma, \kappa)$ be the argumentative setting and ϵ, ϵ' two valid executions given χ . By way of a reductio ad absurdum, we suppose that $\epsilon \neq \epsilon'$.

According to Definition 3.5, $S(t+1)$ is derived from $S(t)$ and the events in $E(t)$, and similarly for S' . Hence, given that κ is common to ϵ and ϵ' , $E(-1) = E'(-1)$ and $S(0) = S'(0)$, the first discrepancy between ϵ , and ϵ' is not to be found in a set of states, but in a set of events, which is not empty as the executions are valid. Let t_0 be the minimal date at which a difference between ϵ and ϵ' is observed. We have $E(t_0) \neq E'(t_0)$, $\forall t < t_0$, $E(t) = E'(t)$, and $\forall t \leq t_0$, $S(t) = S'(t)$. Thus, $\forall e \in \mathbb{E}$, $S(t_0) \models \text{pre}(e) \Leftrightarrow S'(t_0) \models \text{pre}(e)$. Without loss of generality, let us consider an event e_0 such that $e_0 \notin E(t_0)$ and $e_0 \in E'(t_0)$. Two cases can occur, $e_0 \in \mathbb{U}$ or $e_0 \in \mathbb{A}$.

- (i) Let us first show by contradiction that $e_0 \notin \mathbb{U}$. Let us suppose $e_0 \in \mathbb{U}$. As $e_0 \in E'(t_0)$ and $S(t_0) = S'(t_0)$, $S(t_0) \models \text{tri}(e_0)$. Then from 2.c in Definition 3.12, $e_0 \notin E(t_0)$ implies $\exists e \in E(t_0)$ such that $e \succ_{\mathbb{E}} e_0$. Then from 2.b applied to $E'(t_0)$, we get $e \notin E'(t_0)$. Now, either $e \in \mathbb{A}$ or $e \in \mathbb{U}$. If $e \in \mathbb{A}$, as $e \in E(t_0)$, condition 2.d would imply that $S'(t_0) \not\models \text{tri}(e_0)$ which contradicts our assumption. In the second case, if $e \in \mathbb{U}$, then $e \in E(t_0)$ and $e \notin E'(t_0)$ because of the same reasons behind $e_0 \notin E(t_0)$ and $e_0 \in E'(t_0)$. If we apply the same reasoning to e , we get $\exists e' \in \mathbb{U}$ such that $e' \in E(t_0)$, $e' \notin E'(t_0)$, and $e' \succ_{\mathbb{E}} e$. This reasoning can be repeated again on e' and so on. As $\mathbb{U}$ is finite, either the chain will be broken, or an event will be used a second time. The first case means $\exists \tilde{e} \in \mathbb{U}$ such that $\tilde{e} \notin E(t_0)$ and $\tilde{e} \in E'(t_0)$ is false, which makes all the chain false. In the second case, by transitivity we

get $\tilde{e} \succ_{\mathbb{E}} \tilde{e}$. This leads to a contradiction as $\succ_{\mathbb{E}}$ is a strict partial order. Thus, $e_0 \in \mathbb{U}$ is not possible.

- (ii) From (i), $e_0 \in \mathbb{A}$. Then, using condition 1. in Definition 3.13, $e_0 \in E'(t_0)$ implies that $e_0 \in \{a, \exists o \in \mathbb{N}, (a, o) \in \varsigma\}$. ς being the same for ϵ and ϵ' , the rank $o_0 \in \mathbb{N}$ associated to e_0 is the same. Hence, given that $\forall t < t_0, \epsilon(t) = \epsilon'(t)$ and $e_0 \in E'(t_0)$, $e_0 \notin E'(t)$ implies $e_0 \notin E(t)$. The only possibility left is the procrastination of actions. In the case where $\nexists (e, o_0) \in \varsigma$ such that $e \in E(t_0)$, $E(t_0) = \emptyset$ which is in contradiction with condition 2.e in Definition 3.12. Otherwise, we have a contradiction with condition 3. in Definition 3.13. Thus, $e_0 \in \mathbb{A}$ is not possible.

Since the assumption of the existence of t_0 leads to a contradiction for all above cases, we have no alternative but to reject the existence of a minimal date t_0 at which a difference between ϵ and ϵ' is observed, and thus to reject that $\epsilon \neq \epsilon'$.

The unicity of traces is accordingly established. $\square$

B.2 Lemme 3.1

Lemme (Caractérisation d'un état argumentatif). *Let $S^x(t)$ be a state. The two following propositions are equivalent :*

- $\forall e \in \mathbb{U}, S^x(t) \not\models \text{tri}(e)$
- $S^x(t)$ is an argumentative state as defined in Definition 3.9.

Preuve du Lemme 3.1. In our context, $\mathbb{U} = \{\text{makesAcc}, \text{makesUnacc}\}$. As a consequence, we prove that $S^x(t) \not\models \text{tri}(\text{makesAcc}_x)$ is equivalent to (ii) of Definition 3.9 and $S^x(t) \not\models \text{tri}(\text{makesUnacc}_{y,x})$ is equivalent to (i) of Definition 3.9 : for any x, y

$$\begin{aligned} \bullet \neg \text{tri}(\text{makesAcc}_x) &\equiv \neg(p_x \wedge \neg a_x \wedge (\bigwedge_y \neg a_y \vee \neg cA_{y,x})) \\ &\equiv \neg(p_x \wedge (\bigwedge_y \neg a_y \vee \neg cA_{y,x})) \vee a_x \\ &\equiv (p_x \wedge (\bigwedge_y \neg a_y \vee \neg cA_{y,x}) \Rightarrow a_x) \end{aligned}$$

We obtain the desired equivalence with (ii) in Definition 3.9.

$$\begin{aligned} \bullet \neg \text{tri}(\text{makesUnacc}_{y,x}) &\equiv \neg(a_x \wedge a_y \wedge cA_{y,x}) \\ &\equiv \neg(a_x \wedge p_y \wedge a_y \wedge cA_{y,x}) \text{ as } a_y \text{ implies } p_y \\ &\equiv \neg(a_x \wedge p_y \wedge cA_{y,x}) \vee \neg a_y \\ &\equiv (a_x \wedge p_y \wedge cA_{y,x} \Rightarrow \neg a_y) \end{aligned}$$

We obtain the desired equivalence with (i) in Definition 3.9. $\square$

B.3 Proposition 3.4

Proposition (Terminaison). *Given an argumentative setting χ , a time t , an argumentative state $S^x(t)$ and an argument $x \in \mathcal{A}$, if $\text{enunciate}_x \in E^x(t)$, then $\exists t' \in \mathbb{T}, t < t'$ such that $S^x(t')$ is an argumentative state.*

Preuve de la Proposition 3.4. Given an argumentative state $S^x(t)$ and $x \in \mathcal{A}$ such that $\text{enunciate}_x \in E^x(t)$, let us prove that $\exists t' \in \mathbb{T}, t < t'$ such that $\forall e \in \mathbb{U}, S^x(t') \not\models \text{tri}(e)$, which leads to the desired result using Lemma 3.1.

As $\mathbb{U}$ and $\{(x, y) \in \mathcal{A}^2 \mid S^x(t) \models cA_{y,x}\}$ are finite sets, there is a finite number of possible triggerings for $E^x(t)$. Moreover, as there is a finite number of arguments, there is a finite number of paths in the associated graph. The graph being acyclic, each path length is finite. Therefore, there is a finite maximum number of sets of events (M) and therefore $\exists t' \leq (t + M + 1)$ such that $\forall e \in \mathbb{U}, S^x(t') \not\models tri(e)$. $\square$

B.4 Lemme 3.2

Lemme (Caractérisation d'un argument non acceptable). *Given an argumentative state $S^x(t)$, for any $x \in \mathcal{A}$, $S^x(t) \models p_x \wedge \neg a_x \Leftrightarrow S^x(t) \models \exists y, p_x \wedge a_y \wedge cA_{y,x}$.*

Preuve du Lemme 3.2.

- $\Rightarrow$ For any x such that $S^x(t) \models p_x \wedge \neg a_x$, from (ii) of Definition 3.9 we get that $S^x(t) \models \neg p_x \vee (\bigvee_y a_y \wedge cA_{y,x})$. Therefore, as $S^x(t) \models p_x$, $S^x(t) \models \exists y, p_x \wedge a_y \wedge cA_{y,x}$.
- $\Leftarrow$ Let us prove it by contraposition. Let x_0 be such that $S^x(t) \models \neg p_{x_0} \vee a_{x_0}$. If $S^x(t) \models \neg p_{x_0}$, then x_0 is such that $S^x(t) \models \forall y, \neg p_{x_0} \vee \neg a_y \vee \neg cA_{y,x_0}$, which concludes the proof. Otherwise, $S^x(t) \models p_{x_0} \wedge a_{x_0}$. If $S^x(t) \models p_{x_0} \wedge (\exists y, a_y \wedge cA_{y,x_0})$ then $S^x(t) \models tri(makesUnacc_{y,x_0})$, which is not possible as $S^x(t)$ is argumentative. Thus both cases lead to a contradiction.

$\square$

B.5 Proposition 3.5

Proposition (Correction de l'acceptabilité d'un argument). *Given an argumentative state $S^x(t)$ and its associated graph $AF = (\mathcal{A}, \mathcal{R})$ according to Definition 3.14, for any $x \in \mathcal{A}$, x acceptable by $\mathcal{A} \iff S^x(t) \models a_x$.*

Preuve de la Proposition 3.5.

- $\Rightarrow$ Let $x_0 \in \mathcal{A}$ such that x_0 is acceptable by $\mathcal{A}$, let us prove that $S^x(t) \models a_{x_0}$.

Let us suppose that $S^x(t) \models \neg a_{x_0}$.

Moreover, by construction of AF, $S^x(t) \models p_{x_0}$. $S^x(t)$ is an argumentative state so according to Lemma 3.2, $S^x(t) \models p_{x_0} \wedge \neg a_{x_0} \Leftrightarrow S^x(t) \models \exists y, p_{x_0} \wedge a_y \wedge cA_{y,x_0}$. Now, applying (i) of Definition 3.9 to a_y , we get that $\forall z, S^x(t) \models a_y \wedge p_z \wedge cA_{z,y} \Rightarrow S^x(t) \models \neg a_z$.

As there is a finite number of arguments, it is possible to repeat the process we applied for x_0 on z until one of the two scenarios :

- $S^x(t) \models \nexists y, p_z \wedge a_y \wedge cA_{y,z}$.

This leads to trigger the exogenous event $makesAcc_z$ which is not possible as $S^x(t)$ is an argumentative state according to Lemma 3.1.

- $\forall z, S^x(t) \models a_y \wedge p_z \wedge cA_{z,y} \Rightarrow S^x(t) \models \neg a_z$ where $p_z \wedge cA_{z,y}$ is false.

Then in AF , $Att_y = \emptyset$. Therefore, y is acceptable which contradicts that x_0 is acceptable.

So, $S^x(t) \models a_{x_0}$.

$\Leftarrow$ Let $x_0 \in \mathcal{A}$ such that $S^x(t) \models a_{x_0}$. Let us prove that x_0 is acceptable by $\mathcal{A}$.

As $S^x(t) \models a_{x_0}$ and $S^x(t)$ is argumentative, according to Lemma 3.2, we have that $\forall y, S^x(t) \models a_{x_0} \wedge p_y \wedge cA_{y,x_0} \Rightarrow S^x(t) \models \neg a_y$. Then, for any y satisfying the premise, by definition of AF , $(x_0, y) \in \mathcal{A}^2$ and $(y, x_0) \in \mathcal{R}$.

If such a y is acceptable by $\mathcal{A}$, then according to $[\Rightarrow]$, $S^x(t) \models a_y$. In that case, $S^x(t) \models tri(makesUnacc_{y,x_0})$ which contradicts the fact that $S^x(t)$ is an argumentative state.

So, as $\forall y \in \text{Att}_x$, y is not acceptable by $\mathcal{A}$, x_0 is acceptable by $\mathcal{A}$.

□

B.6 Théorème 3.1

Théorème (Correction et complétude). *Given a dialogue Δ and a set of attacks $\mathcal{R}$, given an argumentative setting χ , the argumentative graph AF' associated with the final argumentative state $S^x(t)$, and $AF = (\mathcal{A}, \mathcal{R})$ obtained from $(\Delta, \mathcal{R})$, it holds that $AF' = AF$.*

Preuve de la Proposition 3.1. As AF' is associated with a final argumentative state, we get that $\forall x \in \mathcal{A}, \exists t' \in \mathbb{T}$ such that $t' < t$ and $\text{enunciate}_x \in E^x(t')$.

Now, $\text{eff}(\text{enunciate}_x) \equiv p_x \wedge a_x$. So, $\mathcal{A}' = \mathcal{A}$.

Moreover, by construction of $cA_{y,x}$ and $\mathcal{R}'$, $\mathcal{R} = \mathcal{R}'$. So $AF = AF'$.

Finally, from Proposition 3.5 as $S^x(t)$ is argumentative, $\forall x \in \mathcal{A} = \mathcal{A}', S^x(t) \models a_x \Leftrightarrow x$ is acceptable by $\mathcal{A}$. □

C Preuves du chapitre 4

Dans cette section, nous détaillons les preuves du chapitre 4. Comme dans la section précédente, celles-ci sont données en anglais, par conséquent nous énonçons les lemmes, propriétés et théorèmes en anglais, en conservant leur nom en français.

C.1 Proposition 4.1

Proposition (Cohérence des fluents i et o). $\forall t, \forall x$, if $S^x(t)$ is a state then $S^x(t) \not\models i_x \wedge o_x$.

Preuve de la Proposition 4.1. Let us prove by contradiction that $\forall t \in \mathbb{T}, \forall x, S^x(t) \not\models i_x \wedge o_x$.

We define $t_0 = \min_{t \geq 1} \{t \mid \exists x, (S^x(t-1) \not\models i_x \wedge o_x) \text{ and } (S^x(t) \models i_x \wedge o_x)\}$.

By construction, $\exists x_0, S^x(t_0 - 1) \models \neg i_{x_0} \vee \neg o_{x_0}$ and $S^x(t_0) \models i_{x_0} \wedge o_{x_0}$.

i) If $S^x(t_0 - 1) \models \neg o_{x_0}$ then to reach $S^x(t_0)$, $\exists y_0, S^x(t_0 - 1) \models tri(\Delta_{y_0, x_0}^2)$ which is equivalent to say that $S^x(t_0 - 1) \models p_{x_0} \wedge \neg o_{x_0} \wedge cA_{y_0, x_0} \wedge p_{y_0} \wedge i_{y_0}$.

Moreover, by definition of t_0 , $S^x(t_0 - 1) \models i_{y_0} \wedge \neg o_{y_0}$.

- If $S^x(t_0 - 1) \models i_{x_0} \wedge \neg o_{x_0}$, then $S^x(t_0 - 1) \models tri(\Delta_{y_0, x_0}^1)$ which is impossible as $\Delta_{y_0, x_0}^2 \in E^x(t_0)$ and $\forall x, y, \Delta_{y, x}^1 \succ \Delta_{y, x}^2$.
- If $S^x(t_0 - 1) \models \neg i_{x_0} \wedge \neg o_{x_0}$, then in addition to Δ_{y_0, x_0}^2 , $I_{x_0}^2$ also has to be triggered so that x_0 becomes *in*. Thus, $I_{x_0}^2 \in E^x(t_0 - 1)$.

It means that $S^x(t_0 - 1) \models p_{x_0} \wedge \neg i_{x_0} \wedge \left(\bigwedge_y (p_y \wedge cA_{y, x_0} \wedge o_y) \vee \neg cA_{y, x_0} \right)$.

It especially applies to y_0 and we have that $S^x(t_0 - 1) \models i_{y_0} \wedge o_{y_0}$. This is a contradiction by construction of t_0 .

By contradiction, $S^x(t_0 - 1) \not\models \neg o_{x_0}$

ii) If $S^x(t_0 - 1) \models \neg i_{x_0} \wedge o_{x_0}$ then $I_{x_0}^2 \in E^x(t_0 - 1)$. Therefore, by definition of t_0 , $\forall y, S^x(t_0 - 1) \models (p_y \wedge cA_{y,x_0} \wedge o_y \wedge \neg i_y) \vee \neg cA_{y,x_0}$.

Then, $S^x(t_0 - 1) \models p_{x_0} \wedge o_{x_0} \wedge \left(\bigwedge_y (p_y \wedge cA_{y,x_0} \wedge \neg i_y) \vee \neg cA_{y,x_0} \right)$, i.e. $S^x(t_0 - 1) \models I_{x_0}^1$.

But, $\forall x, I_x^1 \succ I_x^2$. So this is a contradiction.

We proved by contradiction that such a t_0 does not exist. Moreover, for $t = 0$ there is no present argument so they are all unacceptable.

Therefore, $\forall t \in \mathbb{T}, \forall x, S^x(t) \not\models i_x \wedge o_x$. □

C.2 Lemme 4.1

Lemme (Caractérisation d'un état Σ_c -argumentatif). *Let $S^x(t)$ be a state. The two following propositions are equivalent :*

- (i) $S^x(t)$ is a Σ_c -argumentative state as in Definition 4.1.
- (ii) $\forall e \in \mathbb{U}, S^x(t) \not\models tri(e)$.

Preuve du Lemme 4.1.

(i) $\Rightarrow$ (ii) Let $S^x(t)$ be a Σ_c -argumentative state and $e \in \mathbb{U}$ be an exogenous event. Let us prove by exhaustion on e that $S^x(t) \not\models tri(e)$.

- Let x, y such that $e = \Delta_{y,x}^1$ and let us suppose that $S^x(t) \models tri(e)$. It is equivalent to $S^x(t) \models p_x \wedge i_x \wedge p_y \wedge cA_{y,x} \wedge \neg o_y$.

According to Proposition 4.1, $S^x(t) \models p_x \wedge i_x \wedge \neg o_x \wedge p_y \wedge cA_{y,x} \wedge \neg o_y$.

Therefore, $S^x(t) \models p_x \wedge i_x \wedge \neg o_x$.

According to (i) of Definition 4.1, $S^x(t) \models p_x \wedge cA_{y,x} \wedge \neg i_y \wedge o_y$.

So $S^x(t) \models o_y \wedge \neg o_y$.

This is impossible because a state is coherent.

- Let x, y such that $e = \Delta_{y,x}^2$ and let us suppose that $S^x(t) \models tri(e)$. It is equivalent to $S^x(t) \models p_x \wedge \neg o_x \wedge p_y \wedge cA_{y,x} \wedge i_y$.

According to Proposition 4.1, $S^x(t) \models p_x \wedge \neg o_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y$.

Therefore, $S^x(t) \models p_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y$.

According to (ii) of Definition 4.1, $S^x(t) \models p_x \wedge \neg i_x \wedge o_x$.

So $S^x(t) \models o_x \wedge \neg o_x$.

This is impossible because a state is coherent.

- Let x such that $e = I_x^1$ and let us suppose that $S^x(t) \models tri(e)$. It is equivalent to $S^x(t) \models p_x \wedge o_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y) \vee \neg cA_{y,x} \right)$.

According to Proposition 4.1, $S^x(t) \models p_x \wedge \neg i_x \wedge o_x$,

According to (ii) of Definition 4.1, $\exists y, S^x(t) \models p_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y$.

Therefore, $S^x(t) \models cA_{y,x} \wedge \neg i_y \wedge i_y$.

This is impossible because a state is coherent.

- Let x such that $e = I_x^2$ and let us suppose that $S^x(t) \models tri(e)$. It is equivalent to $S^x(t) \models p_x \wedge \neg i_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge o_y) \vee \neg cA_{y,x} \right)$.

According to Proposition 4.1, $S^x(t) \models p_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y \wedge o_y) \vee \neg cA_{y,x} \right)$.

According to (i) of Definition 4.1, $S^x(t) \models p_x \wedge i_x \wedge \neg o_x$.

Therefore, $S^x(t) \models i_x \wedge \neg i_x$.

This is impossible because a state is coherent.

- (ii) $\Rightarrow$ (i) Let $S^x(t)$ be a state such that $\forall e \in \mathbb{U}, S^x(t) \not\models tri(e)$. Let us prove that $S^x(t)$ is a Σ_c -argumentative state, i.e. it verifies (i) and (ii) of Definition 4.1.

- (i) $\Leftarrow$ Let x such that $S^x(t) \models p_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y \wedge o_y) \vee \neg cA_{y,x} \right)$. It is equivalent to $S^x(t) \models p_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y) \vee \neg cA_{y,x} \right) \wedge \left(\bigwedge_y (p_y \wedge cA_{y,x} \wedge o_y) \vee \neg cA_{y,x} \right)$.

As $\forall e \in \mathbb{U}, S^x(t) \not\models tri(e)$, then with $e = I_x^1$ and $e' = I_x^2$, we have that :

- $S^x(t) \not\models (p_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y) \vee \neg cA_{y,x} \right)) \wedge o_x$.
- $S^x(t) \not\models (p_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge o_y) \vee \neg cA_{y,x} \right)) \wedge \neg i_x$.

So $S^x(t) \not\models o_x$ and $S^x(t) \not\models \neg i_x$. Therefore, $S^x(t) \models p_x \wedge i_x \wedge \neg o_x$.

- (i) $\Rightarrow$ Let x such that $S^x(t) \models p_x \wedge i_x \wedge \neg o_x$. As $\forall e \in \mathbb{U}, S^x(t) \not\models tri(e)$, then $\forall y, S^x(t) \not\models tri(\Delta_{y,x}^1)$ and $S^x(t) \not\models tri(\Delta_{y,x}^2)$.

Therefore, $\forall y, S^x(t) \not\models cA_{y,x} \wedge p_y \wedge \neg o_y$ and $S^x(t) \not\models cA_{y,x} \wedge p_y \wedge i_y$.

If $S^x(t) \models p_y$ then $S^x(t) \models o_y \vee \neg cA_{y,x}$ and $S^x(t) \models \neg i_y \vee \neg cA_{y,x}$.

If $S^x(t) \models \neg p_y$ then y has never been enunciated and so is unacceptable, i.e. $S^x(t) \models \neg i_y \wedge o_y$.

Therefore, $\forall y, S^x(t) \models (cA_{y,x} \wedge o_y \wedge \neg i_y) \vee \neg cA_{y,x}$.

So $S^x(t) \models p_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y \wedge o_y) \vee \neg cA_{y,x} \right)$.

- (ii) $\Rightarrow$ Let x such that $S^x(t) \models p_x \wedge \neg i_x \wedge o_x$. As $S^x(t) \not\models tri(I_x^1)$ then $\exists y$ such that $S^x(t) \models \neg((cA_{y,x} \wedge \neg i_y) \vee \neg cA_{y,x})$ i.e. $S^x(t) \models cA_{y,x} \wedge i_y$.

According to Proposition 4.1, $S^x(t) \models cA_{y,x} \wedge i_y \wedge \neg o_y$.

Moreover, as $S^x(t) \models i_y \wedge \neg o_y$, $S^x(t) \models p_y$.

Therefore, $\exists y, S^x(t) \models p_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y$.

- (ii) $\Leftarrow$ Let x such that $\exists y, S^x(t) \models p_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y$.

As $S^x(t) \not\models tri(\Delta_{y,x}^2)$ then $S^x(t) \models o_x$.

According to Proposition 4.1, $S^x(t) \models \neg i_x \wedge o_x$.

Therefore, $S^x(t) \models p_x \wedge \neg i_x \wedge o_x$.

We proved that (i) $\Rightarrow$ (ii) and (ii) $\Rightarrow$, therefore we proved the equivalence. $\square$

C.3 Théorème 4.1

Théorème (Terminaison). *Given a Σ_c -argumentative state $S^x(t)$ and $x \in \mathcal{A}$, if $enunciate_x \in E^x(t)$, then $\exists t' > t \in \mathbb{T}$ such that $S^x(t')$ is a Σ_c -argumentative state.*

Preuve du Théorème 4.1. Let $S^x(t)$ be a Σ_c -argumentative state such that $E(t) = \{enunciate_x\}$. Let us prove that $\exists t' > t$, $S^x(t')$ is a Σ_c -argumentative state.

As $E(t) = \{enunciate_x\}$ then $\forall f \in \mathbb{F} \setminus \{p_x, i_x, o_x\}$ if $S^x(t) \models f$, $S^x(t+1) \models f$.

Moreover, $S^x(t+1) \models \text{eff}(enunciate_x) = p_x \wedge i_x \wedge \neg o_x$.

According to Lemma 4.1, as $S^x(t)$ is a Σ_c -argumentative state then $\forall e \in \mathbb{U}$, $S^x(t) \not\models \text{tri}(e)$.

So, $\forall e \in \mathbb{U} \setminus \{\Delta_{y,x}^1, \Delta_{y,x}^2, \Delta_{x,y}^1, \Delta_{x,y}^2 \mid \forall y\}$, $S^x(t) \not\models \text{tri}(e)$.

As $\forall x_1, y_1, x_2, y_2, \Delta_{y_1, x_1}^1 \succ_{\mathbb{E}} \Delta_{y_2, x_2}^2$, then if $\exists y$, $S^x(t+1) \models \text{tri}(\Delta_{y,x}^1) \vee \text{tri}(\Delta_{x,y}^1)$ then $E(t+1) = \{\Delta_{y,x}^1 \mid S^x(t+1) \models \text{tri}(\Delta_{y,x}^1)\} \cup \{\Delta_{x,y}^1 \mid S^x(t+1) \models \text{tri}(\Delta_{x,y}^1)\}$ i.e. all the exogenous events of the form $\Delta_{b,a}^1$ that can be triggered are triggered.

Then, as $\text{eff}(\Delta_{b,a}^1) = \neg i_a$, $\forall e \in E(t+2)$, e is of the form Δ^2 or I^1 .

As $\forall a, b, c, I_c^1 \succ_{\mathbb{E}} \Delta_{b,a}^2$, if $\exists z$, $S^x(t+2) \models I_z^1$ then $\forall e \in E(t+2)$, e is of the form I^1 .

Therefore, as $\text{eff}(I_z^1) = \neg o_x$, again the only events that can be triggered are of the form Δ^1 or Δ^2 .

Then because of the priority rule R_3 , we are in the same scenario as before. Therefore, the only event that are triggered at each state are Δ^1 followed by I^1 . As long as this chain is not broken, no other events can be triggered. The effects of these two classes of exogenous events being not contradictory, it can be triggered at most as much as the number of arguments. The number of arguments being finite, there exists $t_1 \geq t$ such that no event of this type is triggered.

Now, as the effects of those events cannot trigger I^2 , then the only exogenous events that can be triggered are $\Delta_{y,x}^2$ with x the argument that has just been enunciated. If it is not triggered, then we are in a state where no more exogenous events can be triggered so according to Lemma 4.1, $S^x(t_1)$ is a Σ_c -argumentative state.

Otherwise, $S^x(t_1+1) \models o_x$. Along the same line, the only events that can be triggered in this state are of the form I^2 . If nothing is triggered then we are in a Σ_c -argumentative state, otherwise $\exists y$, $S^x(t_1+2) \models i_y$.

Again, because some arguments (y_i) became *in*, the only events that can be triggered are Δ_{y_i, z_i}^2 . Finally, there is a chain of triggered with Δ^2 being followed by I^2 . As their effect are not cancelling each other, this can happened at most as much as the number of arguments. This number being finite, $\exists t_2 \geq t_1$ such that $\forall e \in \mathbb{U}$, $S^x(t_2) \not\models t_2$. According to Lemma 4.1, $S^x(t_2)$ is a Σ_c -argumentative state. □

C.4 Théorème 4.2

Théorème (Correction). Let $S^x(t)$ be a Σ_c argumentative state and $\mathcal{L}_t$ its associated labelling. Then, $\mathcal{L}_t$ is a complete labelling of the argumentation framework $AF = (Ar, \mathcal{R})$ where $Ar = \{x \in \mathcal{A} \mid S^x(t) \models p_x\}$ and $\mathcal{R} = \{(y, x) \mid S^x(t) \models p_x \wedge p_y \wedge cA_{y,x}\}$.

Preuve du Théorème 4.2. Let $S^x(t)$ be a Σ_c -argumentative state and $\mathcal{L}_t$ its associated labelling. Let us prove that $\mathcal{L}_t$ is a complete labelling of $AF = (Ar, \mathcal{R})$:

i) Let us prove that $\forall x \in Ar$, $\mathcal{L}_t(x) = IN \Leftrightarrow \forall y \in \text{Att}_x$, $\mathcal{L}_t(y) = OUT$.

Let $x \in Ar$ be an argument such that $\mathcal{L}_t(x) = IN$.

By definition of $\mathcal{L}_t$, $S^x(t) \models p_x \wedge i_x \wedge \neg o_x$.

As $S^x(t)$ is a Σ_c -argumentative state, $S^x(t) \models p_x \wedge \left(\bigwedge_y (cA_{y,x} \wedge \neg i_y \wedge o_y) \vee \neg cA_{y,x} \right)$.

Therefore, $\forall y \in \text{Att}_x$, $S^x(t) \models \neg i_y \wedge o_y$ i.e. by definition of $\mathcal{L}_t$, $\mathcal{L}_t(y) = OUT$.

ii) Let us prove that $\forall x \in Ar, \mathcal{L}_t(x) = OUT \Leftrightarrow \exists y \in Att_x, \mathcal{L}_t(y) = IN$.

Let $x \in Ar$ be an argument such that $\mathcal{L}_t(x) = OUT$.

By definition of $\mathcal{L}_t$, $S^x(t) \models p_x \wedge \neg i_x \wedge o_x$.

As $S^x(t)$ is a Σ_c -argumentative state, $\exists y, S^x(t) \models p_x \wedge p_y \wedge cA_{y,x} \wedge i_y \wedge \neg o_y$.

Therefore, $\exists y \in Att_x, S^x(t) \models i_y \wedge \neg o_y$ i.e. by definition of $\mathcal{L}_t$, $\mathcal{L}_t(y) = IN$.

□

C.5 Théorème 4.3

Théorème (Complétude de la transformation $\mathcal{T}_{AF2AL}^{lf}$). *Let AF be an AAF and $\mathcal{L}_c$ be a complete labelling on AF . Then $\exists \varsigma$ a sequence such that the associated labelling $\mathcal{L}_t$ of the final argumentative state $S^x(t)$ verifies $\mathcal{L}_c = \mathcal{L}_t$.*

Preuve du Théorème 4.3. Let $\mathcal{L}_c$ be a complete labelling for $AF = (\mathcal{A}, \mathcal{R})$. In this proof, we will build a sequence ς_c that leads to this complete labelling using the modified transformation.

At time $t = 0$, all the arguments labelled *OUT* by $\mathcal{L}_c$ are enunciated. Then when an argumentative state is reached at t_{IN} , we enunciate all the arguments labelled *IN* by $\mathcal{L}_c$. Based on the characterisation of a complete labelling provided in Caminada (2017), b is labelled *OUT* iff $\exists a \in \mathcal{A}$ that is labelled *IN*. Moreover, a is labelled *IN* iff $\forall b \in Att_a$, b is labelled *OUT*. Therefore, $\forall b$ labelled *OUT*, $\exists a$ such that $S^x(t_{IN} + 1) \models tri(\Delta_{a,b}^1)$. As arguments labelled *IN* have just been stated, their acceptability cannot be changed because there are other updates that can be triggered. After that, for the same reason, we only have $S^x(t_{IN} + 2) \models tri(\Delta_{a,b}^1)$. We reach an argumentative state $S^x(t_U)$ in which arguments labelled *IN* are acceptable and the ones labelled *OUT* are unacceptable.

Finally, we enunciate all the arguments labelled *UNDEC*. As $\forall b \in Att_a$, b is labelled *OUT* and all the other arguments are unacceptable, the only events that can be triggered are concerning the new arguments. According to Caminada (2017), an argument c is labelled *UNDEC* iff $\forall x \in Att_c$, x is not labelled *IN* and $\exists y \in Att_c$ such that y is not labelled *OUT*. Therefore, $\forall c$ labelled *UNDEC*, $\exists y$ that is labelled *UNDEC* such that $S^x(t_U) \models tri(\Delta_{y,c}^1)$. Applying this for all *UNDEC* arguments, we reach a new state $S^x(t_U + 1)$ in which all these arguments are $\neg i \wedge \neg o$, i.e. *UNDEC* for the associated labelling. □

D Annexes du chapitre 5

D.1 Preuves du chapitre 5

D.1.1 Proposition 5.1

Proposition. *Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Alors Deg_A vérifie les axiomes (A1), (A2), (A3) et (A5).*

Preuve de la Proposition 5.1. Soit $A, A' \in WAG$ des QBAF acycliques.

(A1) : Soit f un isomorphisme entre A et A' .

Soit $a \in \mathcal{A}$, montrons que $Deg_A(a) = Deg_{A'}(f(a))$.

Par définition, on a $Deg_{A'}(f(a)) = \varphi_f(\pi_{\mathcal{R}}(f(a)), \pi_{\mathcal{S}}(f(a)), w'(f(a)))$

Si a est tel que $\text{Supp}_a = \text{Att}_a = \emptyset$.

Comme f est un isomorphisme alors $\text{Supp}_{f(a)} = \text{Att}_{f(a)} = \emptyset$ et $w'(f(a)) = w(a)$

On obtient $\varphi_f(\pi_{\mathcal{R}}(f(a)), \pi_{\mathcal{S}}(f(a)), w'(f(a))) = \varphi_f(0, 0, w(a))$

$$= \varphi_f(\pi_{\mathcal{R}}(a), \pi_{\mathcal{S}}(a), w(a)) = \text{Deg}_A(a).$$

Posons $\tilde{\mathcal{A}} = \{a \in \mathcal{A} \mid \text{Deg}_A(a) = \text{Deg}_{A'}(f(a))\}$.

Comme A est acyclique, $\exists a \in \mathcal{A}$ tel que $\text{Supp}_a = \text{Att}_a = \emptyset$. Donc $\tilde{\mathcal{A}} \neq \emptyset$.

Montrons que $\tilde{\mathcal{A}} = \mathcal{A}$. En particulier comme $\tilde{\mathcal{A}} \subseteq \mathcal{A}$, on va montrer que $\tilde{\mathcal{A}} \supseteq \mathcal{A}$:

Supposons qu'il existe $a \in \mathcal{A} \setminus \tilde{\mathcal{A}}$ tel que $\text{Supp}_a \subseteq \tilde{\mathcal{A}}$ et $\text{Att}_a \subseteq \tilde{\mathcal{A}}$.

On a par définition $\text{Deg}_{A'}(f(a)) = \varphi_f(\pi_{\mathcal{R}}(f(a)), \pi_{\mathcal{S}}(f(a)), w'(f(a)))$

$$= \varphi_f(\pi_{\mathcal{R}}(f(a)), \pi_{\mathcal{S}}(f(a)), w(a)).$$

Comme $\forall b_i \in \text{Att}_a, b_i \in \tilde{\mathcal{A}}$ alors, $\text{Deg}_{A'}(f(b_i)) = \text{Deg}_A(b_i)$.

Donc on a $\varphi_{\mathcal{R}}(\text{Deg}_{A'}(f(b_1)), \dots, \text{Deg}_{A'}(f(b_n))) = \varphi_{\mathcal{R}}(\text{Deg}_A(b_1), \dots, \text{Deg}_A(b_n))$, c'est-à-dire $\pi_{\mathcal{R}}(f(a)) = \pi_{\mathcal{R}}(a)$.

De même, on montre que $\pi_{\mathcal{S}}(f(a)) = \pi_{\mathcal{S}}(a)$.

On a donc bien que a vérifie (A1) de la proposition 5.1, c'est-à-dire que $a \in \tilde{\mathcal{A}}$.

Si un tel a n'existe pas alors on a :

- Soit $\mathcal{A} = \tilde{\mathcal{A}}$. Dans ce cas, la proposition est démontrée ;
- Sinon, $\exists a_0 \in \mathcal{A} \setminus \tilde{\mathcal{A}}$ tel que $\text{Att}_{a_0} \not\subseteq \tilde{\mathcal{A}}$ ou $\text{Supp}_{a_0} \not\subseteq \tilde{\mathcal{A}}$.

Prenons par exemple $a_1 \in \text{Att}_{a_0} \setminus \tilde{\mathcal{A}}$.

De la même façon que pour a_1 , $\exists a_2 \in \text{Att}_{a_1} \setminus \tilde{\mathcal{A}}$.

Or comme le graphe est acyclique et que $\mathcal{A}$ est finie, en continuant ce raisonnement on arrive à une contradiction.

En effet, comme A est acyclique, $\exists a \in \mathcal{A}$ tel que $\text{Supp}_a = \text{Att}_a = \emptyset$ et $a \in \tilde{\mathcal{A}}$.

On a donc $\mathcal{A} = \tilde{\mathcal{A}}$.

(A2) : Si $\mathcal{A} \cap \mathcal{A}' = \emptyset$ alors $\forall a \in \mathcal{A}, \text{Att}_a^A = \text{Att}_a^{A \oplus A'}$ et $\text{Supp}_a^A = \text{Supp}_a^{A \oplus A'}$.

Par conséquent, $\pi_{\mathcal{R}}^A(a) = \pi_{\mathcal{R}}^{A \oplus A'}(a)$ et $\pi_{\mathcal{S}}^A(a) = \pi_{\mathcal{S}}^{A \oplus A'}(a)$.

Or $w^A(a) = w^{A \oplus A'}(a)$. Donc on a $\text{Deg}_A(a) = \varphi_f(\pi_{\mathcal{R}}^A(a), \pi_{\mathcal{S}}^A(a), w^A(a))$

$$= \varphi_f(\pi_{\mathcal{R}}^{A \oplus A'}(a), \pi_{\mathcal{S}}^{A \oplus A'}(a), w^{A \oplus A'}(a)) = \text{Deg}_{A \oplus A'}(a).$$

(A3) : On considère pour (A3) que $\mathcal{A} = \mathcal{A}', w = w', \mathcal{R} \subseteq \mathcal{R}'$ et $\mathcal{S} \subseteq \mathcal{S}'$.

Soit $a, b, x \in \mathcal{A}$, et $\mathcal{R}' \cup \mathcal{S}' = \mathcal{R} \cup \mathcal{S} \cup \{(a, b)\}$.

S'il n'y a pas de chemin entre b et x , alors $b \notin \text{Att}_x \cup \text{Supp}_x$.

Donc on a $\text{Att}_x^A = \text{Att}_x^{A'}$ et $\text{Supp}_x^A = \text{Supp}_x^{A'}$.

Par conséquent, $\pi_{\mathcal{R}}^A(x) = \pi_{\mathcal{R}}^{A'}(x)$ et $\pi_{\mathcal{S}}^A(x) = \pi_{\mathcal{S}}^{A'}(x)$.

Or $w^A(x) = w^{A'}(x)$. Donc on a $\text{Deg}_A(x) = \varphi_f(\pi_{\mathcal{R}}^A(x), \pi_{\mathcal{S}}^A(x), w^A(x))$

$$= \varphi_f(\pi_{\mathcal{R}}^{A'}(x), \pi_{\mathcal{S}}^{A'}(x), w^{A'}(x)) = \text{Deg}_{A'}(x).$$

(A5) : Il s'agit d'une conséquence directe des conditions aux limites que nous imposons. □

D.1.2 Proposition 5.2

Proposition (Équivalence). *Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Si $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ sont commutatives alors Deg_A vérifie l'axiome d'équivalence.*

Preuve de la Proposition 5.2. Soit $a, b \in \mathcal{A}$ tel que :

- $w(a) = w(b)$
- $\exists f : \text{Att}_a \rightarrow \text{Att}_b$ bijective telle que $\forall x \in \text{Att}_a, \text{Deg}_A(x) = \text{Deg}_A(f(x))$
- $\exists f' : \text{Supp}_a \rightarrow \text{Supp}_b$ bijective telle que : $\forall y \in \text{Supp}_a, \text{Deg}_A(y) = \text{Deg}_A(f'(y))$

Pour des raisons de lisibilité, nous notons ici $\pi_{\mathcal{R}}(f(a)) = \varphi_{\mathcal{R}}(\{\text{Deg}_A(f(x)) \mid x \in \text{Att}_a\})$.

Nous faisons pareil pour f' et $\varphi_{\mathcal{S}}$.

Comme $\forall x \in \text{Att}_a, \text{Deg}_A(x) = \text{Deg}_A(f(x))$ et que $\varphi_{\mathcal{R}}$ est commutative alors on a $\pi_{\mathcal{R}}(a) = \pi_{\mathcal{R}}(f(a)) = \pi_{\mathcal{R}}(b)$.

De même, $\pi_{\mathcal{S}}(a) = \pi_{\mathcal{S}}(f'(a)) = \pi_{\mathcal{S}}(b)$.

On obtient donc $\text{Deg}_A(a) = \varphi_f(\pi_{\mathcal{R}}(a), \pi_{\mathcal{S}}(a), w(a)) = \varphi_f(\pi_{\mathcal{R}}(f(a)), \pi_{\mathcal{S}}(f(a)), w(b))$
 $= \varphi_f(\pi_{\mathcal{R}}(b), \pi_{\mathcal{S}}(b), w(b)) = \text{Deg}_A(b).$ $\square$

D.1.3 Proposition 5.3

Proposition (Neutralité). Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Si la valeur 0 est un élément neutre pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ alors Deg_A vérifie l'axiome de neutralité.

Preuve de la Proposition 5.3. Soit $A \in \text{WAG}$, un QBAF acyclique et Deg_A une sémantique agrégative.

Soit $a, b, x \in \mathcal{A}$ tels que $w(a) = w(b), \text{Deg}_A(x) = 0, \text{Att}_a \subseteq \text{Att}_b, \text{Supp}_a \subseteq \text{Supp}_b$ et $\text{Att}_b \cup \text{Supp}_b = \text{Att}_a \cup \text{Supp}_a \cup \{x\}$.

Calculons $\text{Deg}_A(b) = \varphi_f(\varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_b\}), \varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_b\}), w(b))$.

On a d'une part $w(b) = w(a)$.

D'autre part, $\text{Att}_b \cup \text{Supp}_b = \text{Att}_a \cup \text{Supp}_a \cup \{x\}$ est équivalent à dire que :

- Soit $\text{Att}_b = \text{Att}_a \cup \{x\}$ et $\text{Supp}_b = \text{Supp}_a$.

On a d'une part $\varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_b\}) = \varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_a\})$.

D'autre part, $\varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_b\}) = \varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_a\}, \text{Deg}_A(x))$.

Comme $\text{Deg}_A(x) = 0$, on a $\varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_b\}) = \varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_a\}, 0)$.

Or 0 est élément neutre de $\varphi_{\mathcal{R}}$ donc $\varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_b\}) = \varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_a\}, 0)$.

Par conséquent, si 0 est élément neutre de $\varphi_{\mathcal{R}}$ alors $\varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_b\}) = \varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_a\})$.

- Soit $\text{Supp}_b = \text{Supp}_a \cup \{x\}$ et $\text{Att}_b = \text{Att}_a$.

On a d'une part $\varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_b\}) = \varphi_{\mathcal{R}}(\{\text{Deg}_A(c) \mid c \in \text{Att}_a\})$.

D'autre part, $\varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_b\}) = \varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_a\}, \text{Deg}_A(x))$.

Comme $\text{Deg}_A(x) = 0$, on a $\varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_b\}) = \varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_a\}, 0)$.

Or 0 est élément neutre de $\varphi_{\mathcal{S}}$ donc $\varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_b\}) = \varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_a\}, 0)$.

Par conséquent, si 0 est élément neutre de $\varphi_{\mathcal{S}}$ alors $\varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_b\}) = \varphi_{\mathcal{S}}(\{\text{Deg}_A(d) \mid d \in \text{Supp}_a\})$.

Soit $a, b, x \in \mathcal{A}$ tel que $w(a) = w(b), \text{Deg}_A(x) = 0, \text{Att}_a \subseteq \text{Att}_b, \text{Supp}_a \subseteq \text{Supp}_b$ et $\text{Att}_b \cup \text{Supp}_b = \text{Att}_a \cup \text{Supp}_a \cup \{x\}$, on a bien que :

$$0 \text{ est élément neutre de } \varphi_{\mathcal{R}} \text{ et } \varphi_{\mathcal{S}} \Rightarrow \text{Deg}_A(a) = \text{Deg}_A(b).$$

□

D.1.4 Proposition 5.4

Proposition (Renforcement). *Soit $A \in WAG$, un QBAF acyclique et Deg_A une sémantique agrégative. Si $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ sont croissantes (respectivement strictement croissantes) et φ_f est croissante selon x et décroissante selon y (respectivement strictement monotones) alors Deg_A vérifie l'axiome de renforcement (respectivement renforcement strict) au sens de l'argumentation.*

Preuve de la Proposition 5.4. Soit $A \in WAG$, un QBAF acyclique et Deg_A une sémantique agrégative, avec $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ croissantes (respectivement strictement croissantes) et φ_f décroissante selon x et croissante selon y (respectivement strictement monotones).

Soit $a, b \in \mathcal{A}$, $C, C' \subseteq \mathcal{A}$ et $x, x', y, y' \in \mathcal{A} \setminus (C \cup C')$ tels que :

- $w(a) = w(b) = w$
- $Att_a = C \cup \{x\}$, $Att_b = C \cup \{y\}$
- $Supp_a = C' \cup \{x'\}$, $Supp_b = C' \cup \{y'\}$
- $Deg_A(x) \leq Deg_A(y)$, $Deg_A(x') \geq Deg_A(y')$

Pour alléger les notations dans cette preuve, on confond les degrés d'acceptabilité des arguments d'un ensemble et l'ensemble lui-même. On note donc $\varphi_{\mathcal{R}}(C, Deg_A(x))$ au lieu de $\varphi_{\mathcal{R}}(\{Deg_A(c) \mid c \in C\}, Deg_A(x))$.

On a par croissance de $\varphi_{\mathcal{R}}$: $\varphi_{\mathcal{R}}(C, Deg_A(x)) \leq \varphi_{\mathcal{R}}(C, Deg_A(y))$.

De même, par croissance de $\varphi_{\mathcal{S}}$, on a : $\varphi_{\mathcal{S}}(C', Deg_A(x')) \geq \varphi_{\mathcal{S}}(C', Deg_A(y'))$.

Comme φ_f est décroissante selon x et croissante selon y , on a :

$$\varphi_f(\varphi_{\mathcal{R}}(C, Deg_A(x)), \varphi_{\mathcal{S}}(C', Deg_A(x')), w) \geq \varphi_f(\varphi_{\mathcal{R}}(C, Deg_A(y)), \varphi_{\mathcal{S}}(C', Deg_A(y')), w).$$

Donc on a bien que $Deg_A(a) \geq Deg_A(b)$.

La preuve est identique pour le renforcement strict avec les postulats de monotonie strictes pour les fonctions d'agrégation. □

D.1.5 Proposition 5.5

Proposition (Monotonie). *Soit A un QBAF acyclique et Deg_A une sémantique agrégative. Si 0 est un élément neutre pour $\varphi_{\mathcal{R}}$ et $\varphi_{\mathcal{S}}$ et que les trois fonctions d'agrégation sont monotones (respectivement strictement monotones), alors Deg_A vérifie l'axiome de monotonie (respectivement de monotonie stricte).*

Preuve de la Proposition 5.5. Soit $a, b \in \mathcal{A}$ tels que $w(a) = w(b)$, $Att_a \subseteq Att_b$ et $Supp_b \subseteq Supp_a$.

Notons $\mathbf{x}_a = (x_1, \dots, x_n) = Att_a$ et $\mathbf{x}_b = (x_{n+1}, \dots, x_m) = Att_b \setminus Att_a$.

Notons également $\mathbf{y}_b = (y_1, \dots, y_p) = Supp_b$ et $\mathbf{y}_a = (y_{p+1}, \dots, y_q) = Supp_a \setminus Supp_b$.

Comme dans la preuve précédente, afin d'alléger les notations, nous notons $\varphi_{\mathcal{R}}(\mathbf{x}_a)$ à la place de $\varphi_{\mathcal{R}}(\{Deg_A(x_{a,i}) \mid x_{a,i} \in \mathbf{x}_a\})$.

On a $\pi_{\mathcal{R}}(b) = \varphi_{\mathcal{R}}(\mathbf{x}_a, \mathbf{x}_b) \geq \varphi_{\mathcal{R}}(\mathbf{x}_a, \mathbf{0}) = \varphi_{\mathcal{R}}(\mathbf{x}_a) = \pi_{\mathcal{R}}(a)$.

De même, $\pi_{\mathcal{S}}(a) = \varphi_{\mathcal{S}}(\mathbf{y}_b, \mathbf{y}_a) \geq \varphi_{\mathcal{S}}(\mathbf{y}_b, \mathbf{0}) = \varphi_{\mathcal{S}}(\mathbf{y}_b) = \pi_{\mathcal{S}}(b)$.

Par décroissance de φ_f sur sa première variable et croissance sur sa seconde, on a :

$$Deg_A(b) = \varphi_f(\pi_{\mathcal{R}}(b), \pi_{\mathcal{S}}(b), w(b)) \leq \varphi_f(\pi_{\mathcal{R}}(a), \pi_{\mathcal{S}}(a), w(a)) = Deg_A(a).$$

La preuve est identique pour l'axiome de monotonie stricte avec les postulats de monotonie stricte pour les fonctions d'agrégation.



D.2 Tableau récapitulatif des axiomes de l'argumentation

Nom	Définition
(A1) : Anonymat	$\forall f$ isomorphisme de $\mathcal{A}$ vers $\mathcal{A}'$, $\forall a \in \mathcal{A}$, $Deg_{\mathcal{A}}(a) = Deg_{\mathcal{A}'}(f(a))$
(A2) : Indépendance	Si $\mathcal{A} \cap \mathcal{A}' = \emptyset$ alors $\forall a \in \mathcal{A}$, $Deg_{\mathcal{A}}(a) = Deg_{\mathcal{A} \oplus \mathcal{A}'}(a)$
(A3) : Directionnalité	Si $\mathcal{A} = \mathcal{A}'$, $w = w'$, $\mathcal{R} \subseteq \mathcal{R}'$ et $\mathcal{S} \subseteq \mathcal{S}'$ alors $\forall a, b, x \in \mathcal{A}$ si $\mathcal{R}' \cup \mathcal{S}' = \mathcal{R} \cup \mathcal{S} \cup \{(a, b)\}$ et qu'il n'y a pas de chemin entre b et x , $Deg_{\mathcal{A}}(x) = Deg_{\mathcal{A}'}(x)$
(A4) : Équivalence	$\forall a, b \in \mathcal{A}$ si : $w(a) = w(b)$ et $\exists f : Att_a \rightarrow Att_b$ et $\exists f' : Supp_a \rightarrow Supp_b$ bijectives telles que : $\forall x \in Att_a, Deg_{\mathcal{A}}(x) = Deg_{\mathcal{A}}(f(x))$ et $\forall y \in Supp_a, Deg_{\mathcal{A}}(y) = Deg_{\mathcal{A}}(f'(y))$ Alors $Deg_{\mathcal{A}}(a) = Deg_{\mathcal{A}}(b)$
(A5) : Stabilité	$\forall a \in \mathcal{A}$, si $Att_a = Supp_a = \emptyset$ alors $Deg_{\mathcal{A}}(a) = w(a)$
(A6) : Neutralité	$\forall a, b, x \in \mathcal{A}$, si $w(a) = w(b)$, $Deg_{\mathcal{A}}(x) = 0$, $Att_a \subseteq Att_b$, $Supp_a \subseteq Supp_b$ et $Att_b \cup Supp_b = Att_a \cup Supp_a \cup \{x\}$ alors
(A7) : Monotonie	$\forall a, b \in \mathcal{A}$ tels que $w(a) = w(b)$, si $Att_a \subseteq Att_b$ et $Supp_b \subseteq Supp_a$ alors Monotonie : $Deg_{\mathcal{A}}(a) \geq Deg_{\mathcal{A}}(b)$ Monotonie stricte : Si $Deg_{\mathcal{A}}(a) > 0$ et $sAtt_a \subset sAtt_b$ ou $Deg_{\mathcal{A}}(b) < 1$ et $sSupp_b \subset sSupp_a$ alors $Deg_{\mathcal{A}}(a) > Deg_{\mathcal{A}}(b)$
(A8) : Renforcement	$\forall a, b \in \mathcal{A}$, $\forall C, C' \subseteq \mathcal{A}$ et $\forall x, x', y, y' \in \mathcal{A} \setminus (C \cup C')$ tels que $w(a) = w(b)$, si $Att_a = C \cup \{x\}$, $Att_b = C \cup \{y\}$, $Supp_a = C' \cup \{x'\}$, $Supp_b = C' \cup \{y'\}$ et $Deg_{\mathcal{A}}(x) \leq Deg_{\mathcal{A}}(y)$, $Deg_{\mathcal{A}}(x') \geq Deg_{\mathcal{A}}(y')$ alors Renforcement : $Deg_{\mathcal{A}}(a) \geq Deg_{\mathcal{A}}(b)$ Renforcement strict : Si $Deg_{\mathcal{A}}(a) > 0$ et $Deg_{\mathcal{A}}(x) < Deg_{\mathcal{A}}(y)$ ou $Deg_{\mathcal{A}}(b) < 1$ et $Deg_{\mathcal{A}}(x') > Deg_{\mathcal{A}}(y')$ alors $Deg_{\mathcal{A}}(a) > Deg_{\mathcal{A}}(b)$
(A9) : Résilience	$\forall a \in \mathcal{A}$, si $0 < w(a) < 1$ alors $0 < Deg_{\mathcal{A}}(a) < 1$
(A10) : Principe de Franklin	$\forall a, b, x, y \in \mathcal{A}$, si $w(a) = w(b)$, $Att_a = Att_b \cup \{x\}$, $Supp_a = Supp_b \cup \{y\}$ et $Deg_{\mathcal{A}}(x) = Deg_{\mathcal{A}}(y)$ Franklin : $Deg_{\mathcal{A}}(a) \leq Deg_{\mathcal{A}}(b)$ Franklin Strict : $Deg_{\mathcal{A}}(a) = Deg_{\mathcal{A}}(b)$
(A11) : Affaiblissement	$\forall a \in \mathcal{A}$, $w(a) > 0$ et $\exists f : Supp_a \rightarrow Att_a$ injective telle que $\forall x \in Supp_a, Deg_{\mathcal{A}}(x) \leq Deg_{\mathcal{A}}(f(x))$ et si $sAtt_a \setminus \{f(x) x \in Supp_a\} \neq \emptyset$ ou $\exists x \in Supp_a$ tel que $Deg_{\mathcal{A}}(x) < Deg_{\mathcal{A}}(f(x))$ alors $Deg_{\mathcal{A}}(a) < w(a)$
(A12) : Consolidation (<i>Strengthening</i>)	$\forall a \in \mathcal{A}$, $w(a) > 0$ et $\exists f : Att_a \rightarrow Supp_a$ injective telle que $\forall x \in Att_a, Deg_{\mathcal{A}}(x) \leq Deg_{\mathcal{A}}(f(x))$ et si $sSupp_a \setminus \{f(x) x \in Att_a\} \neq \emptyset$ ou $\exists x \in Att_a$ tel que $Deg_{\mathcal{A}}(x) < Deg_{\mathcal{A}}(f(x))$ alors $Deg_{\mathcal{A}}(a) > w(a)$

TABLE 1 – Quelques axiomes des sémantiques graduelles en argumentation avec $\mathcal{A}$ et $\mathcal{A}'$ des WAG quelconques (Amgoud & Ben-Naim, 2018).

