Vers une acceptabilité graduelle des arguments dans les systèmes d'argumentation

Gradual acceptability in argumentation systems

C. Cayrol   M.C. Lagasquie-Schiex
ccayrol@irit.fr   lagasq@irit.fr

IRIT-UPS
118 Rte de Narbonne, 31062 Toulouse

Résumé :
L'argumentation est basée sur l'échange et l'évaluation d'arguments interagissant, puis sur la définition d'arguments (ou d'ensembles d'arguments) acceptables en fonction de l'évaluation proposée. Partant du cadre de travail proposé par [Dun95] et utilisant les évaluations graduelles proposées dans [CLS02a], nous introduisons de la gradualité dans la notion d'acceptabilité d'arguments.

Mots-clés : Théorie de l'argumentation

Abstract:
The argumentation is based on exchange and valuation of arguments interacting, then on the definition of accepted arguments (or sets of arguments) with regards to the proposed valuation. In this paper, using the argumentation system of [Dun95] and the valuations proposed in [CLS02a], we introduce graduality in the acceptability of arguments.

Keywords: Theory of argumentation

1 Introduction

[Dun95] a montré que le cadre des systèmes d'argumentation permet aussi bien l'appréhension de tâches d'argumentation intra ou inter-agents que l'étude de nombreux systèmes formels de raisonnement de sens commun ou encore la définition d'une sémantique pour les programmes logiques. L'argumentation est basée sur l'échange et l'évaluation d'arguments supportant des opinions, des assertions. On trouve des applications notamment dans le domaine juridique, dans les systèmes d'aide à la prise de décision collective ou d'aide à la négociation. La caractéristique fondamentale d'un système d'argumentation est la présence d'interactions et notamment de relations de contrariété entre les arguments avancés. Si l'argument prend par exemple la forme d'une preuve logique, on peut avancer des arguments pour une proposition et des arguments contre cette proposition, *i.e.* pour la proposition contraire (dans ce cas, la contrariété repose sur l'inconsistance logique).

Le processus d'argumentation comporte donc une étape d'évaluation de la force relative des arguments en présence, l'objectif final étant de sélectionner les arguments les plus acceptables en fonction de l'évaluation choisie. On distingue :

- une évaluation dite *intrinsèque* qui évalue un argument indépendamment des interactions avec les autres arguments. On peut ainsi exprimer à quel point l'argument augmente la confiance en l'assertion qu'il supporte. Cette évaluation peut prendre différentes formes (voir [KAEF95, Par97, PS97, AC98]).
- une évaluation *des interactions* selon laquelle un argument est évalué en fonction de ses "contrariants", des contrariants de ses contrariants (ses "défenseurs"), etc[1]. Plusieurs approches ont été proposées (voir [Dun95, AC98, JV99, BH01, CLS01, CLS02a]) qui se distinguent par la richesse de l'ensemble des valeurs disponibles pour évaluer un argument.

Évaluation intrinsèque et prise en compte des interactions ont très souvent été utilisées séparément, selon les applications envisagées. On trouve cependant quelques travaux ([AC98]) qui proposent une combinaison de ces deux critères.

La majorité des travaux consiste à definir l'acceptabilité d'un argument par son appartenance à un certain type d'ensembles (les ensembles définis comme *acceptables*) ; on parle d'*acceptabilité collective*. Le cadre de [Dun95] est tout à fait approprié à une telle approche mais conduit à des états binaires : l'argument est accepté ou non.

Notre objectif est d'introduire une gradualité

[1] Il peut exister d'autres types d'interaction (par exemple des arguments qui se renforceraient au lieu de se contrarier).

dans cette notion d'acceptabilité de manière à distinguer entre arguments plus ou moins acceptables et ainsi définir des "niveaux d'acceptabilité". Pour cela, nous allons exploiter les évaluations graduelles des interactions, par la prise en compte de la "qualité" des contrariants, des défenseurs, ...

Dans la section 2, nous nous placerons dans le cadre défini par [Dun95] : un système d'argumentation constitué d'un ensemble d'arguments et d'une relation binaire sur cet ensemble. Nous identifierons différents niveaux d'acceptabilité collective dans le cas où on prend en compte la situation d'un argument par rapport à celle de ses contrariants.

Puis, dans la section 3, nous rappellerons deux types d'évaluation graduelle issus de [CLS01] : une approche locale qui généralise certains travaux existants ([JV99, BH01]) et une approche globale introduite dans [CLS01]. Nous proposerons alors, dans la section 4, une notion d'acceptabilité graduelle basée sur les évaluations graduelles.

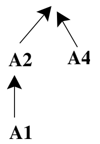
2 Acceptabilité de [Dun95]

2.1 Le cadre de [Dun95] et sa représentation graphique

Nous nous plaçons dans le cadre abstrait défini par [Dun95]. Soit le *système d'argumentation* $<\mathcal{A},\mathcal{R}>$, $\mathcal{A}$ étant un ensemble d'arguments et $\mathcal{R}$ une relation binaire sur $\mathcal{A}$ appelée *relation de contrariété* : soit A_i et $A_j \in \mathcal{A}$, $A_i\mathcal{R}A_j$ signifiera que A_j est contrarié par A_i (la notion de contrariété entre A_i et A_j implique qu'il existe un conflit entre A_i et A_j). Nous ne préciserons pas davantage le format des arguments, ni la relation de contrariété.

Notations : Soit $A \in \mathcal{A}$, l'ensemble $\{A_i \in \mathcal{A} | A_i\mathcal{R}A\}$ est noté $\mathcal{R}^-(A)$ et l'ensemble $\{A_i \in \mathcal{A} | A\mathcal{R}A_i\}$ est noté $\mathcal{R}^+(A)$. $<\mathcal{A},\mathcal{R}>$ définit un graphe orienté $\mathcal{G}$ (dit *graphe des contrariétés*).

Exemple : Le système $<\mathcal{A},\mathcal{R}>$ avec $\mathcal{A} = \{A_1, A_2, A_3, A_4\}$ et $\mathcal{R} = \{(A_2, A_3), (A_4, A_3), (A_1, A_2)\}$ définit le graphe $\mathcal{G}$ ci-contre ayant A_3 pour racine :



Déf. 1 *Soit $\mathcal{G}$ le graphe des contrariétés associé à un système d'argumentation $<\mathcal{A},\mathcal{R}>$, on définit :*

- *Un argument $A \in \mathcal{A}$ tel que $\mathcal{R}^-(A) = \varnothing$ sera une* feuille de $\mathcal{G}$.
- *Un* chemin de A vers B *est une suite d'arguments $\mathcal{C} = A_1 - \ldots - A_n$ telle que $A = A_1$, $A_1\mathcal{R}A_2$, ..., $A_{n-1}\mathcal{R}A_n$, $A_n = B$. La* longueur du chemin *est le nombre d'arcs constituant ce chemin et sera notée $l_\mathcal{C}$.*
  L'ensemble des chemins de A vers B sera noté $\mathcal{C}(A, B)$.
- *Soit 2 chemins $\mathcal{C}_A \in \mathcal{C}(A_1, A_n)$ et $\mathcal{C}_B \in \mathcal{C}(B_1, B_m)$. Ces deux chemins seront dits* dépendants *ssi $\exists A_i \in \mathcal{C}_A$, $\exists B_j \in \mathcal{C}_B$ tel que $A_i = B_j$.* Indépendants *sinon.*
  Ces deux chemins seront dits racine-dépendants en A_n *ssi $A_n = B_m$ et $\forall A_i \neq A_n \in \mathcal{C}_A$, $\nexists B_j \in \mathcal{C}_B$ tel que $A_i = B_j$.*
- *Un* circuit *est un chemin $\mathcal{C} = A_1 - \ldots - A_n - A_1$ tel que $\forall i, j \in [1, n], i \neq j$, $\nexists A_i, A_j \in \mathcal{C}$ tels que $A_i = A_j$.*
  Un circuit est isolé *quand aucun des arguments le composant n'a d'attaquant en dehors du circuit.*
  Deux circuits $\mathcal{C}_A = A_1 - \ldots - A_n - A_1$ et $\mathcal{C}_B = B_1 - \ldots - B_m - B_1$ sont interconnectés *ssi ce sont des chemins dépendants.*

Nous nous inspirons des définitions de [Dun95] pour les notions d'attaque et de défense directe ou indirecte :

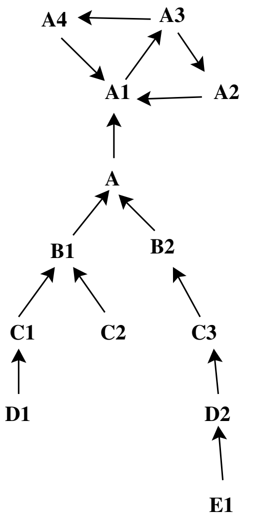
Déf. 2 *Soit $A \in \mathcal{A}$:*

- *Les* attaquants directs *de A sont les éléments de $\mathcal{R}^-(A)$.*
- *Les* défenseurs directs *de A sont les attaquants directs des éléments de $\mathcal{R}^-(A)$.*
- *Les* attaquants indirects *de A sont les éléments A_i définis par : $\exists \mathcal{C} \in \mathcal{C}(A_i, A)$ tel que $l_\mathcal{C} = 2k + 1$, avec $k \geq 1$.*
- *Les* défenseurs indirects *de A sont les éléments A_i définis par : $\exists \mathcal{C} \in \mathcal{C}(A_i, A)$ tel que $l_\mathcal{C} = 2k$, avec $k \geq 2$.*

Si l'argument A est un attaquant (direct ou indirect) de l'argument B, on dira que A *attaque* B. De même, si l'argument A est un défenseur (direct ou indirect) de l'argument B, on dira que A *défend* B.

Déf. 3 *Soit $A \in \mathcal{A}$, une* branche d'attaque *(resp.* de défense*) pour A est un chemin dans $\mathcal{G}$ d'une feuille vers A de longueur impaire (resp. paire). On dira alors que A est* racine d'une branche d'attaque *(resp.* de défense*).*

Toutes ces notions sont illustrées sur l'exemple suivant :



Sur ce graphe $\mathcal{G}$, on a donc (entre autres) :
– un chemin de C_2 vers A de longueur 2 ($C_2 - B_1 - A$),
– 2 circuits $A_1 - A_3 - A_2 - A_1$ et $A_1 - A_3 - A_4 - A_1$, chacun de longueur 3 qui ne sont pas isolés (remarquons que $A_1 - A_3 - A_2 - A_1 - A_3 - A_4 - A_1$ n'est pas un circuit d'après notre définition),
– les deux circuits cités précédemment sont interconnectés (en A_1 et A_3),
– les chemins $D_1 - C_1 - B_1$ et $C_3 - B_2 - A$ sont indépendants, alors que $D_1 - C_1 - B_1 - A$ et $C_3 - B_2 - A$ sont racine-dépendants et que $D_1 - C_1 - B_1 - A$ et $C_2 - B_1 - A$ sont dépendants,
– D_1, C_2, E_1 sont les feuilles de $\mathcal{G}$,
– $D_1 - C_1 - B_1 - A$ est une branche d'attaque pour A, alors que $C_2 - B_1 - A$ est une branche de défense pour A,
– B_1 et B_2 sont les deux attaquants directs de A,
– C_1, C_2 et C_3 sont les trois défenseurs directs de A,
– D_1 et D_2 sont les deux attaquants indirects de A,
– E_1 est le seul défenseur indirect de A.

2.2 Acceptabilité collective

L'idée exploitée ici est que l'acceptabilité d'un argument dépend de son appartenance à certains ensembles (dits *ensembles acceptables* ou *extensions*) caractérisés par des propriétés telles que :

Déf. 4 *Soit* $<\mathcal{A}, \mathcal{R}>$ *un système d'argumentation, on a :*

– *Un ensemble* $E \subseteq \mathcal{A}$ *est* sans conflit *si et seulement si* $\nexists A, B \in E$ *tel que* $A\mathcal{R}B$.
– *Soit* $E \subseteq \mathcal{A}$, $A \in \mathcal{A}$. E défend (collectivement) A *si et seulement si* $\forall B \in \mathcal{A}$, *si* $B\mathcal{R}A, \exists C \in E$ *tel que* $C\mathcal{R}B$.

[Dun95] définit plusieurs sémantiques pour l'acceptabilité collective dont les *sémantiques admissible*, *préférée* et *stable* (avec, pour extensions respectives, les ensembles admissibles, les extensions préférées et les extensions stables) :

Déf. 5 *Soit* $<\mathcal{A}, \mathcal{R}>$ *un système d'argumentation. Soit* $E \subseteq \mathcal{A}$.

– E *est* admissible *si et seulement si* E *est sans conflit et* E *défend tous ses éléments.*
– E *est une* extension préférée *si et seulement si* E *est maximal pour l'inclusion parmi les ensembles admissibles.*
– E *est une* extension stable *si et seulement si* E *est sans conflit et* E *contrarie tout argument n'appartenant pas à* E *(c'est-à-dire* $\forall A \in \mathcal{A} \setminus E, \exists B \in E$ *tel que* $B\mathcal{R}A$*).*

Remarquons que chaque contrariant est pris en compte séparément (il n'y a pas de notion de contrariété globale sur un argument). Nous rappelons aussi quelques propriétés essentielles démontrées dans [Dun95] :

Prop. 1 *Soit* $<\mathcal{A}, \mathcal{R}>$ *un système d'argumentation, on a :*

1. *Tout ensemble admissible est contenu dans une extension préférée.*
2. *Il existe au moins une extension préférée.*
3. *Si* $<\mathcal{A}, \mathcal{R}>$ *est bien-fondé*[2] *alors il possède une et une seule extension préférée qui est aussi la seule extension stable.*
4. *Toute extension stable est aussi une extension préférée (et non vice-versa).*
5. *Il n'existe pas toujours d'extension stable.*

2.3 Différents niveaux d'acceptabilité

Niveaux de base. Sous une sémantique donnée, l'acceptabilité d'un argument dépend, d'après Dung, de son appartenance à une extension de cette sémantique. Cela conduit à trois états possibles :

[2] Un système d'argumentation sera dit *bien-fondé* si et seulement s'il n'existe pas de séquence infinie $A_0, A_1, \ldots, A_n, \ldots$ telle que $\forall i, A_i \in \mathcal{A}$ et $A_{i+1}\mathcal{R}A_i$.

– soit l'argument est *uni-accepté*, car il appartient à toutes les extensions pour cette sémantique,
– soit l'argument est *exi-accepté*, car il appartient à au moins une extension pour cette sémantique,
– soit il est *non-accepté* car il n'appartient à aucune extension pour cette sémantique.

Ces niveaux ne permettent pas de distinguer deux arguments exi-acceptés se contrariant l'un l'autre.

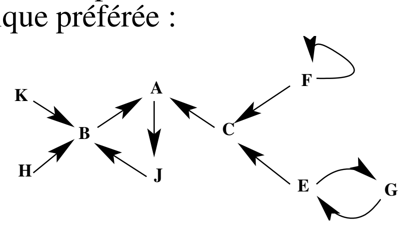
Premier raffinement des niveaux de base. On propose alors une nouvelle définition, celle d'un argument *proprement-accepté*, qui tient compte de la situation d'un argument par rapport à ses contrariants pour une sémantique donnée S et ainsi permet de raffiner la classe des arguments exi-acceptés.

Déf. 6 *A est* proprement-accepté *pour S ssi A appartient à une extension pour S et $\forall B \in \mathcal{A}$ tel que $B\mathcal{R}A$, B n'appartient pas à une extension pour S.*

Ainsi, on essaye de capturer l'idée qu'un argument sera meilleur du point de vue de l'acceptabilité si ses attaquants directs ne sont pas acceptés.

Prop. 2 ([CLS02b]) *Soit S une sémantique au sens de [Dun95]. Soit $A \in \mathcal{A}$. Si A est uni-accepté alors A est proprement-accepté.*

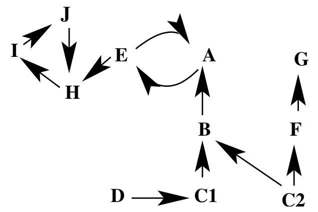
La réciproque est fausse, en général, comme le montre l'exemple suivant dans le cadre de la sémantique préférée :



On a ici 2 extensions préférées $\{K, H, G\}$ et $\{A, E, K, H\}$. L'argument A est proprement-accepté mais pas uni-accepté puisqu'il n'appartient pas à toutes les extensions préférées.

Cette notion permet de raffiner les niveaux déjà vus. Pour une sémantique S et un argument A, on a les différents états :
– A est *uni-accepté* (donc proprement-accepté),
– A est *proprement-accepté* (donc par définition exi-accepté),
– A est *seulement-exi-accepté*, si A n'est pas proprement-accepté mais exi-accepté.
– A est *non-accepté*.

Un exemple Soit le système d'argumentation suivant



On a deux extensions préférées $\{D, C_2, A, G\}$ et $\{D, C_2, I, E, G\}$. Donc, pour la sémantique préférée, les niveaux d'acceptabilité sont les suivants :
– D, C_2 et G sont uni-acceptés,
– I est proprement-accepté,
– A et E sont seulement-exi-acceptés,
– B, C_1, H, J et F sont non-acceptés.

Quelques cas particuliers. Dans tous les cas où il n'existe qu'une seule extension, les trois premières classes se confondent :
– Sous la sémantique préférée, quand il n'y a pas de circuit pair (voir [Dou02]).
– Sous la sémantique de base (c'est une autre sémantique proposée par Dung et non présentée ici qui présente la particularité de n'avoir qu'une seule extension).

Mais on a aussi (voir les preuves dans [CLS02b]) :
– Sous la sémantique stable, la classe des uni-acceptés se confond avec celle des proprement-acceptés.
– Sous la sémantique préférée, quand il n'y a pas de circuit impair, la classe des uni-acceptés se confond avec celle des proprement-acceptés.

3 Évaluation graduelle des interactions

Nous avons proposé deux méthodes d'évaluation distinctes dont l'objectif est de tenir compte de la qualité des attaquants et des défenseurs d'un argument.

3.1 L'approche "locale" de [CLS01] (évaluation générique)

Cette approche propose de calculer la valeur d'un argument uniquement en fonction des valeurs de ses contrariants. Il s'agit d'une approche générique qui recouvre des travaux existants ([BH01, JV99]). On formalise cette évaluation en prenant l'hypothèse suivante : on dispose d'un ensemble W totalement ordonné par une relation notée $\geq$ et admettant un plus petit élément (V_{Min}) et d'une partie V de W, contenant V_{Min} et admettant un plus grand élément V_{Max}.

Déf. 7 *Soit $<\mathcal{A}, \mathcal{R}>$ un système d'argumentation. Une évaluation est une application $v : \mathcal{A} \to V$ telle que :*

1. $\forall A \in \mathcal{A}$, $v(A) \geq V_{Min}$
2. $\forall A \in \mathcal{A}$, *si* $\mathcal{R}^-(A) = \varnothing$, *alors* $v(A) = V_{Max}$
3. $\forall A \in \mathcal{A}$, *si* $\mathcal{R}^-(A) = \{A_1, \ldots, A_n\} \neq \varnothing$, *alors* $v(A) = g(h(v(A_1), \ldots, v(A_n)))$

où $h : V^* \to W$ telle que (V^* dénote l'ensemble des suites finies d'éléments de V)
- $h(x) = x$,
- $h() = V_{Min}$ *et*
- $h(x_1, \ldots, x_n, x_{n+1}) \geq h(x_1, \ldots, x_n)$,

et $g : W \to V$ telle que
- $g(V_{Min}) = V_{Max}$,
- $g(V_{Max}) < V_{Max}$ *et*
- *g est decroissante (si $x \leq y$ alors $g(x) \geq g(y)$).*

Remarquons que $h(x_1, \ldots, x_n) \geq max(x_1, \ldots, x_n)$ est une conséquence des propriétés de la fonction h. Rappelons aussi quelques propriétés issues de [CLS01] :

Prop. 3 *Une évaluation v donnée par la définition 7 induit un pré-ordre total $\succeq$ sur l'ensemble des arguments $\mathcal{A}$ défini par : $A \succeq B$ ssi $v(A) \geq v(B)$.*

Prop. 4 *Soit un circuit isolé de longueur n dans le graphe des contrariétés. Si n est impair, tous les arguments du circuit ont la même valeur qui est un point fixe de g, et si n est pair, chaque argument du circuit a pour valeur un point fixe de g^n.*

Prop. 5 *L'évaluation graduelle donnée par la définition 7 respecte les principes suivants :*

P1 *L'évaluation est maximale pour un argument sans attaquant et non maximale pour un argument attaqué et non défendu.*

P2 *L'évaluation d'un argument est fonction de l'évaluation de tous les attaquants directs (la contrariété directe).*

P3 *L'évaluation d'un argument est une fonction décroissante de l'évaluation de la contrariété directe.*

P4 *La contribution d'un attaquant direct d'un argument ne peut pas diminuer l'évaluation de la contrariété directe pour cet argument.*

La définition 7 caractérise une classe d'évaluations graduelles dont certaines instances sont déjà connues ([JV99][3],[BH01][4]) :

Prop. 6 (Lien avec [JV99]) *Tout étiquetage enraciné de $<\mathcal{A}, \mathcal{R}>$ au sens de [JV99] peut être défini comme une instance v de l'évaluation générique telle que : $V = W = \{-, ?, +\}$ avec $- < ? < +$, V_{Min} dénoté par $-$, V_{Max} dénoté par $+$, g définie par $g(-) = +$, $g(+) = -$, $g(?) = ?$ et h étant la fonction max.*

Prop. 7 (Lien avec [BH01]) *La fonction d'évaluation graduelle de [BH01] peut être définie comme une instance v de l'évaluation générique telle que : $V = [0, 1]$, $W = [0, \infty[$, $V_{Min} = 0$, $V_{Max} = 1$, $g : W \to V$ définie par $g(x) = \frac{1}{1+x}$ et h par $h(\{x_1, \ldots, x_n\}) = x_1 + \ldots + x_n$.*

3.2 L'approche "globale" de [CLS01] (évaluation par tuples)

Ici, l'idée est d'avoir une étiquette qui reflète la structure de la partie du graphe des contrariétés dont l'argument étiqueté est la racine. Pour cela, nous utiliserons des tuples.

[3][JV99] définit plusieurs types d'étiquetage construits autour de 3 valeurs : $+$ (la meilleure valeur), $-$ (la plus mauvaise valeur) et ? (une valeur intermédiaire).

[4][BH01] définit un étiquetage avec des réels $\in [0, 1]$, 1 étant la meilleure valeur. Dans ces travaux, le calcul de $v(A)$ sachant que A est attaqué par $B_1, \ldots, B_n$ est le suivant : $v(A) = \frac{1}{1+\Sigma_{i=1}^{n} v(B_i)}$. Initialement, ces travaux ne concernaient qu'un graphe sans circuit.

L'étiquetage d'un argument par un tuple. L'étiquette pour un argument A est donc un tuple répertoriant les longueurs de toutes les branches d'attaque et de toutes les branches de défense menant à A. Le cas d'un graphe des contrariétés comportant des circuits fait l'objet de règles spécifiques dans lesquelles on considère le circuit comme un méta-argument possédant deux branches, une d'attaque de longueur 1 et une de défense de longueur 2 (chaque argument du circuit est à la fois attaqué et défendu par le circuit) sur lequel se rajoute ensuite l'impact des attaquants extérieurs au circuit et celui des autres circuits interconnectés. Toutes ces idées ont débouché dans [CLS01] sur les règles suivantes :

Règle 1 *Soit l'argument A n'appartenant pas à un circuit.*

- *si A n'est pas attaqué alors* $v(A) = ()$ *(aussi notée : $v(A) = (0)$)*
- *si A a pour attaquants directs les arguments $B_1, \ldots, B_n$ dont les valeurs respectives sont les tuples $(b_1^1, \ldots, b_{m_1}^1), \ldots, (b_1^n, \ldots, b_{m_n}^n)$ alors*

$$v(A) = (b_1^1 + 1, \ldots, b_{m_1}^1 + 1, \ldots, b_1^n + 1, \ldots, b_{m_n}^n + 1)$$

Règle 2 *Soit un argument A appartenant aux circuits $\mathtt{C}_1 \ldots \mathtt{C}_m$. Soit $\mathtt{C}'_1 \ldots \mathtt{C}'_n$ autres circuits interconnectés à l'un des $\mathtt{C}_i$ ou entre eux*[5]*. Soit $X^1 \ldots X^p$ arguments n'appartenant pas aux circuits mais attaquants directs de l'un des circuits*[6]*. On note :*
- *l_i : longueur minimale d'un chemin d'un élément de $\mathtt{C}'_i$ vers A,*
- *l_{X^i} (longueur*[7] *d'un chemin de X^i vers A),*
- *chaque argument X^i a pour valeur : $v(X^i) = (x_1^i, \ldots, x_{k^i}^i)$.*

On a alors :

$$\begin{aligned} v(A) = (&\overbrace{1, 2, \ldots, 1, 2,}^{m\ fois} \\ &1 + l_1, 2 + l_1, \ldots, 1 + l_n, 2 + l_n, \\ &\left.\begin{array}{l} x_1^1 + l_{X^1}, \ldots, x_{k^1}^1 + l_{X^1}, \ldots, \\ x_1^p + l_{X^p}, \ldots, x_{k^p}^p + l_{X_p}) \end{array}\right\} \star \end{aligned}$$

($\star$ autant de fois qu'il y a de chemins de X^i vers A)

[5] A n'appartient donc à aucun des $\mathtt{C}'_i$.

[6] C'est-à-dire attaquants directs d'éléments des circuits : $\forall X^i, \exists \mathtt{C} \in \{\mathtt{C}_1, \ldots, \mathtt{C}_m, \mathtt{C}'_1, \ldots, \mathtt{C}'_n\}$ tel que $\exists Y \in \mathtt{C}$ et $X^i \mathcal{R} Y$.

[7] S'il y a plusieurs chemins menant de X^i à A, on prendra en compte bien-sûr les longueurs de chacun de ces chemins.

Dans [CLS01], le lecteur trouvera de nombreux exemples d'étiquetage complet.

Déf. 8 *Soit v une application de $\mathcal{A}$ dans l'ensemble des suites d'entiers naturels, v sera une* évaluation par tuples *si et seulement si v respecte les règles 1 et 2.*

Comparaison d'arguments à l'aide de tuples. Étant donné que, dans un tuple, les valeurs paires (correspondant aux branches de défense) et les valeurs impaires (correspondant aux branches d'attaque) ne jouent pas les mêmes rôles, nous ne pouvons pas nous contenter d'une simple comparaison lexicographique. Nous avons donc choisi de comparer les tuples en deux phases bien distinctes en adoptant une *attitude prudente* :
- Une "première passe" permet de déterminer et *comparer le nombre de branches d'attaque et le nombre de branches de défense* de chacun. Cela nous donne deux critères (un pour la défense et un pour l'attaque). Si ces deux critères sont en accord, c'est-à-dire que l'un des tuples a plus de branches de défense et moins de branches d'attaque que l'autre, alors on peut conclure. De même, si les deux critères sont en désaccord (l'un des tuples a plus de branches de défense et plus de branches d'attaque que l'autre), les tuples sont considérés comme incomparables.
- Sinon, les deux tuples ayant le même nombre d'attaques et le même nombre de défenses, une "seconde passe" *compare la qualité respective des attaques et des défenses*. La comparaison porte d'une part sur les "parties paires" des tuples, d'autre part sur les "parties impaires" des tuples. Ici aussi, en cas de désaccord, la conclusion est l'incomparabilité des tuples ! Cette comparaison utilise un principe lexicographique.

Cette méthode débouche sur un algorithme décrit dans [CLS01, CLS02b]. À titre d'exemples :
- $(1, 2)$ est meilleur que $(1, 1, 2)$ car on a moins de branches d'attaque dans le premier tuple que dans le second pour un nombre de branches de défense identique.
- $(1, 2)$ est incomparable avec $(1, 1, 2, 2)$ car on a, dans le premier tuple, à la fois moins de branches d'attaque et moins de branches de défense que dans le second.
- $(2, 3)$ est meilleur que $(1, 2)$ car on a, dans le premier tuple, de meilleures branches d'attaque que dans le second (elles sont plus

longues) pour des branches de défense identiques.
- $(2,3)$ est meilleur que $(3,4)$ car on a, dans le premier tuple, de meilleures branches de défense que dans le second (elles sont plus courtes) pour des branches d'attaque identiques.
- $(1,2)$ est incomparable avec $(3,4)$ car on a, dans le premier tuple, à la fois de plus mauvaises branches d'attaque et de meilleures branches de défense que dans le second.

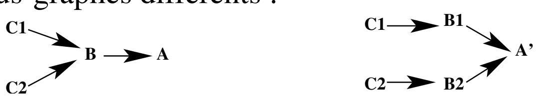
Quelques propriétés. Soit v une évaluation par tuples, on a les propriétés suivantes (preuves dans [CLS01, CLS02b]) :

Prop. 8 *La fonction d'évaluation par tuples associée à l'algorithme décrit dans [CLS01, CLS02a] induit un pré-ordre partiel $\succeq$ sur l'ensemble des arguments dont les éléments maximaux seront les feuilles ayant pour valeur $()$.*

L'argument $\perp$ ayant pour valeur le tuple $\underbrace{(1,\ldots,1)}_{\infty}$ est la valeur minimale[8] *pour la relation $\succeq$.*

Prop. 9 *Soit un argument A ayant pour attaquants directs A_1 de valeur $v(A_1) = (a^1_1,\ldots,a^1_{m_1})$, …, A_n de valeur $v(A_n) = (a^n_1,\ldots,a^n_{m_n})$. Soit l'argument A' ayant pour attaquants directs A^1_1 de valeur $v(A^1_1) = (a^1_1)$, …, $A^1_{m_1}$ de valeur $v(A^1_{m_1}) = (a^1_{m_1})$, …, A^n_1 de valeur $v(A^n_1) = (a^n_1)$, …, $A^n_{m_n}$ de valeur $v(A^n_{m_n}) = (a^n_{m_n})$. Alors $v(A) = v(A')$.*

Cette propriété montre bien l'aspect "indépendant" de la prise en compte de chaque branche même quand les dites-branches ne sont pas indépendantes graphiquement ! Sur l'exemple suivant, A et A' ont exactement la même valeur $(2,2)$ bien qu'ils soient racine de sous-graphes différents :



Prop. 10 *Soit v une évaluation par tuples (donc définie par les règles 1 et 2) associée à l'algorithme décrit dans [CLS01, CLS02a], v respecte les principes suivants :*

P1′ *L'évaluation est maximale pour un argument sans attaquant et non maximale pour un argument attaqué (qu'il soit défendu ou pas).*

P2′ *L'évaluation d'un argument prend en compte toutes les branches dont cet argument est racine.*

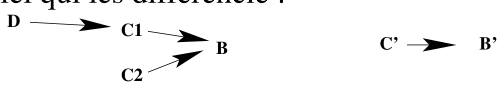
P3′ *L'amélioration de la défense ou la détérioration de l'attaque*[9] *conduisent à augmenter la valeur d'un argument.*

P4′ *L'amélioration de l'attaque ou la détérioration de la défense conduisent à diminuer la valeur d'un argument.*

3.3 Différences essentielles entre évaluations "locale" et "globale"

Dans [CLS01], le lecteur trouvera une comparaison de ces approches avec les autres approches du domaine ([Dun95, JV99, BH01]), ainsi qu'une comparaison approche "locale" – approche "globale".

Rappelons toutefois sur un exemple le point essentiel qui les différencie :



Sur cet exemple, avec l'approche locale, on constate que B a deux attaquants directs (C_2 de valeur maximale et C_1) alors que B' n'en a qu'un (C' de valeur maximale). Donc B' est meilleur que B.

Avec l'approche globale, on constate que deux branches (une d'attaque et une de défense) mènent à B alors qu'il n'y a qu'une branche d'attaque qui mène à B'. Donc B est meilleur que B' (puisqu'il a au moins une défense alors que B' n'en a aucune). Dans ce cas-là, l'attaquant C_1 a perdu son statut négatif d'attaquant, puisque il est en fait "porteur d'une défense" pour B !

4 Acceptabilité graduelle

L'objectif est d'exploiter l'évaluation graduelle des interactions pour définir de nouveaux niveaux d'acceptabilité collective.

[8] Remarquons que cet argument correspond au cas d'un argument contrarié par une infinité d'arguments eux-mêmes jamais attaqués.

[9] Les notions d'amélioration/détérioration de la défense/attaque sont formalisées dans [CLS02b].

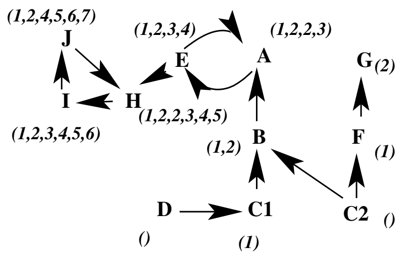
4.1 Comparaison basée sur l'évaluation graduelle

Soit v une évaluation graduelle, on peut définir une *v-préférence* utilisant le pré-ordre $\succeq$ issu de v :

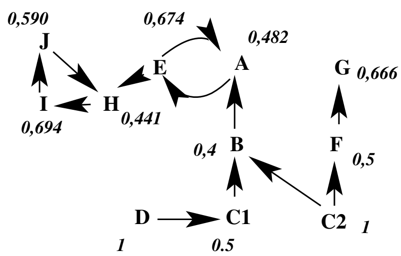
Déf. 9 *Soit $A, B \in \mathcal{A}$, A est v-*préféré à *B ssi $B \not\succ A$.*

Dans le cas où l'évaluation v induit un pré-ordre total sur l'ensemble des arguments, on a : A est *v-préféré à B* ssi $A \succeq B$.

La v-préférence peut être utilisée au sein d'un niveau d'acceptabilité (par exemple, celui des arguments exi-acceptés) pour obtenir des arguments mieux acceptés que d'autres. Illustrons sur un même graphe les résultats fournis par les v-préférences associées à deux évaluations graduelles différentes :



Évaluation par tuples



Évaluation de [BH01]

Avec l'évaluation par tuples, certains arguments étant incomparables, on obtient

$$D, C_2 \succ G \succ \left| \begin{array}{ccc} A & \succ & E \\ & B & \\ H & \succ & I \\ J & \succ & I \end{array} \right| \succ F, C_1$$

Avec l'évaluation de [BH01], on obtient $D, C_2 \succ I \succ E \succ G \succ J \succ C_1, F \succ A \succ H \succ B$.

Si on applique cette v-préférence sans respecter les niveaux d'acceptabilité définis en section 2.3, on arrive à des situations contre-intuitives (par exemple, avec [BH01], E sera v-préféré à G alors que G est uni-accepté et E seulement-exi-accepté !).

4.2 Situation d'un argument par rapport à ses contrariants

On peut aussi utiliser directement la v-préférence pour situer, du point de vue de l'acceptabilité, un argument et ses contrariants.

Déf. 10 *Soit $A \in \mathcal{A}$, A est* bien-défendu pour v *ssi $\forall B \in \mathcal{A}$ tel que $B \mathcal{R} A$, A est v-préféré à B.*

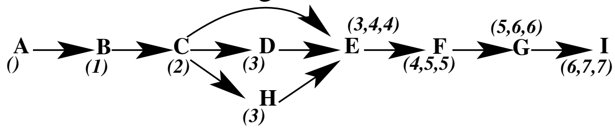
Ainsi, on essaye de capturer l'idée qu'un argument sera mieux accepté s'il est au moins aussi bon que ses attaquants directs (ou incomparable avec eux dans le cas d'un ordre partiel !). L'ensemble des arguments bien-défendus va dépendre de l'évaluation choisie. Si on reprend l'exemple donné en section 4.1, on constate que les arguments bien-défendus sont D, C_2, G, J et A pour l'évaluation par tuples, alors qu'avec l'évaluation de [BH01] les arguments bien-défendus sont D, C_2, G, I et E.

Remarquons aussi que, comme dans le cas des sémantiques de [Dun95], la définition 10 considère les contrariants un par un. Elle n'est donc pas adaptée à une évaluation qui utiliserait la contrariété dans sa globalité (ce qui est par exemple le cas dans [BH01]).

4.3 Compatibilité entre acceptabilité collective et évaluation graduelle

Nous disposons donc d'une nouvelle notion d'acceptabilité (définition 10) dont il faut étudier la compatibilité avec la partition en niveaux proposée en section 2.3.

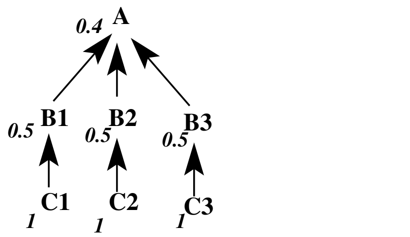
On constate très vite sur un exemple que ces deux approches ne sont pas compatibles dans le cas de l'évaluation globale :



1 seule ext préférée et stable = {A,C,F,I}
G n'appartient pas à l'ext préférée
v(I) < v(G) et v(G) > v(F)

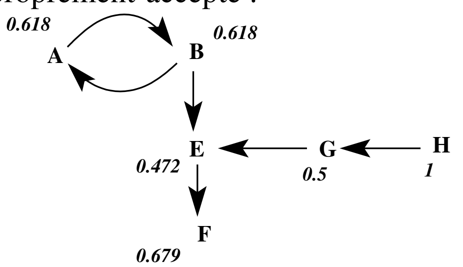
Ici, l'argument I est proprement-accepté mais pas bien-défendu, alors que c'est l'inverse pour l'argument G.

De même pour les évaluations locales (ici avec [BH01]), on peut exhiber des exemples montrant la non-compatibilité :
- l'argument A est proprement-accepté mais pas bien-défendu :



1 seule ext préférée et stable = { C1, C2, C3, A}
B1, B2, B3 n'appartiennent à aucune ext préférée
v(A) < v(Bi) quel que soit i = 1, 2, 3

- l'argument F est bien-défendu mais pas proprement-accepté :



2 ext préférées et stables = {A,H,E} et {B,H,F}
F appartient à 1 ext préférée et stable
v(F) > v(E) et E appartient aussi à 1 ext préférée et stable alors que E attaque F

Cas particuliers menant à la compatibilité. Nous nous plaçons dans le contexte d'un système d'argumentation avec une relation $\mathcal{R}$ finie et n'admettant aucun circuit[10]. Les sémantiques stable et préférée ne proposent alors qu'une seule extension et les niveaux d'acceptabilité uni-acceptés, exi-acceptés, proprement-acceptés fusionnent. Dans ce contexte, il y a au moins deux cas particuliers menant à une compatibilité. Le premier concerne l'évaluation graduelle globale par tuples (preuve dans [CLS02b]) :

Prop. 11 *Soit $\mathcal{G}$ le graphe associé à $<\mathcal{A}, \mathcal{R}>$ un système d'argumentation avec $\mathcal{R}$ finie et n'admettant aucun circuit et vérifiant la condition suivante : $\exists\ A \in \mathcal{A}$ tel que*
- *$\forall X_i$, $\forall i = 1 \dots n$, feuille de $\mathcal{G}$, $\exists$ un seul chemin de X_i vers A, $X_i^1 - \dots - X_i^{l_i} - A$ avec $X_i^1 = X_i$ et l_i la longueur de ce chemin,*
- *tous les chemins de X_i vers A sont racine-dépendants en A,*
- *$\forall A_i \in \mathcal{A}$, $\exists X_j$ une feuille telle que A_i appartienne à un chemin de X_j vers A.*

Soit v une évaluation par tuples. Soit S une sémantique $\in$ {préférée, stable}.

$\forall A \in \mathcal{A}$, A exi-accepté pour $S \Leftrightarrow A$ bien-défendu pour v.

Le second cas concerne l'évaluation graduelle locale générique (preuve dans [CLS02b]) :

Prop. 12 *Soit $<\mathcal{A}, \mathcal{R}>$ un système d'argumentation avec $\mathcal{R}$ finie et n'admettant aucun circuit. Soit S une sémantique $\in$ {préférée, stable}. Soit v une évaluation graduelle locale vérifiant la condition (i) suivante :*

$$(\forall i = 1 \dots n, g(x_i) \geq x_i) \Rightarrow (g(h(x_1, \dots, x_n)) \geq h(x_1, \dots, x_n))$$

$\forall A \in \mathcal{A}$, A exi-accepté pour $S \Leftrightarrow A$ bien-défendu pour v.

Cette propriété utilise le lemme suivant (preuve dans [CLS02b]) :

Lemme 1 *Soit $<\mathcal{A}, \mathcal{R}>$ un système d'argumentation avec $\mathcal{R}$ finie et n'admettant aucun circuit. Soit S une sémantique $\in$ {préférée, stable}. Soit v une évaluation graduelle générique vérifiant la condition (i).*
- *Si A exi-accepté et A n'a qu'un seul contrariant noté B alors $v(A) \geq v(B)$.*
- *Si B non-accepté et B n'a qu'un seul contrariant noté C alors $v(B) \leq v(C)$.*

Remarquons que cette condition (i) est (preuve dans [CLS02b]) :
- fausse pour l'évaluation locale proposée par [BH01],
- vraie pour les évaluations locales définies avec $h = Max$, $(\forall g)$,
- fausse pour les évaluations locales définies avec h telle que $\exists n > 1$ avec $h(x_1, \dots, x_n) > Max(x_1, \dots, x_n)$ ($\forall g$ strictement décroissante).

5 Conclusion

Dans ce papier, nous avons cherché à gradualiser la notion d'acceptabilité d'un argument. Pour cela, nous nous sommes appuyées sur

[10] Il s'agit donc d'un système d'argumentation $(\mathcal{A}, \mathcal{R})$ bien-fondé.

la notion d'acceptabilité collective de [Dun95] qui permet déjà d'avoir 3 niveaux d'acceptabilité (uni-acceptés, exi-acceptés, non-acceptés). Nous avons alors pris en compte à la fois le statut de l'argument par rapport à ses contrariants mais aussi une évaluation graduelle des arguments dans la notion d'acceptabilité collective.

Cela nous a amené à proposer diverses notions applicables à un argument :
– les arguments *proprement acceptés* : ceux dont les contrariants ne sont pas acceptés (pour une sémantique donnée),
– la v-*préférence* entre arguments issue directement d'une évaluation v,
– les arguments *bien-défendus* : ceux qui sont v-préférés à leurs contrariants (pour une évaluation graduelle v donnée).

La première notion permet de raffiner le niveau des exi-acceptés en deux sous-niveaux (les proprement-acceptés et les seulement-exi-acceptés). La seconde notion permet de gradualiser la totalité des niveaux d'acceptabilité mais seulement en l'utilisant à l'intérieur de chaque niveau. La troisième notion permet de définir deux nouveaux niveaux d'acceptabilité (les bien-défendus et les pas-bien-défendus). Malheureusement, nous avons constaté sur divers exemples que, dans le cas général, ces niveaux n'étaient pas compatibles avec ceux proposés par [Dun95] et raffinés à l'aide de la première notion (l'acceptabilité propre) ; et cela, bien que les notions d'argument bien-défendu et d'argument proprement-accepté soient toutes les deux construites à partir des interactions entre arguments. Certains cas particuliers sur lesquels la compatibilité existe ont été identifiés.

Références

[AC98] Amgoud (Leila) et Cayrol (Claudette). – On the acceptability of arguments in preference-based argumentation. *In : Proc. of the 14th Uncertainty in Artificial Intelligence*, éd. par Cooper (G. F.) et Moral (S.). pp. 1–7. – Madison, Wisconsin, 1998.

[BH01] Besnard (Philippe) et Hunter (Anthony). – A logic-based theory of deductive arguments. *Artificial Intelligence*, vol. 128 (1-2), 2001, pp. 203–235.

[CLS01] Cayrol (Claudette) et Lagasquie-Schiex (Marie-Christine). – *Évaluation graduelle d'arguments*. – Rapport de recherche n° 2001-18-R, France, Institut de Recherche en Informatique de Toulouse (I.R.I.T.), novembre 2001.

[CLS02a] Cayrol (Claudette) et Lagasquie-Schiex (Marie-Christine). – Gradual handling of contradiction in argumentation frameworks. *In : Proc. of the 9th IPMU (Information Processing and Management of Uncertainty in Knowledge-Based Systems)*, pp. 83–90. – Annecy, France, 2002.

[CLS02b] Cayrol (Claudette) et Lagasquie-Schiex (Marie-Christine). – *Vers une acceptabilité graduelle des arguments dans les sytèmes d'argumentation*. – Rapport de recherche n° 2002-42-R, France, Institut de Recherche en Informatique de Toulouse (I.R.I.T.), décembre 2002.

[Dou02] Doutre (Sylvie). – *Autour de la sémantique préférée des systèmes d'argumentation*. – Thèse, Université Paul Sabatier, IRIT, 2002.

[Dun95] Dung (Phan Minh). – On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, vol. 77, 1995, pp. 321–357.

[JV99] Jakobovits (Hadassa) et Vermeir (Dick). – Robust semantics for argumentation frameworks. *Journal of logic and computation*, vol. 9(2), 1999, pp. 215–261.

[KAEF95] Krause (Paul), Ambler (Simon), Elvang (Morten) et Fox (John). – A logic of argumentation for reasoning under uncertainty. *Computational Intelligence*, vol. 11 (1), 1995, pp. 113–131.

[Par97] Parsons (Simon). – Normative argumentation and qualitative probability. *LNAI*, vol. 1244, 1997, pp. 466–480.

[PS97] Prakken (Henry) et Sartor (Giovanni). – Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, vol. 7, 1997, pp. 25–75.