Université Paris Cité

LIPADE
Laboratoire d'Informatique PAris DEscartes

Habilitation à diriger des recherches

présentée par

Elise Bonzon

Interaction in Multiagent Systems

COORDINATEUR:
PAVLOS MORAITIS, Professor, Université Paris Cité

RAPPORTEURS:
KATIE ATKINSON, Professor, University of Liverpool
MATTHIAS THIMM, Professor, FernUniversität in Hagen
FRANCESCA TONI, Professor, Imperial College London

EXAMINATEURS:
LEILA AMGOUD, Research director, CNRS, Université Toulouse 3
BENOIT CRABBÉ, Professor, Université Paris Cité
NICOLAS MAUDET, Professor, Sorbonne Université

Soutenue publiquement le 12 juin 2024

Contents

1. Introduction

This habilitation thesis presents my research activities since I obtained my PhD in November 2007, and is based on several papers I published with my colleagues and contains some parts of them.

1.1 Context

I do my research in the *Distributed Artificial Intelligence* group at LIPADE in Université Paris Cité. This team is carrying out theoretical and applied research in the domain of intelligent agents and multi-agent systems. My work focuses on different types of interactions in multi-agent systems.

I do not see research as anything other than a collective effort, and I would like to take this opportunity to thank my co-authors:[1] Douae Ahmadoun (LIPADE[2], Paris), Stéphane Airiau (LAMSADE, Paris), Leila Amgoud (IRIT, Toulouse), Nicholas Asher (IRIT, Toulouse), Cédric Buron (Thales, Paris), Marco Correia (Universidade Nova de Lisboa, Portugal), Jorge Cruz (Universidade Nova de Lisboa, Portugal), Jérôme Delobelle (LIPADE, Paris), Caroline Devred (LERIA, ANgers), Yannis Dimopoulos (University of Cyprus), Dragan Doder (Utrecht University), Louise Dupuis de Tarlé (LAMSADE, Paris), Ulle Endriss (University of Amsterdam), Sébastien Konieczny (CRIL, Lens), Dionysios Kontarinis (LIPADE[3], Paris), Marie-Christine Lagasquie-Schiex (IRIT, Toulouse), Jérôme Lang (LAMSADE, Paris), Alex Lascarides (University of Edinburgh), João Leite (Universidade Nova de Lisboa, Portugal), Nicolas Maudet (LIP6, Paris), Alexis Martin (LIP6, Paris), Pavlos Moraitis (LIPADE, Paris), Stefano Moretti (LAMSADE, Paris), Alan Perotti (CentAI, Italy), Julien Rossit (LIPADE, Paris), Pierre Savéant (Thales, Paris), Onn Shehory (Bar-Ilan University, Israel), Leon van der Torre (University of Luxembourg), Srdjan Vesic (CRIL, Lens), Serena Villata (I3S, Sophia Antipolis) and Bruno Zanuttini (GREYC, Caen).

1.2 Research questions

In multi-agent systems, several agents interact, cooperatively or not, to achieve a given objective, that can be personal to each agent, or common to the group of agents. My main research interest consists in studying such interactions between agents.

These interactions can take different forms: they can be purely strategic when agents seek to satisfy their own interests, and reason about the other agents' strategy. In this context, game theory presents an interesting set of tools to model and study these interactions. Agents can also need to interact with each other to make decisions together or agree on a course of action. Here again, game theory, and more especially cooperative game theory, is a natural way to study these problems.

However, game theory is not suitable anymore when agents need to convince the other agents that

[1] In alphabetical order

[2] Now AI Research Engineer @ Safran Tech

[3] Now Data Scientist in Cartography & Geospatial Data Section at Hellenic Statistical Authority (ELSTAT)

their solution, or idea, is the best one. In this case, the mechanisms proposed in argumentation theory, that allow agents to present arguments to support their positions, are more suitable.

Another interest rose in the question of task allocation mechanisms. In this context, agents need to undertake some tasks to accomplish a common "mission". These missions typically involve different tasks each requiring different skills, tools or resources. In general, these tasks are complementary and interrelated in terms of shared resources, execution time or efficiency.

1.2.1 Argumentation Theory

Argumentation Theory provides a means of reasoning about the exchange and evaluation of interacting arguments. In his seminal work, Dung (1995) proposed an argument framework where the content and the structure of the arguments are abstract. It is then possible to focus on the reasoning process to decide about the status of any specific argument, or group of arguments.
To determine which arguments are acceptable, different argumentation semantics were defined. They can be divided into several categories: *extension-based* semantics, where sets of arguments are accepted together in so-called extensions (see Baroni et al. (2011) for an overview); *scoring semantics*, where a numerical acceptability degree is assigned to each argument (Besnard and Hunter, 2001; da Costa Pereira et al., 2011; Leite and Martins, 2011; Matt and Toni, 2008); and *ranking-based semantics*, where arguments are individually ranked from the most to the least acceptable (Amgoud and Ben-Naim, 2013; Amgoud et al., 2016; Bonzon et al., 2016b; Cayrol and Lagasquie-Schiex, 2005; Grossi and Modgil, 2015; Pu et al., 2014).
Ranking-based semantics address a different question than classical Dung's extension-based semantics. Indeed, they do not provide any indication as to what sets of arguments can be *jointly accepted*. For some applications though, like online debate platforms or decision-making where a large number of arguments is involved, ranking semantics provide highly valuable information, complementary to classical semantics which returns two levels of evaluations (accepted or rejected). In recent years, several ranking-based semantics have been introduced. We proposed two different methods of comparison of these existing ranking-based semantics (an experimental one, and an axiomatic one). Then, based on our axiomatic study, we introduced four new families of semantics: a first one based on the notion of propagation; a second one conceived to address the specificities of persuasion dialogues; a third one combining the strength of extension-based and ranked-based semantics; and finally a fourth one taking account the similarities between arguments.

1.2.2 Argumentative protocols

It is natural to justify your positions and opinions with arguments, so argumentation is at the heart of the deliberative process. A few years ago, most of the existing work on argumentation theory concerned an agent reasoning alone through an argumentation process, and based only on the information available to her. My research was therefore guided by the idea of defining the basis of a theory of multilateral argumentation, potentially involving many agents, and in which each agent has his own opinion (arguments, beliefs, objectives, etc.). An important application of this work is linked to the debate systems that are emerging on the Internet. The success of these platforms in their current form seems to suggest that they can become an important source of information. For example, Debategraph[4] has been used to produce maps for the British newspaper "The Independent" as well as talk shows on CNN, and is supported by the White House, the European Commission and other institutions. These debate systems are, however, still in their infancy. For the moment, these are mainly interfaces in which it is possible to give arguments for or against a given question, without processing or evaluating these arguments.
I thus have been interested in studying protocols for persuasion in such multi-agent systems, first in the context of extension-based semantics. Then, I turned my attention to the dynamical aspect of

[4] https://debategraph.org

gradual semantics and studied the impact of such semantics on the debates and the agents' opinions. I also had an interest in argumentative negotiation and looked at a protocol allowing agents to take into account a partial knowledge of their opponents. Another point of interest was to study how the diversity of views observed in such multi-agent systems is consistent with the views of every individual argumentation framework.

1.2.3 Interaction in Games

Game theory aims at developing mathematical tools to model strategic interactions between several agents. More specifically, Boolean games (that I studied during my PhD) make it possible to represent strategic games succinctly by taking advantage of the power of expression and the conciseness of propositional logic.
I was interested in studying the links between argumentation theory and game theory. More specifically, I wondered how the tools from game theory can be used in the setting of abstract argumentation theory. Many similarities exist between argumentation systems and CP-Boolean games, and we worked on a translation between both systems and then identified some links between the solutions concepts of both theories.
Another interesting question regarding the links between game and argumentation theory is to study how the concepts of cooperative game theory can guide agents as to what argument should be put forward in a debate.

1.2.4 Task Allocation

In today's era of rising technological applications such as robotics, autonomous engines and home devices, there has been a renewed interest in multi-agent task allocation technologies. The tasks at hand typically involve different sub-tasks each requiring different skills, tools or resources of the agents involved. In general, these tasks are complementary and interrelated in terms of shared resources, execution time or efficiency. As many communication disconnections or breakdowns may occur in real-world applications, we need multi-agent task allocation mechanisms in a decentralized and robust manner, to avoid the single-point failure possible in a centralized configuration.
I thus wondered: how to allocate interdependent tasks to agents and form coalitions in a decentralized manner? In particular, we focused our research on the case where agents cannot execute more than one task at a time and are heterogeneous, meaning they have different capabilities. These agents must realize, cooperatively as a team, a mission composed of complex tasks where each of those tasks requires a subset of agents with a specific combination of capabilities for its achievement.
Furthermore, the tasks are interdependent. The quality of a task does not depend only on the agents assigned to it but can also depend on other tasks. Regarding the mission execution environment, we started by considering a static environment, that we extended to a dynamic one where some agents can break down while others can join the mission on the road, and some tasks can fail while new ones can be added. For this reason, we focused on a decentralized configuration to adapt to the nature of the intended applications that can be critical with communication instability and where agents are robots or drones.

2. Argumentation

Argumentation is a reasoning model based on the evaluation, and sometimes exchange, of interacting arguments. The most popular way to represent the argumentation process was proposed by Dung (1995) with abstract argumentation frameworks, in which the structure of arguments is abstract. Such systems can then be modeled by directed graphs, where the nodes represent (abstract) arguments, and the edges represent the attacks between them.

Given an argumentation framework, we need a reasoning process to decide about the status of any specific argument, or group of arguments: answering this question corresponds to defining an argumentation semantics. Various semantics (see Baroni et al. (2011) for an overview) have been formulated to compute these sets of arguments, called extensions (or equivalently labelings Caminada (2006)), from an argumentation framework. An alternative way to evaluate arguments consists of directly reasoning on the arguments themselves by exploiting the topology of the argumentation framework. Following this idea, scoring semantics (assigning numerical acceptability degree to each argument) and ranking-based semantics (returning a ranking on the arguments) have been proposed in recent years. Such semantics address a different question than classical Dung's semantics. Indeed, they do not provide any indication as to what sets of arguments can be *jointly accepted.* For some applications though, like online debate platforms or decision-making where a large number of arguments is involved, ranking semantics provide highly valuable information, complementary to classical semantics which returns two levels of evaluations (accepted or rejected).

In this chapter, we are interested in studying ranking-based semantics: we will propose two different methods of comparison of existing ranking-based semantics (an experimental one, and an axiomatic one). Then, we will introduce four new families of semantics: a first one based on the notion of propagation; a second one conceived to address the specificities of persuasion dialogues; a third one combining the strength of extension-based and ranked-based semantics; and finally a fourth one taking account the similarities between arguments.

Note that the main part of this work has been done by Jérôme Delobelle (Delobelle, 2017) during his PhD thesis, co-supervised with Sébastien Konieczny and Nicolas Maudet.

We will start by giving some background on abstract argumentation theory. My objective in this section is not to give an exhaustive state of the art but to introduce some concepts that will be useful in the rest of this thesis.

2.1 Basic concepts on abstract argumentation

Abstract argumentation has received a great deal of interest since the work of Dung (1995). The idea behind this seminal work is to focus only on the definition of the status of arguments, and to abstract from the content and the structure of those arguments. Arguments are then simply considered abstract entities and can be represented as nodes of a graph.

Definition 2.1 (Argumentation framework (AF))
An **argumentation framework (AF)** is a pair $\langle \mathcal{A}, \mathcal{R} \rangle$ of a set $\mathcal{A}$ of arguments and a binary relation $\mathcal{R}$ on $\mathcal{A}$ called the **attack relation**. $\forall a_i, a_j \in \mathcal{A}$, $(a_i, a_j) \in \mathcal{R}$ means that a_i **attacks** a_j (or a_j is **attacked** by a_i). An AF may be represented by a directed graph, called the **interaction graph**, whose nodes are arguments and edges represent the attack relation.

In the following, $\mathbb{AF}$ will represent the set of all argumentation frameworks.

Definition 2.2 (Path, Defender, Attacker)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$, and $a, b \in \mathcal{A}$. A **path** P from b to a, noted $P(b,a)$, is a sequence $\langle a_0, \ldots, a_n \rangle$ of arguments such that $a_0 = a$, $a_n = b$ and $\forall i < n, (a_{i+1}, a_i) \in \mathcal{R}$. We denote by $l_P = n$ the **length** of P.
A **defender** (resp. **attacker**) of a is an argument situated at the beginning of an even-length (resp. odd-length) path. We denote the multiset of defenders and attackers of a by $\mathcal{R}_n^+(a) = \{b \mid \exists P(b,a) \text{ with } l_P \in 2\mathbb{N}\}$ and $\mathcal{R}_n^-(a) = \{b \mid \exists P(b,a) \text{ with } l_P \in 2\mathbb{N}+1\}$ respectively. The **direct attackers** of a are arguments in $\mathcal{R}_1^-(a)$. An argument a is **defended** if $\mathcal{R}_2^+(a) \neq \emptyset$.
A **defense root** (resp. **attack root**) is a non-attacked defender (resp. attacker). We denote the multiset of defense roots and attack roots of a by $\mathcal{B}_n^+(a) = \{b \in \mathcal{R}_n^+(a) \mid |\mathcal{R}_1^-(b)| = 0\}$ and $\mathcal{B}_n^-(a) = \{b \in \mathcal{R}_n^-(a) \mid |\mathcal{R}_1^-(b)| = 0\}$ respectively. A path from b to a is a **defense branch** (resp. **attack branch**) if b is a defense (resp. attack) root of a. Let us note $\mathcal{B}^+(a) = \bigcup_n \mathcal{B}_n^+(a)$ and $\mathcal{B}^-(a) = \bigcup_n \mathcal{B}_n^-(a)$.

The goal of the argumentation process is to evaluate these arguments, taking into account the existing conflicts between them, to determine their degree of acceptability.
In Dung's framework, the *acceptability of an argument* depends on its membership to some sets, called extensions. These extensions characterize collective acceptability. The main characteristic properties are:

Definition 2.3 (Conflict-free set, Acceptability)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework. Let $S \subseteq \mathcal{A}$.

- **Conflict-free set**: S is *conflict-free* for AF if and only if there exists no a_i, a_j in S such that $a_i \mathcal{R} a_j$.
- **Acceptability**: An argument a is *acceptable w.r.t. S* for AF if and only if $\forall b \in \mathcal{A}$ such that $(b,a) \in \mathcal{R}$, $\exists c \in S$ such that $(c,b) \in \mathcal{R}$. S is *acceptable* for AF if and only if $\forall a \in S$, a is acceptable w.r.t. S for AF.

Then several *semantics for acceptability* have been defined in Dung (1995). For instance:

Definition 2.4 (Extension-based semantics)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and $S \subseteq \mathcal{A}$.

- **Admissible**: S is an *admissible set* of AF if and only if S is conflict-free and acceptable for AF.
- **Preferred extension**: S is a *preferred extension* of AF if and only if S is maximal (w.r.t. $\subseteq$) among the admissible sets for AF.
- **Stable extension**: S is a *stable extension* of AF if and only if S is conflict-free and S attacks every argument in $\mathcal{A} \setminus S$.
- **Complete extension**: S is a *complete extension* of AF if and only if $\forall a \in \mathcal{A}$, if a is acceptable w.r.t. S, then $a \in S$
- **Grounded extension**: S is the *grounded extension* of $\langle \mathcal{A}, \mathcal{R} \rangle$ if and only if S is the minimal

(w.r.t. $\subseteq$) complete extension.

We denote by $\mathcal{E}_\sigma(\mathsf{AF})$ the set of extensions of AF for the semantics $\sigma \in \{$**co**(mplete), **pr**(eferred), **st**(able), **gr**(ounded)$\}$.

An alternative way to represent extension-based semantics is by using a labeling-based approach Caminada (2006). Indeed, rather than stating in terms of sets of arguments, a labeling function can be used to assign a label to each argument. The idea of labeling is to associate exactly one label to each argument, which can either be `in`, `out` or `undec`. The label `in` indicates that the argument is explicitly accepted, the label `out` indicates that the argument is explicitly rejected, and the label `undec` indicates that the status of the argument is undecided, meaning that one abstains from a judgment whether the argument is accepted or rejected.

Definition 2.5 (Labeling-based semantics)
Let $\Lambda = \{\texttt{in}, \texttt{out}, \texttt{undec}\}$. Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework. A labeling on AF is a total function $\mathcal{L} : \mathcal{A} \to \Lambda$. A **labeling-based semantics** in a function λ such that for every argumentation framework AF we have that $\lambda(\mathsf{AF})$ is a set of labelings on AF.

In the labeling-based approach, a semantics definition relies on some legality constraints relating the label of an argument to those of its attackers.

Definition 2.6 (Legal labeling)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, and $\mathcal{L}$ be a labeling of AF.

- An `in`-labeled argument is said to be **legally** `in` if and only if all its attackers are labeled `out`.
- An `out`-labeled argument is said to be **legally** `out` if and only if it has at least one attacker that is labeled `in`.
- An `undec`-labeled argument is said to be **legally** `undec` if and only if not all its attackers are labeled `out` and it does not have an attacker that is labeled `in`.

To simplify the technical treatment in the following, some extension-based semantics are defined by referring to the commitment relation between labelings (Baroni et al., 2011).

Definition 2.7 (Commitment relation between labelings)
Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be two labelings. We say that $\mathcal{L}_2$ is **more or equally committed** than $\mathcal{L}_1$, denoted $\mathcal{L}_1 \sqsubseteq \mathcal{L}_2$, if and only if $\texttt{in}(\mathcal{L}_1) \subseteq \texttt{in}(\mathcal{L}_2)$ and $\texttt{out}(\mathcal{L}_1) \subseteq \texttt{out}(\mathcal{L}_2)$.

On this basis, the labeling-based definitions of several argumentation semantics can be introduced.

Definition 2.8 (labeling-based definitions of argumentation semantics)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework.

- An **admissible labeling** of AF is a labeling $\mathcal{L}$ where every `in`-labeled argument is legally `in` and every `out`-labeled argument is legally `out`.
- A **complete labeling** of AF is a labeling where every `in`-labeled argument is legally `in`, every `out`-labeled argument is legally `out` and every `undec`-labeled argument is legally `undec`.
- A **stable labeling** of AF is a complete labeling without `undec`-labeled arguments.
- The **grounded labeling** of AF is the minimal (w.r.t. $\sqsubseteq$) labeling among all complete labelings.
- A **preferred labeling** of AF is a maximal (w.r.t. $\sqsubseteq$) labeling among all complete labelings.

We denote by $\mathcal{L}_\sigma(\mathsf{AF})$ the set of labelings of AF for the semantics $\sigma \in \{\text{co, pr, st, gr}\}$.

For an argumentation framework AF with at least one extension (resp. labeling), we say that an argument is *skeptically accepted* if it belongs to all of AF's extensions (resp. it is labeled `in` in all of AF's labelings). An argument is *credulously accepted* if it belongs to at least one of AF's extensions (resp. it is labeled `in` in at least one of AF's labelings). Given a semantics σ, we denote by $sa_\sigma(\mathsf{AF})$ (resp. $ca_\sigma(\mathsf{AF})$) the set of skeptically (resp. credulously) accepted arguments in AF.

A more fine-grained notion of a justification status has also been introduced in (Wu and Caminada, 2010) with a labeling-based justification status of the arguments in an argumentation framework. Concretely, the justification status of an argument consists of the set of labels that could reasonably be assigned to the argument w.r.t. the complete semantics.

Definition 2.9 (Justification status)
Let $\Lambda = \{\texttt{in}, \texttt{out}, \texttt{undec}\}$. Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and $x \in \mathcal{A}$. The justification status of x is the outcome yielded by the function $\mathcal{JS} : \mathcal{A} \to 2^\Lambda$ such that $\mathcal{JS}(x) = \{\mathcal{L}(x) \mid \mathcal{L} \in \mathcal{L}_{co}(\mathsf{AF})\}$.

For example, if an argument is labeled either `in` or `undec` in all the complete labelings then the justification status of this argument is $\{\texttt{in}, \texttt{undec}\}$. Thus, there are 6 possible statuses to be considered: $\{\texttt{in}\}$, $\{\texttt{out}\}$, $\{\texttt{undec}\}$, $\{\texttt{in}, \texttt{undec}\}$, $\{\texttt{out}, \texttt{undec}\}$ and $\{\texttt{in}, \texttt{out}, \texttt{undec}\}$.

Extension-based and labeling-based semantics (see Baroni et al. (2011) for an overview) aims at evaluating which sets of arguments can be accepted together. Thus, these semantics evaluate sets of arguments in a binary way (sets of arguments are or are not extensions (or `in`) for a given semantics).

However, this binary evaluation can be too rough for some applications, for example for online debate platforms (Leite and Martins, 2011), and the need for a more focused evaluation of each argument has been put forward. This led to the idea of ranking-based semantics (see e.g. Amgoud and Ben-Naim, 2013; Amgoud et al., 2016; Bonzon et al., 2016b; Cayrol and Lagasquie-Schiex, 2005; Grossi and Modgil, 2015; Patkos et al., 2016; Pu et al., 2015b) and scoring-based semantics (see e.g. Besnard and Hunter, 2001; da Costa Pereira et al., 2011; Leite and Martins, 2011; Matt and Toni, 2008), where the aim is to evaluate each argument in an argumentation system by exploiting the topology of the argumentation framework. The ranking-based semantics associates with any argumentation framework a ranking of the arguments from the most to the least acceptable ones. The scoring semantics assigns a numerical acceptability degree to each argument, taking into account various criteria from the argumentation framework. If one defines scoring-based semantics, then this straightforwardly induces corresponding ranking-based semantics. This allows having a large number of levels of acceptability and not only the classical accepted/rejected (or accepted/rejected) evaluations obtained with extension-based semantics, but on the other hand the joint acceptability of arguments is no longer captured.

Formally, ranking-based semantics rank-orders a set of arguments in an argumentation framework from the most acceptable to the weakest one(s) by comparing pairs of arguments.

Definition 2.10 (Ranking-based semantics)
A **ranking-based semantics** σ associates to any argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ a ranking $\succeq^\sigma_{\mathsf{AF}}$ on $\mathcal{A}$, where $\succeq^\sigma_{\mathsf{AF}}$ is a preorder (a reflexive and transitive relation) on $\mathcal{A}$.
$a \succeq^\sigma_{\mathsf{AF}} b$ means that a is at least as acceptable as b ($a \simeq^\sigma_{\mathsf{AF}} b$ is a shortcut for $a \succeq^\sigma_{\mathsf{AF}} b$ and $b \succeq^\sigma_{\mathsf{AF}} a$, and $a \succ^\sigma_{\mathsf{AF}} b$ is a shortcut for $a \succeq^\sigma_{\mathsf{AF}} b$ and $b \not\succeq^\sigma_{\mathsf{AF}} a$).

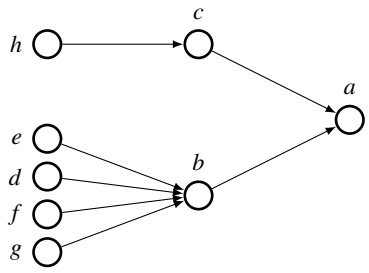
We denote $\sigma(\mathsf{AF})$ the ranking on $\mathcal{A}$ returned by σ. When there is no ambiguity about the argumentation framework in question, we will use $\succeq^\sigma$ instead of $\succeq^\sigma_{\mathsf{AF}}$.

2.2 Ranking semantics

Amgoud and Ben-Naim (2013) presents four considerations regarding extension-based semantics:

- *Killing*: The impact of an attack from an argument a to an argument b is drastic, that is, if b belongs to an extension, then a is automatically excluded from that extension.
- *Existence*: One successful attack against an argument a has the same effect as any number of successful attacks.
- *Absoluteness*: The three possible statuses of the arguments are absolute, that is, they make sense even without comparing them with each other.
- *Flatness*: All the accepted arguments have the same level of acceptability.

These considerations seem rational in applications like paraconsistent reasoning (Besnard and Hunter, 2008) where arguments are represented as formulas and attacks correspond to contradictions between these formulas. Here, the killing and existence considerations seem essential to capture the fact that one, and only one, attack is lethal and prevent any contradiction between arguments and thus obtain a consistent set of formulas.
However, in other applications like decision-making, online debate platforms or when additional information exists in the framework, some of these considerations are debatable.
For example, imagine an online debate represented by the following argumentation framework, where argument a is the subject of the debate:



In a classical setting, b and c have the same status: they both do not belong to any extension and are not accepted. However, it could make sense that the level of acceptability of argument b, which is widely attacked, should be lower than that of c, in contradiction with Existence and Absoluteness considerations.
On the other hand, arguments a, e, d, f, g and h are all accepted and have the same level of acceptability. But, in applications like decision-making, it is reasonable to consider that the level of acceptability of a should be lower than that of h for example, which is not attacked, in contradiction with Flatness consideration.

This shows that if classical semantics can be well-suited for reasoning, they are not suited for applications like decision-making.

2.2.1 Properties of ranking-based semantics

Many ranking-based semantics with different behaviors have been introduced in recent years (Amgoud and Ben-Naim, 2013; Amgoud et al., 2016; Bonzon et al., 2016b; Cayrol and Lagasquie-Schiex, 2005; Grossi and Modgil, 2015; Pu et al., 2014, 2015b). These semantics have often been defined along with a set of properties, each of which represents a specific criterion (e.g. quantity or quality of the direct attackers of an argument), aiming to highlight the unique behavior of the proposed semantics (Amgoud and Ben-Naim, 2013; Cayrol and Lagasquie-Schiex, 2005; Matt and Toni, 2008).
Note, however, that none of these properties are mandatory. Indeed, some of these properties are not independent of each other because incompatibilities and dependencies can exist between

them (Besnard et al., 2017; Bonzon et al., 2016a). These properties are therefore an excellent indication to better understand the behavior of these semantics and remain the only way to compare existing ranking-based semantics with new ones. They also are a first step to the ambitious research question of defining fully characterized classes of ranking-based semantics with respect to a subset of properties.
In this section, I recall the intuition of these properties and let the interested reader look for the original works for details.

First, it seems natural that the ranking on the set of abstract arguments should be defined only based on the attacks between arguments and should not depend on the identity of the arguments.

Property 1 — Abstraction (Abs). (Amgoud and Ben-Naim, 2013)
The ranking on $\mathcal{A}$ should be defined only based on the attacks between arguments.

Property 2 — Independence (In). (Amgoud and Ben-Naim, 2013; Matt and Toni, 2008)
The ranking between two arguments a and b should be independent of any argument that is neither connected to a nor b.

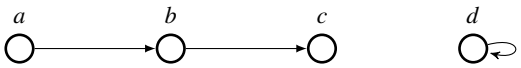
Example 2.1 Consider the two following argumentation frameworks:

The property Abstraction ensures that the ranking between a and b is the same as the one between c and d.
The property Independence ensures that the ranking between a and b (and the one between c and d) remains the same after the fusion of the two frameworks. ∎

Property 3 — Void Precedence (VP). (Amgoud and Ben-Naim, 2013; Cayrol and Lagasquie-Schiex, 2005; Matt and Toni, 2008)
A non-attacked argument is ranked strictly higher than any attacked argument.

Property 4 — Self-Contradiction (SC). (Matt and Toni, 2008)
A self-attacking argument is ranked lower than any non self-attacking argument.

Example 2.2 Consider the following argumentation framework:

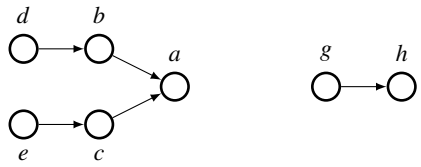


The property Void Precedence ensures that a, which is not attacked, is strictly more acceptable than both b and c which are attacked.
The property Self-Contradiction ensures that d which attacks itself is strictly less acceptable than a, b and c. ∎

Property 5 — Cardinality Precedence (CP). (Amgoud and Ben-Naim, 2013)
The greater the number of direct attackers for an argument, the weaker the level of acceptability of this argument.

Property 6 — Quality Precedence (QP). (Amgoud and Ben-Naim, 2013)
The greater the acceptability of one direct attacker for an argument, the weaker the level of acceptability of this argument.

Example 2.3 Consider the following argumentation framework:



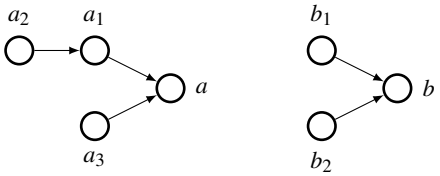
The property Cardinality Precedence ensures that h is strictly more acceptable than a, because a has more direct attackers.
If we suppose that g is strictly more acceptable than b and c, then the property Quality Precedence ensures that a is strictly more acceptable than h. ■

Property 7 — Defense Precedence (DP). (Amgoud and Ben-Naim, 2013)
For two arguments with the same number of direct attackers, a defended argument is ranked higher than a non-defended argument.

Property 8 — Counter-Transitivity (CT). (Amgoud and Ben-Naim, 2013)
If the direct attackers of b are at least as numerous and acceptable as those of a, then a should be at least as acceptable as b.

Property 9 — Strict Counter-Transitivity (SCT). (Amgoud and Ben-Naim, 2013)
If CT is satisfied and either the direct attackers of b are strictly more numerous or acceptable than those of a, then a is strictly more acceptable than b.

Example 2.4 Consider the two following argumentation frameworks:



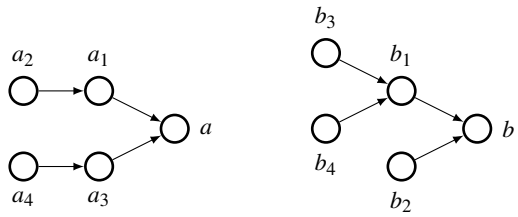
Both arguments a and b have two attackers, but a is defended by a_2 whereas b is not defended. Defense Precedence ensures that a is ranked higher than b.
As b has two strong attackers (they are not attacked), whereas a has one strong (a_3) and one weak (a_1, attacked by a_2) attackers, (Strict) Counter-Transitivity ensures that a is (strictly) more acceptable than b.

■

Definition 2.11 (Simple and distributed defense)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ and $a \in \mathcal{A}$. The **defense** of a is **simple** if and only if every defender of a attacks exactly one direct attacker of a. The defense of a is **distributed** if and only if every direct attacker of a is attacked by at most one argument.

Property 10 — Distributed-Defense Precedence (DDP). (Amgoud and Ben-Naim, 2013)
The best defense is when each defender attacks a distinct attacker to weaken all of them, instead of focusing on a specific direct attacker to greatly weaken it (but at the price of leaving its other attackers unaffected).

Example 2.5 Consider the two following argumentation frameworks:



The two arguments a and b have the same number of attackers and the same number of defenders. The defense of a is simple because a_2 directly attacks a_1 and a_4 directly attacks a_3; distributed because a_1 and a_3 are attacked by exactly one argument. Conversely, the defense of b is also simple but not distributed (b_1 is directly attacked by two arguments). Distributed-Defense Precedence ensures that a is more acceptable than b.

■

The following properties check if some change in an AF can improve or degrade the ranking of one argument. These properties have been proposed informally by Cayrol and Lagasquie-Schiex (2005), in the context of their semantics. In Bonzon et al. (2016a), we proposed a formalization that generalizes them for any argumentation framework.

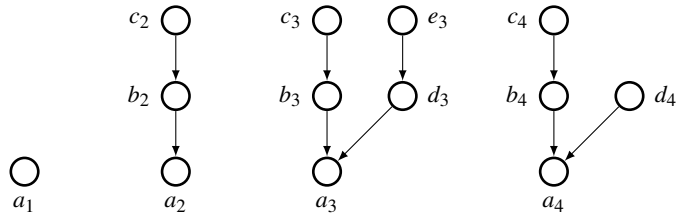
Property 11 — Strict addition of Defense Branch ($\oplus$DB).
Adding a defense branch to any argument improves its ranking.

It makes sense to treat differently non-attacked arguments. So in Cayrol and Lagasquie-Schiex (2005), this property is defined in a more specific way:

Property 12 — Addition of Defense Branch (+DB).
Adding a defense branch to any attacked argument improves its ranking.

Property 13 — Addition of Attack Branch (+AB).
Adding an attack branch to any argument degrades its ranking.

Example 2.6 Consider the following argumentation framework:



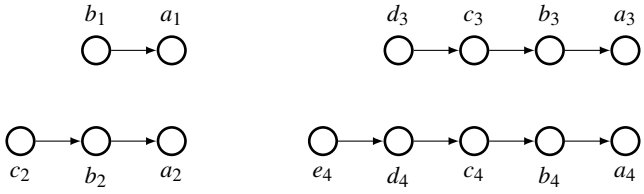
Addition of Attack Branch (+AB) ensures that a_1, which has no attack branch, is more acceptable than b_2, b_3, d_3 and b_4 which have one attack branch; and that a_2 is more acceptable than a_4 (both have one defense branch with the same length but a_4 has also an attack branch while a_2 has not).
Strict addition of Defense Branch ($\oplus$DB) ensures that a_3 is more acceptable than a_2, which is more acceptable than a_1. Indeed, a_3 has one more defense branch than a_2, which has one more defense branch than a_1. In addition, a_4 with one defense branch and one attack branch should be more acceptable than b_2, b_3, d_3 and b_4 which have no defense branch.
Addition of Defense Branch (+DB) ensures allows the same conclusion as $\oplus$DB, except for a_1. Indeed, as a_1 is not attacked, nothing can be said about its ranking in comparison with the other arguments.

■

Property 14 — Increase of Attack branch (↑AB).
Increasing the length of an attack branch of an argument improves its ranking.

Property 15 — Increase of Defense branch (↑DB).
Increasing the length of a defense branch of an argument degrades its ranking.

Example 2.7 Consider the following argumentation framework:



Increase of Attack branch (↑AB) ensures that a_3, which has an attack branch of length 3, is more acceptable than a_1, which has one attack branch of length 1.
Increase of Defense branch (↑DB) ensures that a_2, which has a defense branch of length 2, is more acceptable than a_4, which has a defense branch of length 4. ∎

To this set of properties from the literature, we added some other important properties in (Bonzon et al., 2016a; Delobelle, 2017).
The first one, *Total*, allows to make a distinction between the semantics which return a total preorder or a partial preorder between arguments. Indeed, too many incompatibilities can be problematic, especially if we want to use argumentation for decision-making or online debate platforms. *Argument Equivalence* ensures that the acceptability of an argument depends only on (the structure of) its attackers and defenders. This property is related to a well-known property satisfied by the classical semantics, called Directionality (Baroni et al., 2011), which states that an argument can only be affected by arguments following the direction of the attacks (i.e. an argument a cannot be affected by another argument b if there exists no path from b to a). *Non-attacked Equivalence* is a particular case of Argument Equivalence that focuses on the comparison between the non-attacked arguments: if the arguments are affected only by the arguments in their ancestors' graph, then the non-attacked arguments should be unaffected by the remaining part of the argumentation framework and should have the same ranking. Another possibility to detect when two arguments are equally acceptable consists of taking into account their direct attackers, which is captured by *Ordinal Equivalence*. Finally, *Attack vs Full Defense* describes the behavior adopted by semantics concerning the notion of defense and can be viewed as some kind of compatibility with usual Dung's semantics. The idea is that a defended argument is always better than an attacked argument.

Property 16 — Total (Tot). All pairs of arguments can be compared.

Property 17 — Argument Equivalence (AE).
If two arguments have the same ancestors' graph, then they are equally acceptable.

Property 18 — Non-attacked Equivalence (NaE).
All the non-attacked argument have the same rank.

Property 19 — Ordinal Equivalence (OE).
If two arguments a and b have the same number of direct attackers and that, for each direct attacker of a, there exists a direct attacker of b such that the two attackers are equally acceptable, then a and b are equally acceptable too.

Property 20 — Attack vs Full Defense (AvsFD).
An argument without any attack branch is ranked higher than an argument only attacked by one

non-attacked argument.

2.2.2 Links between these properties

Each of these properties aims to capture a particular principle. These different principles may be contradictory, or they may overlap. (Amgoud and Ben-Naim, 2013; Besnard et al., 2017) highlighted some incompatibilities and dependencies between properties, and we pursued this work in (Bonzon et al., 2016a, 2023; Delobelle, 2017).

Definition 2.12 (Incompatibility)
Two properties are **incompatible** if there exists an argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ and $a, b \in \mathcal{A}$ such that when one property states that $a \succ_{AF} b$, the other property states that $b \succeq_{AF} a$.

Proposition 2.1 For every ranking-based semantics, the following pairs of properties are incompatible:

1. Cardinality Precedence (CP) and Quality Precedence (QP) (Amgoud and Ben-Naim, 2013)
2. Self-Contradiction (SC) and Cardinality Precedence (CP) (Besnard et al., 2017)
3. Self-Contradiction (SC) and Counter-Transitivity (CT) (Besnard et al., 2017)
4. Self-Contradiction (SC) and Strict Counter-Transitivity (SCT) (Besnard et al., 2017)
5. Cardinality Precedence (CP) and Attack vs Full Defense (AvsFD)
6. Cardinality Precedence (CP) and Addition of a Defense Branch (+DB)
7. Cardinality Precedence (CP) and Strict Addition of a Defense Branch ($\oplus$DB)
8. Void Precedence (VP) and Strict Addition of a Defense Branch ($\oplus$DB)
9. Argument Equivalence (AE) and Self-Contradiction (SC)

Moreover, no ranking-based semantics can simultaneously satisfy Addition of a Defense Branch (+DB), Strict Counter-Transitivity (SCT) and Argumentation Equivalence (AE).

Some of these results are not surprising. Indeed, some properties have different views on the notion of defense. It is the case, for example, with the properties CP and SCT which consider that any additional (defense or attack) branch should harm a given argument while +DB or $\oplus$DB state that an additional defense branch should have a positive impact on this argument.

Definition 2.13 (Implication)
A property P **implies** an other property Q if and only if for any ranking-based semantics σ, if σ satisfies P then σ satisfies Q.

Proposition 2.2 For every ranking-based semantics, the following pairs of properties are not independent:

1. Strict Counter-Transitivity (SCT) implies Void Precedence (VP) (Amgoud and Ben-Naim, 2013)
2. Counter-Transitivity (CT) and Strict Counter-Transitivity (SCT) imply Defense Precedence (DP) (Amgoud and Ben-Naim, 2013)
3. Counter Transitivity (CT) implies Non-attacked Equivalence (NaE)
4. Counter Transitivity (CT) implies Ordinal Equivalence (OE)
5. Strict Counter-Transitivity (SCT) and Ordinal Equivalence (OE) imply Counter-Transitivity (CT)

6. Strict Addition of Defense Branch ($\oplus$DB) implies Addition of a Defense Branch (+DB)
7. Argument Equivalence (AE) implies Non-attacked Equivalence (NaE)
8. Ordinal Equivalence (OE) implies Non-attacked Equivalence (NaE)
9. Void Precedence (VP) and Quality Precedence (QP) imply Attack vs Full Defense (AvsFD)
10. Cardinality Precedence (CP) implies Addition of an Attack Branch (+AB)

Interestingly, even if each property aims to catch a particular behavior, some of them remain connected. For example, if the properties SCT and OE are both satisfied, then one can directly consider VP, DP, CT and NaE satisfied too.
All the results obtained in this section are summed up in Figure 2.1.

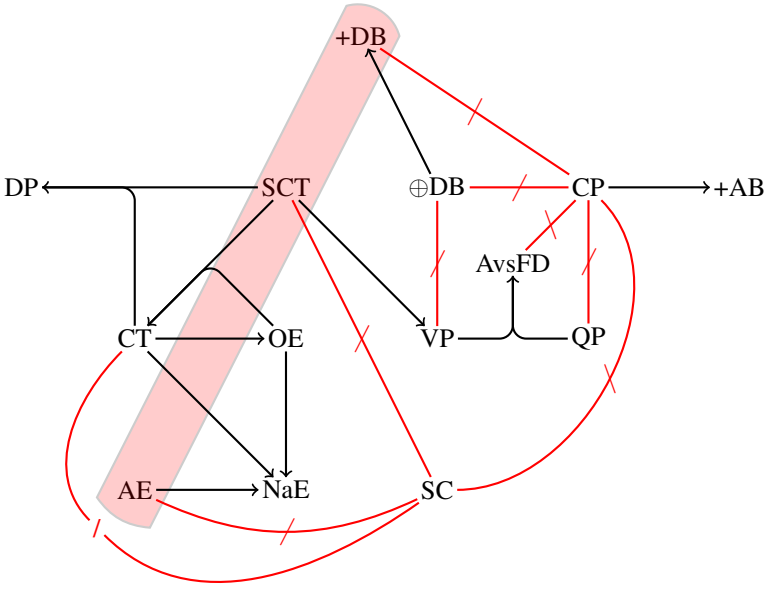


Figure 2.1: Graph which represents the relation between properties ($X \rightarrow Y$ means that X implies Y, $X \not\rightarrow Y$ means that X and Y are not compatible and the properties into the red rectangle cannot be simultaneously satisfied.)

2.3 Argumentation Ranking Semantics based on Propagation

While many principles presented in Section 2.2.1 remain discussed and sometimes controversial, there is a common consensus over the fact that non-attacked arguments should have the highest rank. Indeed, the non-attacked arguments play a key role in the classical semantics (extension-based semantics) to select the accepted arguments. It could then be interesting to observe what is happening when more importance is given to them in ranking-based semantics while preserving the principles concerning the quality and the number of attackers or defenders. However, the impact these arguments should have on the other arguments is not always clear: should the arguments directly attacked by them be less acceptable than the other arguments? Should the impact of a non-attacked argument be the same as an attacked argument? A median approach that allows control of their impact in the argumentation framework could be interesting to define. For this purpose, we introduce three ranking-based semantics based on the propagation principle for which the influence of non-attacked arguments is more or less important.[1]

[1] An interested reader can find more details in (Bonzon et al., 2016b).

The propagation method that we define for our ranking-based semantics has two steps:

1. During the first step, a positive initial weight is assigned to each argument. The score of 1 attached to non-attacked arguments is set to be higher (or equal) than the score of attacked arguments, which is an ε between 0 and 1. The value of this ε is chosen accordingly to the degree of influence of the non-attacked arguments that we want: the smaller the value of ε is, the more important the influence of non-attacked arguments on the order prevails. But it is also possible to assign the same initial weight to all the arguments in the argumentation framework if one considers that all the arguments should have the same influence.
2. Then, during the second step, each argument propagates step by step its value into the argumentation framework in changing their polarities to comply with the attack relation meaning (attack or defense). For each argument, we accumulate and store the weights from its attackers and defenders in the argumentation framework.

Before formally defining the propagation principle, we want to pay close attention to a particular case concerning the selection of attackers and defenders of an argument. It could happen that an argument attacks or defends another argument through several paths with the same length. For example, on the following argumentation framework, two paths of length 2 exist from e to g: $\langle e,d,g\rangle$ and $\langle e,h,g\rangle$.

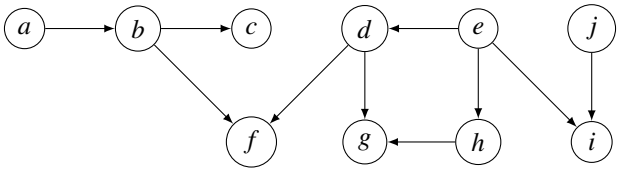


Figure 2.2: The argumentation framework AF_c

So there are two possibilities for e to propagate its value to g (or equally g receives the value from e): either g receives one value from e because it is its only defender or g receives two values from e because there exist two paths between both arguments. We consider that none of these options is better than the other, which is why we include a new parameter $\oplus$ aiming to select the set (S) of arguments at the end of the path without taking into account the number of possible paths, or the multiset (M) which encodes the fact that there are several possible paths.

Definition 2.14 (Attacker, Defender according to $\oplus$)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R}\rangle$ be an argumentation framework and $x, y \in \mathcal{A}$ be two arguments. Let $\oplus \in \{M, S\}$, where M (respectively S) stands for multiset (respectively set). The (multi)set of arguments such that there exists a path to x with a length of n is denoted by $\mathcal{R}_n^{\oplus}(x) = \{y \mid \exists P(y,x) \text{ with } l_P = n\}$.

To return to AF_c (Figure 2.2), the result is $\mathcal{R}_2^S(g) = \{e\}$ if we consider the set, whereas the result is $\mathcal{R}_2^M(g) = \{e,e\}$ with the multiset.
The propagation vector of an argument contains all the values received by its attackers and defenders.

Definition 2.15 (Propagation vector)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R}\rangle$ be an argumentation framework and $\oplus \in \{M, S\}$. The valuation $\mathcal{V}$ of $x \in \mathcal{A}$, at step i, is given by:

$$\mathcal{V}_i^{\varepsilon,\oplus}(x) = \begin{cases} v_\varepsilon(x) & \text{if } i = 0 \\ \mathcal{V}_{i-1}^{\varepsilon,\oplus}(x) + (-1)^i \sum\limits_{y \in \mathcal{R}_i^{\oplus}(x)} v_\varepsilon(y) & \text{otherwise} \end{cases}$$

where $v_\varepsilon : \mathcal{A} \to \mathbb{R}^+$ is a valuation function giving an initial weight to each argument, with $\varepsilon \in [0,1]$ such that $\forall y \in \mathcal{A}$,

$$v_\varepsilon(y) = \begin{cases} 1 & \text{if } \mathcal{R}_1^{\oplus}(y) = \emptyset \\ \varepsilon & \text{otherwise} \end{cases}$$

The **propagation vector** of x is denoted $\mathcal{V}^{\varepsilon,\oplus}(x) = \langle \mathcal{V}_0^{\varepsilon,\oplus}(x), \mathcal{V}_1^{\varepsilon,\oplus}(x), \dots \rangle$.

The first step of the propagation principle is ensured by the valuation function v_ε where 1 is assigned to the non-attacked arguments and ε for the attacked arguments. The propagation is then defined step by step: at step i, we add or remove (according to the value of $(-1)^i$) the accumulated score of x until the previous step ($\mathcal{V}_{i-1}^{\varepsilon,\oplus}(x)$) and the initial score (v_ε) received from arguments at the beginning of a path with a length of i ($\mathcal{R}_i^{\oplus}$).

Example 2.8 Let us compute the valuations $\mathcal{V}$ with $\varepsilon = 0.75$ for each argument in AF_c (Figure 2.2 on the facing page). These results are given in the following table. If no distinction exists between the set and multiset then the value is put in the same cell. Otherwise, the cell is divided into two parts (valuation for set at left and for multiset at right).

$\mathcal{V}_i^{0.75,\oplus}$	a,e,j		b,d,h		c		f		g		i	
	S	M	S	M	S	M	S	M	S	M	S	M
0	1		0.75		0.75		0.75		0.75		0.75	
1	1		-0.25		0		-0.75		-0.75		-1.25	
2	1		-0.25		1		1.25		0.25	1.25	-1.25	

Let's focus on argument f.

- Step $i = 0$: As it is attacked, its initial weight is 0.75: $\mathcal{V}_0^{0.75,\oplus}(f) = 0.75$
- Step $i = 1$: The direct attackers (b and d which are also attacked) propagate negatively their weights of 0.75 to f, so $\mathcal{V}_1^{0.75,\oplus}(f) = \mathcal{V}_0^{0.75,\oplus}(f) - (v_{0.75}(d) + v_{0.75}(b)) = -0.75$
- Step $i = 2$: f receives positively the weights of 1 from a and e which are non-attacked, so $\mathcal{V}_2^{0.75,\oplus}(f) = \mathcal{V}_1^{0.75,\oplus}(f) + (v_{0.75}(a) + v_{0.75}(e)) = 1.25$

As there exists no path to f with a length higher than 2, this score remains the same, and $\mathcal{V}^{0.75,\oplus}(f) = \langle 0.75, -0.75, 1.25 \rangle$.

■

It is important to note that $\mathcal{V}^{\varepsilon,\oplus}(x)$ may be infinite (this may occur when an argument is involved in at least one cycle). Moreover, the valuation $\mathcal{V}_n^{\varepsilon,\oplus}(x)$ of an argument x is not even necessarily bounded as $n \to \infty$. After a finite number of steps though, an argument is bound to receive the influence of exactly the same arguments as in a previous step of the vector (which means that the vector can be finitely encoded). More precisely, this number of steps is in the order of the least common multiplier of the cycle lengths occurring in the argumentation graph. As ranking-based semantics are not concerned with the exact values of arguments, but only with their relative ordering, this is sufficient for our purpose.

Once the propagation vector is computed for each argument in the argumentation framework, the goal is now to compare these vectors to provide a ranking between all the arguments. For this purpose, we introduce three ranking-based semantics (more exactly six ranking-based semantics since we have the set/multiset version of each semantics which can return different rankings between arguments for the same argumentation framework) giving more and more importance to non-attacked arguments: Propa_ε, $\text{Propa}_{1+\varepsilon}$ and $\text{Propa}_{1\to\varepsilon}$.

2.3.1 Propa$_\varepsilon$

Once the propagation vector is calculated for each argument in the argumentation framework, we can compare the different vectors to obtain an order between all the arguments. We want the influence of arguments to quickly decrease with the length of a path, so an option is to use a lexicographical comparison to compare these vectors.

Definition 2.16 (Lexicographical order)
A **lexicographical order** between two vectors of real numbers $V = \langle V_1, \ldots, V_n \rangle$ and $V' = \langle V'_1, \ldots, V'_n \rangle$ is defined as $V \succ_{lex} V'$ iff $\exists i \leq n$ s.t. $V_i > V'_i$ and $\forall j < i, V_j = V'_j$. $V \simeq_{lex} V'$ means that $V \not\succ_{lex} V'$ and $V' \not\succ_{lex} V$; and $V \succeq_{lex} V'$ means that $V' \not\succ_{lex} V$.

For the first semantics Propa$_\varepsilon$ we just compare the propagation vectors for a given ε.

Definition 2.17 (Propa$_\varepsilon$)
Let $\oplus \in \{M, S\}$ and $\varepsilon \in]0,1]$. The ranking-based semantics $\text{Propa}_\varepsilon^{\varepsilon,\oplus}$ associates to any argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ a ranking $\succeq^{\mathcal{V}}_{\mathsf{AF}}$ on $\mathcal{A}$ such that $\forall x, y \in \mathcal{A}$,

$$x \succeq^{\mathcal{V}}_{\mathsf{AF}} y \text{ if and only if } \mathcal{V}^{\varepsilon,\oplus}(x) \succeq_{lex} \mathcal{V}^{\varepsilon,\oplus}(y)$$

Example 2.8 **(continuing from p. 21)** Let us compute the ranking returned by $\text{Propa}_\varepsilon^{0.75,S}$ step by step, and lexicographically compare the propagation vectors of each argument in AF_c.

$\mathcal{V}_i^{0.75,S}$	a,e,j	b,d,h	c	f	g	i	
0	1	0.75	0.75	0.75	0.75	0.75	$a \simeq b \simeq c \simeq d \simeq e \simeq f \simeq g \simeq h \simeq i \simeq j$
1	1	-0.25	0	-0.75	-0.75	-1.25	$a \simeq e \simeq j \succ b \simeq c \simeq d \simeq f \simeq g \simeq h \simeq i$
2	1	-0.25	1	1.25	0.25	-1.25	$a \simeq e \simeq j \succ c \succ b \simeq d \simeq h \succ f \succ g \succ i$
							$a \simeq^{\mathcal{V}} e \simeq^{\mathcal{V}} j \succ^{\mathcal{V}} c \succ^{\mathcal{V}} b \simeq^{\mathcal{V}} d \simeq^{\mathcal{V}} h \succ^{\mathcal{V}} f \succ^{\mathcal{V}} g \succ^{\mathcal{V}} i$

So, the ranking returned by $\text{Propa}_\varepsilon^{0.75,S}$ is:

$$a \simeq^{\mathcal{V}} e \simeq^{\mathcal{V}} j \succ^{\mathcal{V}} c \succ^{\mathcal{V}} b \simeq^{\mathcal{V}} d \simeq^{\mathcal{V}} h \succ^{\mathcal{V}} f \succ^{\mathcal{V}} g \succ^{\mathcal{V}} i$$

■

If these semantics focus mainly on the attackers and defenders, they also take into account the fact that if there exist non-attacked arguments among them, these will be more influential than attacked arguments. However, this influence depends also on the value of ε. Indeed, two values of ε can lead to different orders. On Example 2.8, with $\varepsilon = 0.75$, if we focus on f, which is defended twice, and h, which is attacked (and not defended), we can see that h is better than f. But if we take $\varepsilon < 0.5$, we obtain the opposite case.
So, with Propa$_\varepsilon$ semantics, an argument with only (but numerous) defense branches can be worse than an argument only attacked by one non-attacked argument. A possible point of view is to focus only on the attackers in saying that the more an argument is directly attacked, the less acceptable the argument is. It is the case, for instance, with the semantics proposed by Amgoud and Ben-Naim (2013). But other approaches are possible, as we shall see now.

2.3.2 Propa$_{1+\varepsilon}$

If we do not want the influence of non-attacked arguments to be drowned in by the influence of attacked arguments, we have to split and lexicographically compare the influence of the two kinds of arguments. To do so, we need to define the shuffle operation.

Definition 2.18 (Shuffle)
The **shuffle** $\cup_s$ between two vectors of real numbers $V = \langle V_1, \ldots, V_n \rangle$ and $V' = \langle V'_1, \ldots, V'_n \rangle$ is defined as $V \cup_s V' = \langle V_1, V'_1, V_2, V'_2, \ldots, V_n, V'_n \rangle$.

Definition 2.19 (Propa$_{1+\varepsilon}$)
Let $\oplus \in \{M, S\}$ and $\varepsilon \in]0,1]$. The ranking-based semantics $\text{Propa}^{\varepsilon,\oplus}_{1+\varepsilon}$ associates to any argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ a ranking $\succeq^{\hat{\nu}}_{\mathsf{AF}}$ on $\mathcal{A}$ such that $\forall x, y \in \mathcal{A}$,

$$x \succeq^{\hat{\nu}}_{\mathsf{AF}} y \text{ if and only if } \mathcal{V}^{0,\oplus}(x) \cup_s \mathcal{V}^{\varepsilon,\oplus}(x) \succeq_{lex} \mathcal{V}^{0,\oplus}(y) \cup_s \mathcal{V}^{\varepsilon,\oplus}(y)$$

With these semantics, we simultaneously look at the result of the two propagation vectors $\mathcal{V}^{0,\oplus}$ and $\mathcal{V}^{\varepsilon,\oplus}$ step by step, using the shuffle operation, starting with the first value of the propagation vector $\mathcal{V}^{0,\oplus}$ (i.e. the one that takes into account non-attacked arguments only). In the case where two arguments are still equally acceptable, we compare the first value of the propagation vector $\mathcal{V}^{\varepsilon,\oplus}$. Then, in the case of equality, we move to the second step and so on.

Example 2.8 (continuing from p. 21) Let us first compute the propagation number of each argument in AF_c when $\varepsilon = 0$ and when $\varepsilon = 0.75$.

$\mathcal{V}^{0,\oplus}_i$	a,e,j		b,d,h		c		f		g		i	
	S	M	S	M	S	M	S	M	S	M	S	M
0	1		0		0		0		0		0	
1	1		-1		0		0		0		-2	
2	1		-1		1		2		1	2	-2	

$\mathcal{V}^{0.75,\oplus}_i$	a,e,j		b,d,h		c		f		g		i	
	S	M	S	M	S	M	S	M	S	M	S	M
0	1		0.75		0.75		0.75		0.75		0.75	
1	1		-0.25		0		-0.75		-0.75		-1.25	
2	1		-0.25		1		1.25		0.25	1.25	-1.25	

Let us focus on the argument f. Its propagation vector when $\varepsilon = 0$ is $\mathcal{V}^{0,\oplus}(f) = \langle 0,0,2 \rangle$ and $\mathcal{V}^{0.75,\oplus}(f) = \langle 0.75, -0.75, 1.25 \rangle$ when $\varepsilon = 0.75$. Let us now use the shuffle $\cup_s$ to combine these two propagation vectors:

$$\mathcal{V}^{0,\oplus}(f) \cup_s \mathcal{V}^{0.75,\oplus}(f) = \langle 0,0,2 \rangle \cup_s \langle 0.75, -0.75, 1.25 \rangle = \langle 0, 0.75, 0, -0.75, 2, 1.25 \rangle$$

Of course, the same method is used for all the others arguments. Comparing them with the lexicographical order gives the following ranking, for $\oplus = S$:

$$a \simeq^{\hat{\nu}} e \simeq^{\hat{\nu}} j \succ^{\hat{\nu}} c \succ^{\hat{\nu}} f \succ^{\hat{\nu}} g \succ^{\hat{\nu}} b \simeq^{\hat{\nu}} d \simeq^{\hat{\nu}} h \succ^{\hat{\nu}} i$$

■

It is also important to notice that Propa$_{1+\varepsilon}$, conversely to Propa$_\varepsilon$, returns the same ranking whatever the value of ε, which removes the problem of choosing "a good" ε.

Proposition 2.3 Let $\oplus \in \{M, S\}$. For any argumentation framework AF, for any $\varepsilon, \varepsilon' \in]0,1]$, it holds that

$$\text{Propa}^{\varepsilon,\oplus}_{1+\varepsilon}(\mathsf{AF}) = \text{Propa}^{\varepsilon',\oplus}_{1+\varepsilon}(\mathsf{AF})$$

2.3.3 Propa$_{1\to\varepsilon}$

A last possibility is to give a higher priority to the non-attacked arguments, by propagating only their weights in the graph. In other words, the acceptability of an argument depends only on its attack and defense roots. But if two arguments are still equivalent for this comparison (*i.e.* they have the same number of roots at each step), they are compared using the Propa$_\varepsilon$ method. Technically, the priority to the non-attacked arguments is given by using $\varepsilon = 0$. So, we first compare the propagation vectors $\mathcal{V}^{0,\oplus}$. And if the two propagation vectors are identical, we restart with a non-zero ε and compare the propagation vectors $\mathcal{V}^{\varepsilon,\oplus}$.

Definition 2.20 (Propa$_{1\to\varepsilon}$)
Let $\oplus \in \{M,S\}$ and $\varepsilon \in]0,1]$. The ranking-based semantics $\text{Propa}^{\varepsilon,\oplus}_{1\to\varepsilon}$ associates to any argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R}\rangle$ a ranking $\succeq^{\overline{\nu}}_{\mathsf{AF}}$ on $\mathcal{A}$ such that $\forall x, y \in \mathcal{A}$,

$x \succeq^{\overline{\nu}}_{\mathsf{AF}} y$ if and only if $\mathcal{V}^{0,\oplus}(x) \succ_{lex} \mathcal{V}^{0,\oplus}(y)$ or $(\mathcal{V}^{0,\oplus}(x) \simeq_{lex} \mathcal{V}^{0,\oplus}(y)$ and $\mathcal{V}^{\varepsilon,\oplus}(x) \succeq_{lex} \mathcal{V}^{\varepsilon,\oplus}(y))$

Example 2.8 **(continuing from p. 21)** Let us compute the ranking returned by $\text{Propa}^{0.75,S}_{1\to\varepsilon}$, step by step, for AF_c beginning by the case $\varepsilon = 0$ and pursuing with $\varepsilon = 0.75$.

$\mathcal{V}^{0,S}_i$	a,e,j	b,d,h	c	f	g	i	
0	1	0	0	0	0	0	$a \simeq b \simeq c \simeq d \simeq e \simeq f \simeq g \simeq h \simeq i \simeq j$
1	1	-1	0	0	0	-2	$a \simeq e \simeq j \succ b \simeq c \simeq d \simeq f \simeq g \simeq h \simeq i$
2	1	-1	1	2	1	-2	$a \simeq e \simeq j \succ f \succ c \simeq g \succ b \simeq d \simeq h \succ i$

$\mathcal{V}^{0.75,S}_i$	a,e,j	b,d,h	c	f	g	i	
0	1	0.75	0.75	0.75	0.75	0.75	$a \simeq e \simeq j \succ f \succ c \simeq g \succ b \simeq d \simeq h \succ i$
1	1	-0.25	0	-0.75	-0.75	-1.25	$a \simeq e \simeq j \succ f \succ c \succ g \succ b \simeq d \simeq h \succ i$
2	1	-0.25	1	1.25	0.25	-1.25	$a \simeq^{\overline{\nu}} e \simeq^{\overline{\nu}} j \succ^{\overline{\nu}} f \succ^{\overline{\nu}} c \succ^{\overline{\nu}} g \succ^{\overline{\nu}} b \simeq^{\overline{\nu}} d \simeq^{\overline{\nu}} h \succ^{\overline{\nu}} i$

One can remark that f has one more defense branch than c and g (when $\oplus = S$), which have also one more defense branch than b, d and h. This difference has a direct impact on the ranking between these arguments because one can see that at the end of the step $i = 2$ (when $\varepsilon = 0$), f is strictly more acceptable than g and c which are strictly more acceptable than b, d and h.
As some arguments are still equally acceptable (in particular c and g which have only one defense root), we restart the process with $\varepsilon = 0.75$. Thanks to this second process, c, which is directly attacked only once, is now strictly more acceptable than g which is directly attacked twice ($\mathcal{V}^{0.75,S}_1(c) = 0 > -0.75 = \mathcal{V}^{0.75,S}_1(g)$). So we obtain the following ranking for $\oplus = S$:

$$a \simeq^{\overline{\nu}} e \simeq^{\overline{\nu}} j \succ^{\overline{\nu}} f \succ^{\overline{\nu}} c \succ^{\overline{\nu}} g \succ^{\overline{\nu}} b \simeq^{\overline{\nu}} d \simeq^{\overline{\nu}} h \succ^{\overline{\nu}} i$$

■

Without surprise, Propa$_{1\to\varepsilon}$ returns the same ranking whatever the value of ε.

Proposition 2.4 Let $\oplus \in \{M,S\}$. For any argumentation framework AF, for any $\varepsilon, \varepsilon' \in]0,1]$, it holds that

$$\text{Propa}^{\varepsilon,\oplus}_{1\to\varepsilon}(\mathsf{AF}) = \text{Propa}^{\varepsilon',\oplus}_{1\to\varepsilon}(\mathsf{AF})$$

Finally, one can remark that if we choose $\varepsilon = 0$, the three kinds of semantics return exactly the same order.

Proposition 2.5 Let $\oplus \in \{M,S\}$, for all $\mathsf{AF} \in \mathbb{AF}$,

$$\text{Propa}^{0,\oplus}_{\varepsilon}(\mathsf{AF}) = \text{Propa}^{0,\oplus}_{1+\varepsilon}(\mathsf{AF}) = \text{Propa}^{0,\oplus}_{1\to\varepsilon}(\mathsf{AF})$$

In this case, only the weights propagated by the non-attacked arguments are taken into account. This can make sense, but when there is no non-attacked argument in the AF, all the arguments have a propagation vector composed only of 0, therefore trivializing the result.

2.4 Analysis of ranking-based semantics

We are now able to check which properties are satisfied by some ranking-based semantics introduced in the literature. Some of these semantics have not been originally defined as ranking-based

semantics but rather as gradual semantics. But, as explained in the previous section, we can always induce ranking-based semantics from gradual ones. However, as ranking-based semantics associates a unique ranking to any argumentation framework (see Definition 2.10 on page 12), the scores assigned by a gradual semantics should be unique too. This is why we leave the social argumentation frameworks (Leite and Martins, 2011) and the equational approach (Gabbay, 2012), which may both return multiple rankings, out of this study.[2] I will not present in detail each of these semantics in this document, an interested reader can refer to the original papers, or to Bonzon et al. (2016a, 2023).
The semantics considered in this study are the following:

Categoriser-based ranking semantics (Cat) (Besnard and Hunter, 2001)
Originally, the *categoriser* function was proposed for "deductive" arguments, by assigning a value to a tree of such arguments where each value captures the relative strength of an argument taking into account the strength of its attackers which takes into account the strength of its attackers, and so on. Pu et al. (2014) proved the existence and uniqueness of such a solution for any argumentation framework. The categoriser-based ranking semantics builds a ranking from the categoriser values obtained. The higher the categoriser value of an argument, the more acceptable the argument.

Discussion-based semantics (Dbs) (Amgoud and Ben-Naim, 2013)
This semantics was proposed to take into account only the number of attackers/defenders of a given argument, whatever their quality: the less attackers and the more defenders an argument, the more acceptable the argument. The method lexicographically ranks the arguments based on the number of attackers and defenders. Concretely, we start by comparing the number of direct attackers of each argument. If some arguments are still equivalent (they have the same number of direct attackers), the size of paths is recursively increased until a difference is found or the threshold is reached.

Burden-based semantics (Bbs) (Amgoud and Ben-Naim, 2013)
This semantics follows the same idea as Dbs in considering only direct attackers of arguments. However, the approach is different. Indeed, instead of computing all the possible paths that lead to an argument like Dbs does, each argument receives, at each step, a *burden number* which is simultaneously computed based on the burden numbers of their direct attackers at the previous step. Two arguments are lexicographically compared based on their burden numbers.

α-Burden-based semantics (α-Bbs) (Amgoud et al., 2016)
A broad class of ranking semantics that allows one to choose to which extent to prioritize the quality of attacks (i.e. the scores of the direct attackers) over their quantity (i.e. the number of direct attackers) (or vice versa). This principle, called compensation, can be checked when several weak attacks (*i.e.* direct attackers of an argument are attacked) could have the same impact as one strong attack (*i.e.* direct attackers are not attacked). The parameter α is both used for the compensation and to ensure the uniqueness of the solution. Indeed, contrary to Bbs where the lexicographical order is used, α-Bbs uses a fixed-point iteration to find the burden number of each argument.

Valuation with tuples (Tuples) (Cayrol and Lagasquie-Schiex, 2005)
This semantics is defined as a "global" approach where only the defense and attack branches of an argument are taken into consideration to compare arguments. A structure, called tupled value, is first defined to store, for each argument, the set of the lengths of the branches

[2] It was discovered (Amgoud et al., 2017b) that the conjecture about the uniqueness of social models only holds up to 3 arguments in the argumentation framework.

(attack and defense branches are considered separately) leading to this argument. When cycles exist in the AF, there may be no non-attacked argument and thus no branch. The solution proposed in (Cayrol and Lagasquie-Schiex, 2005) is to consider that a cycle is like an infinity of branches which gives an infinite acyclic graph. Once the tupled values have been computed for each argument, the next step consists in comparing them. To do so, the number of attack and defense branches of two arguments are first compared and, in case of a tie, the values inside each tuples are lexicographically compared. Thus, the priority is given to the quantity, and the quality is taken into consideration only if the quantity cannot allow to decide between two arguments.

Game of argumentation strategy (2ZG) (Matt and Toni, 2008)

Matt and Toni (2008) compute the strength of an argument using a two-person zero-sum strategic game. This game confronts two players, a proponent and an opponent for a given argument, where the strategies of the players are sets of arguments. The goal of the game is to evaluate the interactions between the strategies chosen by the two players. Matt and Toni ensure that, in a dispute, it is better for the proponent of an argument to have more attacks against opponents to this argument and fewer attacks from them. To capture this idea, they introduced the notion of degree of acceptability of a set of arguments with respect to another one.

Fuzzy labeling (FL) (da Costa Pereira et al., 2011)

da Costa Pereira et al. (2011) study how an agent changes her mind in response to new information/arguments. For this, they combine belief revision and argumentation in a single framework close to Dung's framework, called fuzzy argumentation framework, where a degree of trust is first assigned to each argument. When a new argument is proposed, it has more or less influence on the evaluation of existing arguments according to its degree of trust. Even if this work does not directly propose a ranking-based semantics, the score obtained by each argument after computation could be used to rank the arguments. To compare this semantics with the existing ranking-based semantics in the classical framework, we consider that all arguments are trusted.

Iterated Graded Defense (IGD) (Grossi and Modgil, 2015, 2019)

This semantics proposes a generalization of Dung's notion of acceptability. The theory is based on two assumptions: (A1) having fewer direct attackers is better than having more; and (A2) having more direct defenders is better than having fewer. To catch these two principles, Grossi and Modgil define a generalization of the notion of defense initially defined by Dung.

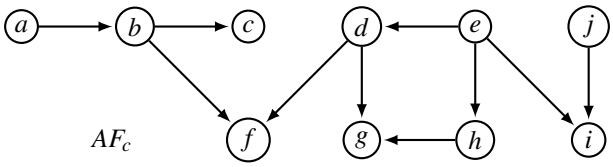
Counting semantics (CS) (Pu et al., 2015a,b)

This semantics allows to rank arguments by counting the number of their respective attackers and defenders. However, contrary to the tuple-based semantics which only focuses on the branches, the counting semantics takes into account a large part of paths that leads to a given argument (and which continues the process even if a difference is found, contrary to the Discussion-based semantics). To assign a value to each argument, they consider an argumentation framework as a dialogue game between proponents of a given argument x (*i.e.* the defenders of x) and opponents of x (*i.e.* the attackers of x). The idea is that an argument is more acceptable if it has many arguments from proponents and few arguments from opponents. In (Pu et al., 2015a), they deepen their work by presenting some complements about how the damping factor α allows to control the convergence speed of the computation for the counting semantics.

I will also add the propagation semantics (see Section 2.3 on page 19).

2.4.1 An Experimental Comparison of Ranking-based Semantics

As it can be easily checked with the rankings obtained with the different ranking-based semantics on AF_c (the AF and the results are given in Figure 2.3), the ranking-based semantics we presented here mostly return distinct rankings between arguments.



Semantics	*Ranking between arguments*
FL	$a \simeq c \simeq e \simeq f \simeq g \simeq j \succ b \simeq d \simeq h \simeq i$
2ZG	$a \simeq e \simeq j \succ c \simeq f \simeq g \succ b \simeq d \simeq h \simeq i$
Cat	$a \simeq e \simeq j \succ c \succ b \simeq d \simeq f \simeq g \simeq h \succ i$
1-Bbs	
Dbs	$a \simeq e \simeq j \succ c \succ b \simeq d \simeq h \succ f \simeq g \succ i$
Bbs	
0.5-Bbs	
CS	
$\text{Propa}_{\varepsilon}^{0.75,M}$	
$\text{Propa}_{\varepsilon}^{0.75,S}$	$a \simeq e \simeq j \succ c \succ b \simeq d \simeq h \succ f \succ g \succ i$
5-Bbs	$a \simeq e \simeq j \succ c \succ f \simeq g \succ b \simeq d \simeq h \succ i$
IGD	
$\text{Propa}_{\varepsilon}^{0.3,M}$	
$\text{Propa}_{1+\varepsilon}^{\varepsilon,M}$	
$\text{Propa}_{\varepsilon}^{0.3,S}$	$a \simeq e \simeq j \succ c \succ f \succ g \succ b \simeq d \simeq h \succ i$
$\text{Propa}_{1+\varepsilon}^{\varepsilon,S}$	
$\text{Propa}_{1\to\varepsilon}^{\varepsilon,S}$	$a \simeq e \simeq j \succ f \succ c \succ g \succ b \simeq d \simeq h \succ i$
Tuples	$a \simeq e \simeq j \succ f \simeq g \succ c \succ b \simeq d \simeq h \succ i$
$\text{Propa}_{1\to\varepsilon}^{\varepsilon,M}$	

Figure 2.3: Rankings obtained on AF_c with some existing ranking-based semantics

However, this difference concerns a subset of arguments (here b, c, d, f, g, h) and not all the arguments. Conversely, some common behaviors seem to appear between the semantics like the fact that a, e and j are always equally acceptable and more acceptable than all the other arguments (except for the semantics FL) or that i is always ranked last (even if it can be a tie). Thus, it could be interesting to know if such information can be generalized to all the argumentation frameworks. The goal of this section consists of evaluating experimentally how different or similar are these ranking-based semantics. For this purpose, we have chosen to compute and compare the ranking of each ranking-based semantics on several randomly generated argumentation frameworks.
But before explaining how to compute and compare the different rankings, we decided to exclude some ranking-based semantics from this study. Indeed, it is difficult to compare total and partial preorders (because some arguments could be incomparable), which is why the semantics that return a partial preorder, like IGD (Grossi and Modgil, 2015) and Tuples (Cayrol and Lagasquie-Schiex, 2005), were excluded. The semantics 2ZG (Matt and Toni, 2008) was also excluded from this study because, according to the authors, this semantics can only be used for argumentation frameworks with less than a dozen arguments. Indeed, the size of the players' strategy spaces grows exponentially fast with the total number of arguments in the argumentation framework considered so when it contains more than twelve arguments, the computation becomes almost impossible.
To experimentally compare the rankings returned by the ranking-based semantics, we considered a significantly large experimental setting with a great variety of benchmarks. This set was separated

into two parts: the first one contains AFs similar to those used in online debates and the second one contains AFs generated from standard random AF generators.

Debate graphs

Ranking-based semantics seem to be a promising tool to evaluate online debates (Amgoud, 2019; Egilmez et al., 2013). We, therefore, chose to randomly generate argumentation frameworks[3] with the same characteristics as the most popular online debates. This type of graph is characterized by a specific argument (the issue) that plays the role of the main question of the debate and several branches converging towards this argument with the condition that all arguments must be connected to this issue. To build such a graph (an instance is illustrated in Figure 2.4), we created a new generator using several tools from the `networkx` package. First, a directed star graph with $n+1$ nodes is created where the central node is the target argument connected to n outer nodes (i.e. n arguments that directly attack the target argument). For each node representing a branch of the star, a directed tree containing m additional nodes is generated where edges are oriented into this node.

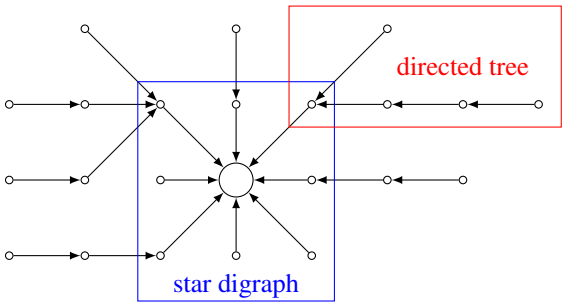


Figure 2.4: A debate graph

For our experiment, we generated 10000 debate graphs whose value of the integer n varies randomly between 6 and 15 and whose value of the integer m varies randomly between 1 and 6. In the following, we collectively refer to this group of AFs as `debateGraph10000`. Figure 2.5 on the next page shows the distribution of the number of arguments in the benchmark `debateGraph10000`.

Random graphs

We also considered an experimental setting representing three different models used during the ICCMA competition[4] as a way to generate random argumentation graphs:

1. the Erdös-Rényi model (ER) which generates graphs by randomly selecting attacks between arguments;
2. the Barabasi-Albert model (BA) which provides networks, called scale-free networks, with a structure in which some nodes have a huge number of links, but in which nearly all nodes are connected to only a few other nodes; and
3. the Watts-Strogatz model (WS) which produces graphs that have small-world network properties, such as high clustering and short average path lengths.

[3] Ideally, we would have liked to create a set of AFs from real online debates but actual online debates have additional features (e.g. a support relation, or votes on arguments) that are not taken into account by the semantics studied in this work.

[4] http://argumentationcompetition.org/

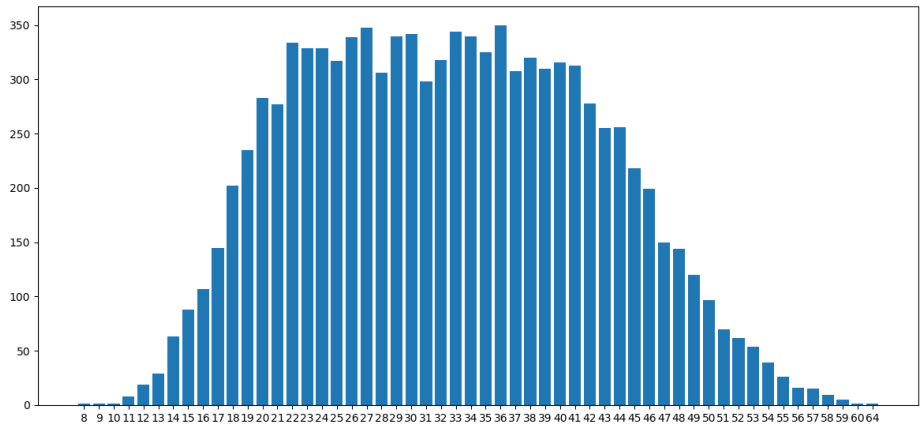


Figure 2.5: Histogram showing the distribution of the number of arguments in the benchmark `debateGraph10000`. The average number of arguments is 32.5176.

The generation of these three types of AFs was done by the AFBenchGen2 generator (Cerutti et al., 2017). We generated a total of 9460 AFs almost evenly distributed between the three models (3000 AFs for the WS model and 3230 AFs for the ER and BA models). For each model, the number of arguments varies among $\texttt{Arg} = \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$. In the following, we collectively refer to the group of AFs generated using the Erdös-Rényi model (resp. Barabasi-Albert model and Watts-Strogatz model) as `rER` (resp. `rBA` and `rWS`). Finally, the notation `randomAF` refers to the union of these three groups.

Comparison and results

Let us now detail how we compare these rankings to represent the concordance of the ranking-based semantics. A way to compare these semantics based on the rankings previously computed consists in using Kendall's tau coefficient (Kendall, 1938). This value corresponds to the total number of rank disagreements over all unordered pairs of arguments between two rankings from distinct semantics. It, therefore, allows us to obtain a dissimilarity degree between the two rankings.
From the rankings computed for each argumentation framework in input, we compute Kendall's tau coefficient between all pairs of rankings.[5] Finally, for each pair of ranking-based semantics, we average Kendall's tau coefficients computed from rankings for each argumentation framework and multiply the result by 100 to obtain a percentage of dissimilarity. All the results are given in a symmetric matrix (Table 2.1 on the following page).
Thus, the biggest dissimilarity degree between two ranking-based semantics is observed between the discussion-based semantics (Dbs) and the fuzzy labelings (FL) with a value of 25.43%. FL clearly stands out from the other semantics with a degree of dissimilarity always greater than 24%. But, globally, the other ranking-based semantics seems to share a solid common basis with a dissimilarity degree often smaller than 10%.

To better represent the "closeness" between these ranking-based semantics, from the previous matrix, we compute a dendrogram, which is a graphical representation of the results of hierarchical cluster analysis. In our case, the method used is a stepwise algorithm for n semantics which merges

[5] The code and benchmarks are available online at
https://github.com/jeris90/comparison_rankingsemantics.git

	Cat	Bbs	Dbs	0.3-Bbs	1-Bbs	10-Bbs	FL	CS	$\text{Propa}_{\varepsilon}^{0.5,S}$	$\text{Propa}_{\varepsilon}^{0.5,M}$	$\text{Propa}_{1+\varepsilon}^{0.5,S}$	$\text{Propa}_{1+\varepsilon}^{0.5,M}$	$\text{Propa}_{1\to\varepsilon}^{0.5,S}$	$\text{Propa}_{1\to\varepsilon}^{0.5,M}$
Cat	0	1.88	2.00	1.56	0	2.21	25.09	1.55	2.84	1.94	3.38	2.48	7.63	7.27
Bbs	1.88	0	0.77	2.48	1.88	2.13	25.37	1.44	2.43	1.22	3.01	1.79	7.44	6.79
Dbs	2.00	0.77	0	2.49	2.00	2.40	25.43	0.97	2.45	0.67	3.04	1.25	7.48	6.25
0.3-Bbs	1.56	2.48	2.49	0	1.56	3.28	25.33	2.08	3.68	2.93	4.24	3.49	8.44	8.22
1-Bbs	0	1.88	2.00	1.56	0	2.21	25.09	1.55	2.84	1.94	3.38	2.48	7.63	7.27
10-Bbs	2.21	2.13	2.40	3.28	2.21	0	24.55	2.46	2.97	1.76	2.40	1.19	6.17	5.46
FL	25.09	25.37	25.43	25.33	25.09	24.55	0	25.35	24.40	24.94	24.16	24.71	24.32	25.06
CS	1.55	1.44	0.97	2.08	1.55	2.46	25.35	0	2.71	1.47	3.30	2.05	7.62	6.94
$\textbf{Propa}_{\varepsilon}^{0.5,S}$	2.84	2.43	2.45	3.68	2.84	2.97	24.40	2.71	0	1.86	0.60	2.43	5.06	7.40
$\textbf{Propa}_{\varepsilon}^{0.5,M}$	1.94	1.22	0.67	2.93	1.94	1.76	24.94	1.47	1.86	0	2.41	0.60	6.84	5.60
$\textbf{Propa}_{1+\varepsilon}^{0.5,S}$	3.38	3.01	3.04	4.24	3.38	2.40	24.16	3.30	0.60	2.41	0	1.84	4.47	6.82
$\textbf{Propa}_{1+\varepsilon}^{0.5,M}$	2.48	1.80	1.25	3.49	2.48	1.19	24.71	2.05	2.43	0.60	1.84	0	6.28	5.03
$\textbf{Propa}_{1\to\varepsilon}^{0.5,S}$	7.63	7.44	7.48	8.44	7.63	6.17	24.32	7.62	5.06	6.84	4.47	6.28	0	2.45
$\textbf{Propa}_{1\to\varepsilon}^{0.5,M}$	7.27	6.79	6.25	8.22	7.27	5.46	25.06	6.94	7.40	5.60	6.82	5.03	2.45	0

Table 2.1: Average of Kendall's tau coefficients on `debateGraph10000` and `randomAF`. The darker the color of the cell, the greater the dissimilarity between the two ranking-based semantics.

two semantics or clusters with the least dissimilarities at each step until obtaining a unique cluster. Several operators exist (Tan et al.) to compute the distance between the new cluster and the other clusters like the single link (minimum), complete link (maximum), group average, median, etc. However, a few numbers of inputs make the differences negligible between these methods, so we chose the average method to compute the dendrogram illustrated in Figure 2.6 on the next page. On this dendrogram, the height of the branch between two clusters indicates how different they are from each other: the greater the height, the greater the difference. Four groups emerge from this study: one containing the semantics Dbs, Bbs and CS (which have a dissimilarity degree always smaller than 1.5%), another one containing the semantics Cat, 0.3-Bbs and 1-Bbs (which have a dissimilarity degree always smaller than 1.56%), another one containing $\text{Propa}_{\varepsilon}^{0.5,S}$ and $\text{Propa}_{1+\varepsilon}^{S}$ (which have a dissimilarity degree equals to 0.6%), and the last one containing 10-Bbs and and $\text{Propa}_{\varepsilon}^{0.5,M}$ and $\text{Propa}_{1+\varepsilon}^{M}$ (which have a dissimilarity degree always smaller than 1.76%). The propagation semantics $\text{Propa}_{1\to\varepsilon}$ seem closer to the third group of semantics with a dissimilarity degree between 4% and 6% with all these ranking-based semantics. Among these groups, one can observe that some semantics are very close like Bbs and Dbs with a dissimilarity value of 0.77%. An important observation is that the categoriser-based semantics and the α-Burden-based semantics always return the same ranking (their dissimilarity degree is 0%) when $\alpha = 1$, as noticed in Amgoud et al. (2016).

2.4.2 Properties × Ranking-based semantics

We are now in a position to check which of the properties introduced in Section 2.2.1 on page 13 are satisfied by these ranking-based semantics. Recall that, among these ranking-based semantics, some of them (*e.g.* α-burden-based semantics, the propagation semantics) are configurable with one or several parameters. Thus, two values of a parameter could give different rankings. It is why we consider that a property is satisfied by these ranking-based semantics only if the property is satisfied for all the values of a parameter.

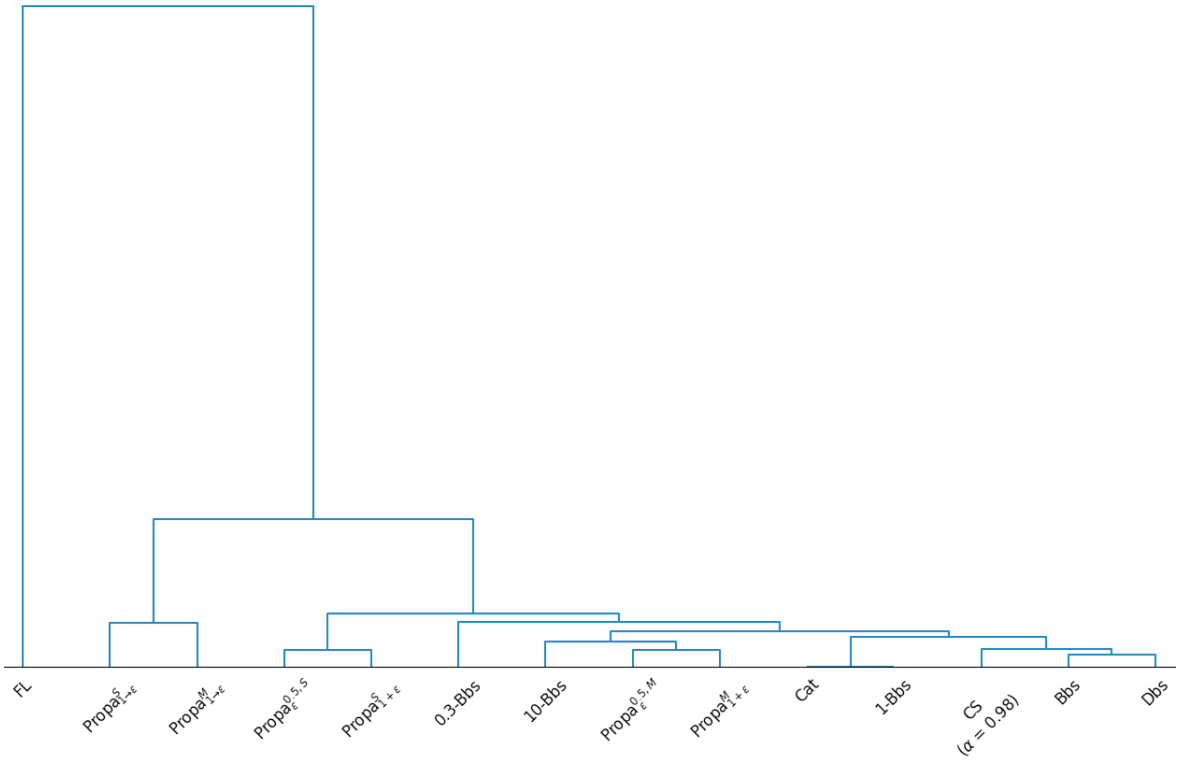


Figure 2.6: Dendrogram representing the relationships between the ranking-based semantics studied in this work

Proposition 2.6 The properties that are satisfied by each ranking-based semantics (the other properties are not satisfied by the corresponding ranking-based semantics):

- The categoriser-based ranking semantics (Cat) satisfies Abs, In, VP, DP, CT, SCT, ↑AB, ↑DB, +AB, Tot, NaE, AE and OE.
- The discussion-based semantics (Dbs) satisfies Abs, In, VP, DP, CT, SCT, CP, ↑AB, ↑DB, +AB, Tot, NaE, AE and OE.
- The burden-based semantics (Bbs) satisfies Abs, In, VP, DP, CT, SCT, CP, DDP, ↑AB, ↑DB, +AB, Tot, NaE, AE and OE.
- Let $\alpha \in]0,+\infty[$. The α-burden-based semantics (α-Bbs) satisfies Abs, In, VP, DP, CT, SCT, ↑AB, ↑DB, +AB, Tot, NaE, AE and OE.
- The fuzzy labeling (FL) satisfies Abs, In, CT, QP, Tot, NaE, AE, OE and AvsFD.
- Let $\alpha \in]0,1[$. The counting semantics (CS) satisfies Abs, VP, DP, CT, SCT, ↑AB, ↑DB, +AB, Tot, NaE, AE and OE.
- The tuples-based semantics (Tuples) satisfies Abs, In, VP, +DB, ↑AB, ↑DB, +AB, NaE, AE, OE and AvsFD.
- The ranking-based semantics 2ZG satisfies Abs, In, VP, +AB, SC, Tot, NaE and AvsFD.
- The iterated graded defense semantics (IGD) satisfies Abs, In, VP, +AB, NaE and AE.
- The ranking-based semantics $\text{Propa}^{\varepsilon,\oplus}_{\varepsilon}$ satisfies Abs, In, VP, DP, ↑AB, ↑DB, +AB, NaE, Tot and AE. When $\oplus = M$, $\text{Propa}^{\varepsilon,M}_{\varepsilon}$ also satisfies CT, SCT and OE. The other properties are not satisfied.

- The ranking-based semantics $\text{Propa}_{1+\varepsilon}^{\varepsilon,\oplus}$ satisfies Abs, In, VP, DP, DDP, ↑AB, ↑DB, +AB, Tot, NaE, AE and AvsFD. When $\oplus = M$, $\text{Propa}_{1+\varepsilon}^{\varepsilon,M}$ satisfies CT, SCT and OE. The other properties are not satisfied.
- The ranking-based semantics $\text{Propa}_{1\to\varepsilon}^{\varepsilon,\oplus}$ satisfies Abs, In, VP, DP, DDP, +DB, ↑AB, ↑DB, +AB, Tot, NaE, AE and AvsFD. When $\oplus = M$, $\text{Propa}_{1\to\varepsilon}^{\varepsilon,M}$ satisfies OE. The other properties are not satisfied.

We also checked what are the properties satisfied by the usual Dung's grounded semantics which is the only semantics to return an unique extension. The idea is to give some hints on the compatibility of these properties with classical semantics. Note that, in this case, this is a degenerate ranking semantics with only two levels (accepted/rejected):

Proposition 2.7 The grounded semantics (Gr) satisfies Abs, In, Tot, NaE, AE and AvsFD. The other properties are not satisfied.

We summarize all these results in Table 2.2.

Properties	Cat	Dbs	Bbs	α-Bbs	FL	CS	$\text{Propa}_{\varepsilon}$	$\text{Propa}_{1+\varepsilon}$	$\text{Propa}_{1\to\varepsilon}$	Tuples	2ZG	IGD	Gr
Abs	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
In	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓
VP	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	×
DP	✓	✓	✓	✓	×	✓	✓	✓	✓	×	×	×	×
CT	✓	✓	✓	✓	✓	✓	$\checkmark_M$	$\checkmark_M$	×	×	×	×	×
SCT	✓	✓	✓	✓	×	✓	$\checkmark_M$	$\checkmark_M$	×	×	×	×	×
CP	×	✓	✓	×	×	×	×	×	×	×	×	×	×
QP	×	×	×	×	✓	×	×	×	×	×	×	×	×
DDP	×	×	✓	×	×	×	×	✓	✓	×	×	×	×
SC	×	×	×	×	×	×	×	×	×	×	✓	×	×
⊕DB	×	×	×	×	×	×	×	×	×	×	×	×	×
+DB	×	×	×	×	×	×	×	×	✓	✓	×	×	×
↑AB	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	×	×	×
↑DB	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	×	×	×
+AB	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	×	×	×
Tot	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	×	✓
NaE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
AE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓
OE	✓	✓	✓	✓	✓	✓	$\checkmark_M$	$\checkmark_M$	$\checkmark_M$	✓	×	×	×
AvsFD	×	×	×	×	✓	×	×	✓	✓	✓	✓	×	✓

Table 2.2: Properties satisfy by the studied ranking semantics. A cross × means that the property is not satisfied, symbol ✓ means that the property is satisfied and the shaded cells highlight the results already proven in the literature.

2.4.3 Discussion

Several observations can be made regarding these axioms and the results reported in Table 2.2.

Some properties seem to be widely accepted and shared by almost all semantics. It is the case with the properties Abs, In, VP, +AB, Tot, NaE and AE. We recall that the input is a Dung's abstract argumentation framework without information about the nature of arguments, so only the attacks have to be taken into account, hence the importance of Abs. Concerning property Independence (In), it seems difficult to explain the fact that an argument can influence other arguments without an existing link between them. The only semantics which does not satisfy this property is the counting semantics (CS) which needs the maximal indegree to guarantee convergence. All the semantics consider the non-attacked arguments as the best arguments in an argumentation framework (VP),

even if sometimes (for FL and Gr) non-attacked arguments are not ranked strictly higher than other arguments. Nevertheless, there are situations where VP does not seem appropriate (e.g. see the study of protocatalepsis in the context of persuasion Section 2.5 on page 35). NaE and AE are also satisfied by all semantics (except for 2ZG which does not satisfy AE, and is incompatible with SC). This is a kind of compatibility principle with usual Dung's semantics (the grounded semantics satisfies them too) where only your attackers should impact your ranking, not the arguments you attack. It is also interesting to note that almost all the semantics satisfy the property Total allowing a direct utilization in real applications wanting to distinguish all the arguments. A last property satisfied by almost all semantics is +AB, which states that adding an attack branch toward an argument degrades its ranking. This also seems to be a perfectly natural requirement for ranking semantics: the more you are attacked, the worse you are. Furthermore, this property is one of the main reasons to explain the difference observed between the semantics FL which does not satisfy it and all the other semantics. Indeed, FL extends the complete semantics by considering varying degrees of acceptability (rather than the three classical ones: in, out and undec) and thus does not take into account the number of attackers, while all the other semantics do.

Some properties are very discriminatory and provide a rough classification of semantics. If some incompatibilities between properties exist, some other properties allow to separate the semantics into sub-classes groups. It is the case with the properties (S)CT and AvsFD (or +DB) which are always satisfied by at least one semantics (except for $\text{Propa}_{1+\varepsilon}$ where both are accepted). Indeed, different approaches (without being incompatible) concerning the defense are considered by these properties. The semantics that satisfy AvsFD take care of the whole branches of attack/defense. Whereas for the semantics that satisfies (S)CT, a defense branch (that still ends by an attack towards the argument) always penalizes it. Such properties reveal the elements in an argumentation framework causing differences between the rankings.

More specific properties. These properties operate at different levels. There are 'local' properties (*e.g.* CP, QP, DP, DDP, (S)CT) focusing on the direct attackers (or defenders) which can be justified in some situations but seem hardly general (and sometimes impossible to reconcile with some more global properties, as Proposition 2.1 on page 18 shows). And properties related to 'change' (*e.g.* $\oplus$DB, +DB, $\uparrow$AB, $\uparrow$DB, +AB) seem very appealing because they specify how the ranking should be affected based on the comparison of attack and defense branches. They allow, for example, to categorize the semantics according to the behavior towards some basic requirements like the defense with +DB (see the previous observation). Another example of their interest is that, in focusing on the two semantics Tuples and 2ZG, it seems clear that the properties related to 'change' satisfied by these two semantics allow to distinguish these two semantics.

Defining axiomatically the worst arguments is not obvious. Interestingly, while almost all semantics agree axiomatically on which arguments should be the best in an argumentation framework (VP), there is no consensus regarding the *worst* arguments. SC is very interesting in that respect. It observes that a self-contradicting argument is intrinsically flawed, without even requiring other arguments to defeat it. But as can be observed none of the semantics comply with it, except the one of Matt and Toni (2ZG) who introduced the property. The explanation is that all semantics consider that an argument that attacks itself is a path like the other ones. So an argument that attacks itself (and by no other argument) is better than an argument that is attacked several times. On the other hand, another possibility is when the properties +AB and $\uparrow$AB are satisfied together. Indeed, one can consider the worst argument as the one which is directly attacked by a maximum (+AB) of non-attacked arguments ($\uparrow$AB).

The interplay of axioms is often instructive. In Propositions 2.1 and 2.2 on page 18, we have

identified some implications and incompatibilities between axioms. Let us focus, for example, on the relation of incompatibility between VP and $\oplus$DB. One can easily remark that $\oplus$DB is more general than +DB, and in a sense more natural: the property is stated for *any* cases, and it does not treat some arguments (the non-attacked arguments here) differently. But it contradicts VP in this case. +DB is a less "systematic" property (it was the original one proposed in Cayrol and Lagasquie-Schiex (2005)) but is compatible with VP: if one accepts that non-attacked arguments should be the best (VP), then adding a defense branch cannot *always* improve the situation of a given argument.

This set of axioms is yet to be augmented. This can be observed with the semantics Categorizer, α-Burden-based semantics and Propa$_\varepsilon$ which satisfy the same set of properties, whereas they have quite different definitions and behaviors as it is revealed by our experimental study. This means that at least one logical property (if it exists) is lacking to discriminate these operators.

Towards an application-oriented axiomatic analysis. Let us recall that we do not claim that all of the properties presented in Section 2.2.1 are required. However, at this level of abstraction, they allow us to compare and better understand the ranking-based semantics. Indeed, while our objective has been to offer the broadest possible picture of ranking-based semantics by presenting a large catalogue of properties, we certainly believe that the relevance of each axiom should be ultimately evaluated with respect to the application at hand. Depending on the context, a designer may only focus a subset of axioms, or even challenge a specific property often assumed in existing semantics (as, for instance, in the case of persuasion where VP may not be desirable (see Section 2.5 on the facing page)). In line with the work initiated in Vesic et al. (2022) for gradual semantics, it would be interesting to target the mandatory properties for some practical aspects of argumentation (persuasion, negotiation, online debate, etc.).

Link between extension-based semantics and ranking-based semantics. One can note that Abs, In, AE, NaE, Tot and AvsFD are satisfied by the grounded semantics. However, among the properties widely accepted by the ranking-based semantics, VP and +AB are not satisfied by the grounded semantics. An explanation is related to the fact that extension-based semantics (and in particular grounded semantics) consider that the impact of an attack from an argument to another one is drastic. In other words, the grounded semantics falsifies VP and +AB because an attack can "kill" another argument (see the discussion in Amgoud and Ben-Naim (2013)) while the ranking-based semantics suppose that an attack does not "kill" but can just weaken the attacked argument. However, these two principles are not totally incompatible with grounded semantics to some extent. Indeed, as suggested in Thimm and Kern-Isberner (2014), a weak version of Void Precedence, which states that non-attacked arguments should be at least as acceptable as (and not strictly more acceptable) attacked arguments, can also be defined.

Property 21 — weak Void Precedence (wVP). (Thimm and Kern-Isberner, 2014)
The rank of a non-attacked argument is higher or equal to the rank of any attacked argument.

Clearly, Void Precedence implies weak Void Precedence so all the semantics which satisfy VP also satisfy wVP. But it is interesting to note that the grounded semantics satisfies wVP because the non-attacked arguments are always accepted but can be equal to some other attacked arguments. Following the same reasoning, the weak version of some other existing properties[6] can be defined and satisfied by the grounded semantics. Let us check which of them is satisfied by the grounded semantics.

[6]Defining the weak version of a property means replacing the strict comparison operator between two arguments with a strict or equal comparison operator without changing the conditions.

Proposition 2.8 The grounded semantics satisfies the weak version of VP, DP, QP, DDP, SC, $\oplus$DB, $\uparrow$DB, $\uparrow$AB and +AB.

It is clear that in this case, the grounded semantics satisfies many more properties. This is because there exist only two levels of acceptability (accepted/rejected) to evaluate the arguments. Thus, if an argument is considered as accepted (resp. rejected) then it will always be considered more (resp. less) acceptable regardless of the acceptability of the other argument. But in general, these weak properties are less interesting for the ranking-based semantics precisely because of the many levels of acceptability. They can nevertheless make sense when they are combined.[7]

2.5 Ranking semantics for persuasion

As discussed in the previous section, if many ranking-based semantics were compared based on their properties, the relevance of some properties may be very much dependent on the context of the application. Indeed, what is often missing to compare these approaches is thus a clear indication of the applications they target. For example, for online debate platforms, satisfying the property Total (Tot) may seem natural to ensure the comparison between all the arguments and thus guarantee a result to the users. Conversely, for the same platforms where votes are assigned to each argument and represent their social support, a possibility would be to reward a more aggressive non-attacked argument. Thus, such property as Non-attacked Equivalence (NaE), considering that all the non-attacked arguments should be equally acceptable, should not be satisfied.
Among the many existing fields in argumentation (e.g., negotiation, persuasion, deliberation), we choose to focus on the context of persuasion in argumentation. Persuasion is an activity that involves one party (the persuader) trying to induce another party (the persuadee) to believe (or not believe) certain information or to do (or not do) some action.
Among the processes used in this specific domain, we shall concentrate on two well-documented phenomena in persuasion (procatalepsis and fading) and draw a parallel between each of them and existing properties for ranking-based semantics.[8]

Procatalepsis

The *procatalepsis* principle aims to strengthen an argument by dealing with possible counter-arguments before their audience can raise them. To illustrate this principle, we extend an example from (Besnard and Hunter, 2008, p.85): a (fictional) sales pitch intended to persuade someone to buy a specific car.

Example 2.9

($a1$) The car x is a high-performance family car with a diesel engine and a price of 32000

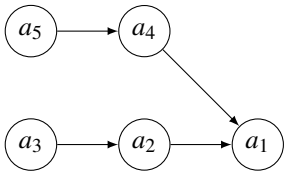
($a2$) In general, diesel engines have inferior performance compared with gasoline engines

($a3$) But, with these new engines, the difference in performance [...] is negligible

($a4$) In addition, even though the price of the car seems high

($a5$) It will be amortized because diesel engines run longer before breaking than any other kind of engine.

[7] See Section 2.6 on page 45 for a discussion on this subject.
[8] This work was originally published in (Bonzon et al., 2017, 2021)



The seller's objective is here to increase the acceptability of a_1. For this purpose, he anticipates two arguments a_2 and a_4 (which could have been proposed by the customer later in the discussion) by providing two counter-arguments a_3 and a_5 which attack a_2 and a_4 respectively. ■

In this kind of persuasion context, it can be more convincing to state the more plausible counter-argument to a_1 to provide a convincing defense against them, than simply stating a_1 alone. These anticipations make it possible to persuade the interlocutor that any attack against a_1 is in vain. In addition, it becomes difficult for the persuadee to find arguments against a_1 if the persuader anticipates most of them. In terms of ranking, a_1 with several defense branches could be seen as strictly more acceptable than a_1 without any branch.

To define the procatalepsis principle in the context of abstract argumentation, we first need to define the argumentation frameworks that we call *persuasion pitches*. A persuasion pitch is a tree-shaped argumentation framework where an argument x, called the *targeted argument*, has only defense branches.

Definition 2.21 (Persuasion pitch) An argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ is a **persuasion pitch** with $x \in \mathcal{A}$ as the targeted argument if the following conditions are satisfied:

1. $\mathcal{B}^-(x) = \emptyset$,
2. $\mathcal{B}^+(x) \neq \emptyset$,
3. $\forall y \in \mathcal{A} \backslash \{x\}$, there exists a unique path from y to x.

A persuasion pitch is denoted by $\mathcal{P}_k(x)$ where $k = |\mathcal{B}^+(x)|$ is the number of defense branches of its targeted argument x. When the targeted argument x of the persuasion pitch is clear, we will use $\mathcal{P}_k$ instead of $\mathcal{P}_k(x)$.

Condition (1) means that the targeted argument has no attack branch while condition (2) states that it has at least one defense branch (the length of defense branches does not matter here). Condition (3) guarantees that the argumentation framework is in the shape of a tree centered on the targeted argument x, i.e. it contains only the arguments used in the pitch which are related to x (i.e. for which a path to x exists).

Example 2.9 **(continuing from p. 35)** This argumentation framework is an example of persuasion pitch $\mathcal{P}_2$ where a_1 is the targeted argument with two defense branches. ■

Property 22 — Procatalepsis (PR).
A ranking-based semantics satisfies Procatalepsis if a number of defense branches are sufficient to make a targeted argument from a persuasion pitch at least as acceptable as when it does not have any (i.e. it is not attacked).

As it can be too restrictive to claim that adding a unique defense branch will always be sufficient to make the targeted argument at least as acceptable as a non-attacked argument, we deliberately made this definition a little more flexible than the original one. Indeed, one can, for example, think that in Example 2.9, the defense branch composed of arguments a_2 and a_3 is not sufficient on its own to make a_1 more acceptable than when it was not attacked and that it is the combination of the two defense branches that allow it. This flexibility allows us to keep the idea that the more defense branches there are, the harder it will be for the opponent to find arguments to attack a_1.

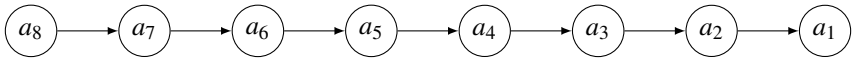
It is striking that procatalepsis blatantly contradicts the property Void Precedence (VP), considering

that a non-attacked argument is strictly more acceptable than an attacked argument.

Proposition 2.9 Void Precedence (VP) and Procatalepsis (PR) are incompatible.

Fading

The *fading* effect states that long lines of argumentation become ineffective in practice because the audience easily loses track of the relation between the arguments.
This principle is supported in practice by the work of Tan et al. (2016) which shows (in the context of their study, an extensive analysis of persuasive debates which took place on the subreddit "ChangeMyView"[9]), that arguments located at a distance greater than 10 from another argument (i.e., 5 rounds of back-and-forth), have no impact in the debate.
More precisely, the fading principle concerns the limit until which the length of a path between an argument and another one is too long to have an impact on the targeted argument.



It seems natural to think that the closer an attacker (respectively, defender) is to an argument, the more effect it has on this argument. For example, argument a_2 should have more impact on a_1 than any other argument in this argumentation framework because a_2 is the direct attacker of a_1. The idea is then that the impact is gradually reduced when the length of the path between two arguments increases. This principle of attenuation is already partially captured by two existing properties: Increase of an attack branch (↑AB) and Increase of a defense branch (↑DB) which state that increasing the length of an attack (resp. defense) branch of an argument increases (resp. decreases) its level of acceptability.

2.5.1 Ranking-based semantics taking into account the persuasion principles

Our objective here was to build a ranking-based semantics that allows us to catch the procatalepsis principle and the fading effect.

- For the fading effect, a solution could be to use an attenuation factor to gradually decrease the impact of arguments. This is used for instance by the counting semantics Pu et al. (2015b) where the damping factor α (a value between 0 and 1) is used.
- For the procatalepsis principle, we want that an attacked argument with many defense roots and few or no attack roots (like a_1 in Example 2.9 on page 35) can be more acceptable than a non-attacked argument. To achieve this goal, a solution is to only take into account the defense roots and the attack roots of an argument. Indeed, if we consider that a defense (respectively, attack) root has a positive (respectively, negative) effect on an argument, then, a sufficient number of defense roots (which can be caught by a parameter) would allow this argument to become more acceptable than a non-attacked argument.

Propagation with attenuation

We propose to adapt the propagation principle introduced in Section 2.3 on page 19 with the elements previously put forward. Recall that the idea of propagation is to assign a positive initial value to each argument in the argumentation framework (arguments may start with the same initial value or start with distinct values where non-attacked arguments have greater value than attacked ones). Then each argument propagates its value into the argumentation framework, alternating the polarity according to the considered path (negatively if it is an attack path, positively if it is a defense one).

[9] https://www.reddit.com/r/changemyview/

But, to catch the persuasion principle, we formally redefine the propagation principle by including a damping factor δ which allows decreasing the impact of attackers situated further away along a path (the longer the path length i, the smaller the δ^i).

Definition 2.22 (Attenuated propagation)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework. The valuation $\mathcal{V}$ of $x \in \mathcal{A}$, at step i, is given by:

$$\mathcal{V}_i^{\varepsilon,\delta}(x) = \begin{cases} v_\varepsilon(x) & \text{if } i = 0 \\ \mathcal{V}_{i-1}^{\varepsilon,\delta}(x) + (-1)^i \delta^i \sum\limits_{y \in \mathcal{R}_i^S(x)} v_\varepsilon(y) & \text{otherwise} \end{cases}$$

with $\delta \in]0,1[$ be an attenuation factor and $v_\varepsilon : \mathcal{A} \to \mathbb{R}^+$ is a valuation function giving an initial weight to each argument, with $\varepsilon \in [0,1]$ such that $\forall y \in \mathcal{A}$,

$$v_\varepsilon(y) = \begin{cases} 1 & \text{if } \mathcal{R}_1^S(y) = \emptyset \\ \varepsilon & \text{otherwise} \end{cases}$$

One can remark that the parameter $\oplus$, allowing to make a distinction between the use of the set ($\oplus = S$) or the multiset ($\oplus = M$) to select the attackers or defenders of an argument, is missing in the previous definition compared to the original definition of the propagation principle. Indeed, in this new definition, we choose to only use the set ($\oplus = S$) to guarantee a result to the method.

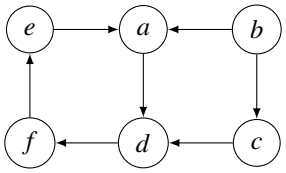


Figure 2.7: The argumentation framework AF_1

Example 2.10 Let us compute the valuation $\mathcal{V}$ of each argument in AF_1, depicted in Figure2.7, when $\varepsilon = 0.5$ and $\delta = 0.4$. The results, at each step, are given in Table 2.3 on the next page. Let us focus on the argument f:

- Step $i = 0$, f begins with an initial weight of 0.5 because it is attacked, $\mathcal{V}_0^{0.5,0.4}(f) = 0.5$
- Step $i = 1$, it negatively receives the value attenuated by δ and sent by its direct attacker d which is also attacked, $\mathcal{V}_1^{0.5,0.4}(f) = \mathcal{V}_0^{0.5,0.4}(f) - 0.4 \times v_{0.5}(d) = 0.3$
- Step $i = 2$, it positively receives the weights from a and c attenuated by δ^2, $\mathcal{V}_2^{0.5,0.4}(f) = \mathcal{V}_1^{0.5,0.4}(f) + 0.4^2 \times (v_{0.5}(a) + v_{0.5}(c)) = 0.46$
- Step $i = 3$, it negatively receives the weight of 1 from b and the weight of 0.5 from e attenuated by δ^3, $\mathcal{V}_3^{0.5,0.4}(f) = \mathcal{V}_2^{0.5,0.4}(f) - 0.4^3 \times (v_{0.5}(b) + v_{0.5}(e)) = 0.364$
- And so on.

■

The following proposition answers the question of convergence of the valuation $\mathcal{V}$. The convergence is guaranteed by the use of the damping factor, but also because the set of arguments that attack or defend an argument for a given length of path is finite and limited by the number of arguments in an argumentation framework. It is why the use of the multiset is impossible here: when a high number of cycles exists, the multiset of arguments can increase very fast with respect to the length of the considered path and the damping factor (even small) is not enough to guarantee the convergence of

$\mathcal{V}_i^{0.5,0.4}$	a	b	c	d	e	f
0	0.5	1	0.5	0.5	0.5	0.5
1	-0.1	1	0.1	0.1	0.3	0.3
2	-0.02	1	0.1	0.34	0.38	0.46
3	-0.052	1	0.1	0.308	0.316	0.364
⋮	⋮	⋮	⋮	⋮	⋮	⋮
14	-0.0402	1	0.1	0.3161	0.3506	0.3736

Table 2.3: Computation of the valuation $\mathcal{V}$ of each argument from AF_1 when $\varepsilon = 0.5$ and $\delta = 0.4$

the formula.

Proposition 2.10 Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, $\delta \in]0,1[$ and $\varepsilon \in]0,1]$. For all $x \in \mathcal{A}$, the sequence $\{\mathcal{V}_i^{\varepsilon,\delta}(x)\}_{i=0}^{+\infty}$ converges.

Let us now compute the propagation number of an argument using a fixed-point iteration (the outcome is guaranteed with the previous proposition).

Definition 2.23 (Propagation number)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, $\delta \in]0,1[$ and $\varepsilon \in]0,1]$. The **propagation number** of an argument $x \in \mathcal{A}$ is:

$$\mathcal{V}^{\varepsilon,\delta}(x) = \lim_{i \to +\infty} \mathcal{V}_i^{\varepsilon,\delta}(x)$$

Example 2.10 **(continuing from p. 38)**
The propagation number of each argument in AF_1 is represented in the shaded cell in Table 2.3. Thus, $\mathcal{V}^{0.5,0.4}(a) = -0.0402$, $\mathcal{V}^{0.5,0.4}(b) = 1$, $\mathcal{V}^{0.5,0.4}(c) = 0.1$, $\mathcal{V}^{0.5,0.4}(d) = 0.3161$, $\mathcal{V}^{0.5,0.4}(e) = 0.3506$ and $\mathcal{V}^{0.5,0.4}(f) = 0.3736$. ∎

Variable-depth propagation

Let us now define ranking-based semantics using the propagation number and taking into consideration the persuasion principles. As stated previously, a solution to catch the procatalepsis principle is to only take into account the roots of the arguments. Formally, it is possible when $\varepsilon = 0$. Indeed, in this case, the non-attacked arguments propagate their weights ($v_\varepsilon(y) = 1$) in the argumentation framework, while attacked arguments have an initial weight of 0. Thus, the propagation number of each argument is only based on the value received by their attack or defense roots. Any pairwise strict comparison (based on propagation number) resulting from this process is fixed.
Let us take the example of the argumentation framework illustrated in Figure 2.8 on the next page. On the one hand, a_1 and b_1 have the same propagation number when $\varepsilon = 0$ because they both have two defense branches and no attack branch. On the other hand, c_1 has three defense branches. Therefore, since c_1 has more branches of defense than a_1 and b_1, c_1 must be more acceptable than a_1 and b_1 to be in agreement with the procatalepsis principle.
However, we do not want our semantics to be too "disconnected" from the principles that make the strength of ranking-based semantics: distinguishing arguments by taking into account the quality and the quantity of the arguments that attack or defend them. This is why we apply a second phase to break ties among arguments equally valued in the first phase. From a technical point of view, this means we re-run the propagation phase but this time setting an initial weight $\varepsilon \neq 0$ to take into account the attacked arguments. For example, if a_1 and b_1 have the same propagation number when $\varepsilon = 0$ in the argumentation framework represented in Figure 2.8, b_1 is directly attacked only once

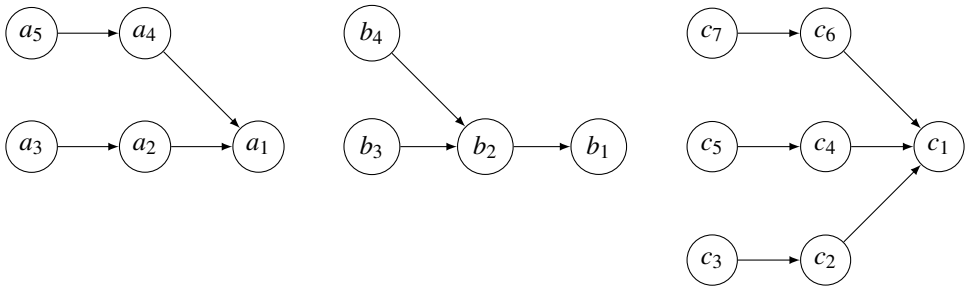


Figure 2.8: Two arguments a_1 and b_1 with two defense branches (but with different configurations) involving a similar propagation number when $\varepsilon = 0$ and c_1 which has three defense branches.

while a_1 is directly attacked twice, so one can consider that b_1 could be more acceptable than a_1.

Definition 2.24 (Variable-Depth Propagation)
Let $\varepsilon \in]0,1]$ and $\delta \in]0,1[$. The ranking-based semantics Variable-Depth Propagation $\mathsf{vdp}^{\varepsilon,\delta}$ associates to any argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ a ranking $\succeq_{\mathsf{AF}}^{\mathsf{vdp}^{\varepsilon,\delta}}$ on $\mathcal{A}$ such that $\forall x, y \in \mathcal{A}$,

$$x \succeq_{\mathsf{AF}}^{\mathsf{vdp}^{\varepsilon,\delta}} y \text{ if and only if } \mathcal{V}^{0,\delta}(x) > \mathcal{V}^{0,\delta}(y) \text{ or } (\mathcal{V}^{0,\delta}(x) = \mathcal{V}^{0,\delta}(y) \text{ and } \mathcal{V}^{\varepsilon,\delta}(x) \geq \mathcal{V}^{\varepsilon,\delta}(y))$$

Example 2.10 **(continuing from p. 38)** According to the previous definition, we first need to compute the propagation number of each argument when $\varepsilon = 0$. Argument b is the only non-attacked argument, so the propagation number of each argument is only based on the value that it propagates. The valuations of each argument at each step are given in the following table:

$\mathcal{V}_i^{0,0.4}$	a	b	c	d	e	f
0	0	1	0	0	0	0
1	-0.4	1	-0.4	0	0	0
2	-0.4	1	-0.4	0.16	0	0
3	-0.4	1	-0.4	0.308	0	-0.064
4	-0.4	1	-0.4	0.308	0.0256	-0.064
⋮	⋮	⋮	⋮	⋮	⋮	⋮
14	-0.4105	1	-0.4	0.1642	0.0263	-0.0657

It is why, until step $i = 3$, e has a valuation of 0, but during step $i = 4$, it receives a positive value from b, so $\mathcal{V}_4^{0,0.4}(e) = 0.4^4 \times v_0(b) = 0.0256$.
We finally obtain the following pre-ranking: $b \succ d \succ e \succ f \succ c \succ a$
As no arguments are equally acceptable here, it is not necessary to perform the second phase. Thus, $\forall \varepsilon \in]0,1]$, $\mathsf{vdp}^{\varepsilon,0.4}$ returns the following ranking:

$$b \succ^{\mathsf{vdp}^{\varepsilon,0.4}} d \succ^{\mathsf{vdp}^{\varepsilon,0.4}} e \succ^{\mathsf{vdp}^{\varepsilon,0.4}} f \succ^{\mathsf{vdp}^{\varepsilon,0.4}} c \succ^{\mathsf{vdp}^{\varepsilon,0.4}} a$$

■

Let us give another example where the second phase is needed to distinguish two arguments.

Example 2.11 Let us compute the ranking returned by $\mathsf{vdp}^{0.5,0.4}$ for the argumentation framework depicted in Figure 2.8, beginning by the case $\varepsilon = 0$.

$\mathcal{V}_i^{0,0.4}$	$a_3,a_5,b_3,b_4,c_3,c_5,c_7$	a_2,a_4,c_2,c_4,c_6	b_2	a_1	b_1	c_1
0	1	0	0	0	0	0
1	1	-0.4	-0.8	0	0	0
2	1	-0.4	-0.8	0.32	0.32	0.48

According to the definition of vdp, we first compare the propagation number of each argument when $\varepsilon = 0$. The result is the following ranking:

$$a_3 \simeq a_5 \simeq b_3 \simeq b_4 \simeq c_3 \simeq c_5 \simeq c_7 \succ c_1 \succ a_1 \simeq b_1 \succ a_2 \simeq a_4 \simeq c_2 \simeq c_4 \simeq c_6 \succ b_2$$

We can see that some arguments are still equally acceptable, in particular a_1 and b_1. So, according to the definition of vdp, we restart the process with a non-zero ε (here $\varepsilon = 0.5$):

$\mathcal{V}_i^{0.5,0.4}$	$a_3,a_5,b_3,b_4,c_3,c_5,c_7$	a_2,a_4,c_2,c_4,c_6	b_2	a_1	b_1	c_1
0	1	0.5	0.5	0.5	0.5	0.5
1	1	0.1	-0.3	0.1	0.3	-0.1
2	1	0.1	-0.3	0.42	0.62	0.38

$$a_3 \simeq^{\text{vdp}^{0.5,0.4}} a_5 \simeq^{\text{vdp}^{0.5,0.4}} b_3 \simeq^{\text{vdp}^{0.5,0.4}} b_4 \simeq^{\text{vdp}^{0.5,0.4}} c_3 \simeq^{\text{vdp}^{0.5,0.4}} c_5 \simeq^{\text{vdp}^{0.5,0.4}} c_7$$
$$\succ^{\text{vdp}^{0.5,0.4}}$$
$$c_1$$
$$\succ^{\text{vdp}^{0.5,0.4}}$$
$$a_1$$
$$\succ^{\text{vdp}^{0.5,0.4}}$$
$$b_1$$
$$\succ^{\text{vdp}^{0.5,0.4}}$$
$$a_2 \simeq^{\text{vdp}^{0.5,0.4}} a_4 \simeq^{\text{vdp}^{0.5,0.4}} c_2 \simeq^{\text{vdp}^{0.5,0.4}} c_4 \simeq^{\text{vdp}^{0.5,0.4}} c_6$$
$$\succ^{\text{vdp}^{0.5,0.4}}$$
$$b_2$$

With this second process, a_1 and b_1 can be distinguished. Indeed, they have two defense branches of length 2, so during the step where ε is 0, they receive the same values from their defense roots. However, one can remark that a_1 is directly attacked twice while b_1 is directly attacked once. So, during the second process where the initial scores of the attacked arguments are also propagated, a_1 receives one more negative value than b_1 ($\mathcal{V}_1^{0.5,0.4}(a_1) = 0.1 < 0.3 = \mathcal{V}_1^{0.5,0.4}(b_1)$). ∎

2.5.2 Analysis and properties

Influence of the parameters

The definition of the propagation number is based on two parameters: ε and δ. Let us characterize their roles and their impacts on the ranking computed when the variable-depth propagation is used. Recall that the parameter ε has a key role to distinguish the two phases aiming to compute the ranking between arguments. However, a concern might be that the value of ε might change the ranking obtained. We show that this is not the case:

Proposition 2.11 Let $\delta \in]0,1[$ and $\varepsilon, \varepsilon' \in]0,1]$. For any AF, we have $\text{vdp}^{\varepsilon,\delta}(\text{AF}) = \text{vdp}^{\varepsilon',\delta}(\text{AF})$.

Please note that even though different values of ε do not change the ranking between arguments returned by vdp, this parameter remains mandatory to distinguish the two steps used in the definition of vdp: the first one where non-attacked arguments are the only arguments to propagate their value in the argumentation graph ($\varepsilon = 0$) and the second one where all arguments propagate their value

($\varepsilon \neq 0$). However, this is a purely internal artifact without any effect on the outcome of the method. To make this clear, we note vdp^{δ} instead of $\mathsf{vdp}^{\varepsilon,\delta}$ to describe our parametrized ranking semantics in general.

The parameter δ is defined as the damping factor allowing us to decrease the impact of the argument when the length of the path increases. Following this, there intuitively exists a length such that the impact of arguments situating at the beginning of this path is negligible compared to the nearest arguments. Thus, the role of this parameter is to choose the scope of influence of the arguments in the argumentation framework, in addition to guaranteeing the convergence of the valuation $\mathcal{V}$. For instance, with a value of δ close to 0, only the nearest arguments (so a small part of the argumentation framework) are taken into consideration to compute the different propagation numbers, whereas with a value of δ close to 1, (almost) all the argumentation framework will be inspected. Consequently, two different values of δ can produce different rankings for the same argumentation framework. Following the principle of the fading effect, it is natural to assume that arguments located at a long distance from another argument become ineffective. In terms of design, it seems very interesting to have the ability to control this parameter to specify a maximal depth after which arguments see their influence on the value of others vanish.
To better understand how to take the fading principle into account in using δ, let us detail the algorithm used to compute the propagation numbers.

1. A positive number is assigned to each argument: $\forall a \in \mathcal{A}$, $\mathcal{V}_0^{\varepsilon,\delta}(a) = 1$ if a is non-attacked or $\mathcal{V}_0^{\varepsilon,\delta}(a) = \varepsilon$ otherwise,
2. We increase the step i by 1 and we add (or subtract) the score computed during the previous step ($\mathcal{V}_{i-1}^{\varepsilon,\delta}(a)$) and the attenuated weights (v_ε and δ^i) received from defenders (or attackers) at the beginning of a path with a length of i ($\mathcal{R}_i^S(a)$):

$$\mathcal{V}_i^{\varepsilon,\delta}(a) = \mathcal{V}_{i-1}^{\varepsilon,\delta}(a) + (-1)^i \delta^i \sum_{b \in \mathcal{R}_i^-(()a)} v_\varepsilon(b)$$

3. If, between two steps, the difference, for all valuations P, is smaller than a fixed precision threshold μ (*i.e.* $\forall a \in \mathcal{A}, |\mathcal{V}_i^{\varepsilon,\delta}(a) - \mathcal{V}_{i-1}^{\varepsilon,\delta}(a)| < \mu$) then the process is stopped[10] and the last values correspond to the propagation number of each argument. If it is not the case, we go back to 2).

Thus, given a precision threshold, one can choose δ according to the maximal expected depth.

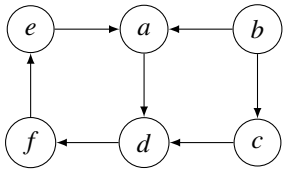
Proposition 2.12 Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, $i \in \mathbb{N}\backslash\{0\}$ be the maximal depth and μ be the precision threshold. If $\delta < \sqrt[i]{\frac{\mu}{\max_{a \in \mathcal{A}}(|\mathcal{R}_i^S(a)|)}}$ then, for all $a \in \mathcal{A}$, the sequence $\{\mathcal{V}_i^{\varepsilon,\delta}(a)\}_{i=0}^{+\infty}$ converges before step $i+1$.

Through the formula given in Proposition 2.12, we can determine, for each maximal depth, which value of δ should be used.

On the diversity of rankings

As shown in Figure 2.9 on the facing page, for a given argumentation framework, different values of δ can produce different rankings. Indeed, when $\delta \in \{0.0001, 0.2, 0.4, 0.6\}$, vdp^{δ} provides the same ranking, whereas, when $\delta \geq 0.8$, c becomes more acceptable than f and d becomes more acceptable than b.

[10] In practice, we consider that a process is stopped when, for each valuation, the difference between two steps is smaller than a precision threshold.



δ	vdp^δ
0.0001	$b \succ d \succ e \succ f \succ c \succ a$
0.2	
0.4	
0.6	
0.8	$d \succ b \succ e \succ c \succ f \succ a$
0.9	$d \succ e \succ b \succ c \succ f \succ a$

Figure 2.9: An AF and the rankings returned by vdp^δ for different values of δ

In light of these differences, one may be worried that the diversity of rankings could be so high that the semantics becomes too sensitive to small modifications of the parameter δ. To check this, we applied our variable-depth propagation on 1000 randomly generated argumentation frameworks[11] for different values of $\delta \in \{0.001, 0.2, 0.4, 0.6, 0.8, 0.9\}$. Then, we measured the dissimilarity degree between two rankings from two different values of δ using Kendall's tau coefficient (Kendall, 1938). This coefficient corresponds to the total number of rank disagreements over all unordered pairs of arguments between two rankings. It, therefore, allows us to obtain a dissimilarity degree between the two rankings.
Table 2.4 contains, for each pair of δ, the average Kendall's tau coefficient, from the results previously computed, which we multiply by 100 to obtain a percentage of dissimilarity.

δ	**0.001**	**0.2**	**0.4**	**0.6**	**0.8**	**0.9**
0.001	0	0.06	0.55	4.09	10.62	13.74
0.2	0.06	0	0.52	4.13	10.63	13.64
0.4	0.55	0.52	0	3.71	10.13	13.3
0.6	4.09	4.13	3.71	0	6.82	9.86
0.8	10.62	10.63	10.13	6.82	0	3.16
0.9	13.74	13.64	13.3	9.86	3.16	0

Table 2.4: Percentage of dissimilarity between the rankings from vdp^δ with $\delta \in \{0.001, 0.2, 0.4, 0.6, 0.8, 0.9\}$

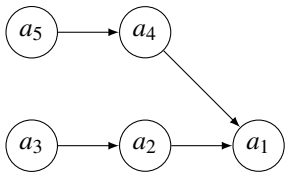
The results show that the obtained rankings stay pretty close since the biggest dissimilarity between the smallest and largest value of δ is 13.74%. This dissimilarity remains overall very small, showing that the semantics remain quite stable as the parameter varies.
The question now is whether these differences are only caused by the fading effect or if δ has an impact on other domains too.

Properties satisfied by vdp

We now investigate the properties satisfied by our variable-depth propagation semantics vdp. Before checking all properties discussed in the literature, we start by inspecting the case of Void Precedence because this property contradicts the procatalepsis principle.
One of the very distinctive features of vdp is that an attacked argument can have a better score (and so a better rank) than a non-attacked argument. Indeed, when a given argument has many defense branches, it receives many positive weights. However, as depicted in the following example, this feature is not guaranteed for all values of δ.

[11] The generation algorithms are based on the three algorithms used for producing the benchmarks of the competition ICCMA'15 Thimm and Villata (2015)

Example 2.12 Let us compute the rankings of the argumentation framework which represents the sales pitch aiming to persuade someone to buy a car used to explain the procatalepsis principle (see Example 2.9 on page 35), in using variable-depth propagation with several values of δ.



δ	vdp^δ
0.0001	$a_5 \simeq a_3 \succ a_1 \succ a_2 \simeq a_4$
0.2	
0.4	
0.6	
0.8	$a_1 \succ a_5 \simeq a_3 \succ a_2 \simeq a_4$
0.9	

Indeed, one can remark that the non-attacked argument a_3 (respectively a_5) is strictly more acceptable than each attacked argument (including a_1) when $\delta \in \{0.0001, 0.2, 0.4, 0.6\}$ but a_1 becomes strictly more acceptable than a_3 (respectively a_5) for the value of $\delta \in \{0.8, 0.9\}$. Thus, according to the choice of δ, this argument, which is attacked, can obtain a greater score than the score of non-attacked arguments.

■

Let us formally determine which are, for a given argumentation framework, the values of δ that ensure that the non-attacked arguments are more acceptable than the attacked arguments:

Proposition 2.13 Let $\text{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and $x, y \in \mathcal{A}$ such that $\mathcal{R}_1^-(x) = \emptyset$ and $\mathcal{R}_1^-(y) \neq \emptyset$. If $\delta < \delta^M$ such that $\delta^M = \sqrt{\frac{1}{\max_{z \in \mathcal{A}}(|\mathcal{R}_2^-(z)|)}}$ then $\mathcal{V}^{0,\delta}(x) > \mathcal{V}^{0,\delta}(y)$.

De facto, there exists a threshold for the parameter δ which VP is satisfied.

Corollary 2.1 For any argumentation framework, if $\delta < \delta^M$ then vdp^δ satisfies VP.

Thus, our method departs from other approaches in its treatment of the Void Precedence property, but to a certain extent only. Let us take the example of persuasion pitches to illustrate this.[12] In a persuasion pitch, a single line of defense is not enough to be more convincing than a non-attacked argument. On the other hand, when this condition is met, a simple condition for the violation of VP in persuasion pitches can be stated:

Proposition 2.14 Let $\mathcal{P}_k = \langle \mathcal{A}, \mathcal{R} \rangle$ be a persuasion pitch with $x \in \mathcal{A}$ as the targeted argument and $y \in \mathcal{A}$ be a non-attacked argument. Then,
(i) if $k = 1$ then $y \succ_{\mathcal{P}_k}^{\text{vdp}^\delta} x$;
(ii) if $k \geq 2$ and $\delta > \sqrt[m]{\frac{1}{k}}$ where m is the length of the longest defense branch of x then $x \succ_{\mathcal{P}_k}^{\text{vdp}^\delta} y$.

Interestingly, it turns out that in the context of our method, the Void Precedence property is related to the property Defense Precedence.

Proposition 2.15 If vdp^δ satisfies VP then it satisfies DP.

However please note that this is not the case in general because some ranking-based semantics satisfy VP but not DP (see Table 2.5 on the next page).
Let us now check which properties, among those defined in Section 2.2.1 on page 13, are satisfied by the variable-depth propagation semantics vdp.

Proposition 2.16 Let $\delta \in]0,1[$. vdp^δ satisfies Abs, In, Tot, NaE, +AB, AE and AvsFD. The other properties are not satisfied.

[12] Recall Definition 2.21 on page 36.

All these results are reported in Table 2.5. For comparison, we also include in this table the results of the ranking semantics presented in Section 2.4 on page 24.

Properties	Cat	Dbs	Bbs	α-Bbs	CS	Propa$_\varepsilon$	Propa$_{1+\varepsilon}$	Propa$_{1\to\varepsilon}$	Tuples	2ZG	IGD	vdp$^{\delta}$	vdp$^{\delta'}$	vdp$^{\delta''}$
Abs	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
In	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	✓
VP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	✓
DP	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	×	✓
CT	✓	✓	✓	✓	✓	×	×	×	×	×	×	×	×	×
SCT	✓	✓	✓	✓	✓	×	×	×	×	×	×	×	×	×
CP	×	✓	✓	×	×	×	×	×	×	×	×	×	×	×
QP	×	×	×	×	×	×	×	×	×	×	×	×	×	×
DDP	×	×	✓	×	×	×	✓	✓	×	×	×	×	×	×
SC	×	×	×	×	×	×	×	×	×	✓	×	×	×	×
$\oplus$DB	×	×	×	×	×	×	×	×	×	×	×	×	×	×
+DB	×	×	×	×	×	×	×	✓	✓	×	×	×	✓$_i$	✓$_i$
↑AB	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	✓$_i$	✓$_i$
↑DB	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	✓$_i$	✓$_i$
+AB	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	✓	✓	✓
Tot	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	×	✓	✓	✓
NaE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
AE	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓
OE	✓	✓	✓	✓	✓	×	×	×	✓	×	×	×	×	×
AvsFD	×	×	×	×	×	×	✓	✓	✓	✓	×	✓	✓	✓
PR	×	×	×	×	×	×	×	×	×	×	×	×	✓	×

Table 2.5: Summary of the properties satisfied by vdp ($\forall\delta$, for $max(\delta^m, \delta^M) < \delta'$ and for $\delta^m < \delta'' < \delta^M$) and some existing ranking semantics studied in the literature. A cross × means that the property is not satisfied, symbol ✓ means that the property is satisfied, and ✓$_i$ means that the i-version of the property is satisfied.

Each of these existing ranking-based semantics satisfies Void Precedence and therefore violates the principle of procatalepsis (PR).

Proposition 2.17 The ranking-based semantics Cat, Dbs, Bbs, α-Bbs, CS, Propa$_\varepsilon$, Propa$_{1+\varepsilon}$, Propa$_{1\to\varepsilon}$, Tuples, 2ZG and IGD does not satisfy PR.

We first remark that for any value of δ, vdp satisfies the properties accepted by almost all the existing ranking-based semantics (Abs, In, +AB, NaE, AE and Tot). The only exception concerns VP, but it is intended by design and discussed earlier. We can also note that vdp always satisfies property AvsFD, and for a specific δ ($\delta^m < \delta$) the property +DB. These three conditions are necessary to catch the procatalepsis principle. Indeed, AvsFD and +DB state that increasing the number of defense branches improves the acceptability of an argument, and the failure to satisfy VP is necessary to allow the attacked arguments to become more acceptable than non-attacked arguments.

2.6 Combining Extension-based Semantics and Ranking-based Semantics

As we have seen in the previous sections, there are two kinds of evaluations of arguments: at the level of sets of arguments (with extension-based or labeling-based semantics) or the level of single arguments (with ranking-based or gradual semantics). These two ways to evaluate the information encoded in an argumentation framework are interesting and target different kinds of applications. The starting point of this work is the observation that these two kinds of evaluation are in a sense orthogonal. They both can be used to extract some information about the status/strength/situation

of (sets of) arguments. Instead of seeing these approaches as mutually exclusive, one natural idea is to try to take the best of both worlds and combine them.

As already discussed in this document, extension-based semantics (Dung, 1995) are closely related to models of logic programs so they exhibit an all-or-nothing evaluation of sets of arguments. Amgoud and Ben-Naim (2013) underlines some characteristics specific to extension-based semantics (see section 2.2.1 on page 13). This kind of evaluation can be useful to define arguments from logical formulas. The *killing* and *existence* considerations seem essential to capture the fact that one attack is lethal and prevent any contradiction between arguments and thus obtain a consistent set of formulas.

However, in other applications, some of these properties can be discussed. For example, on online debate platforms, agents argue for or against a particular topic (in the form of a question or an affirmation) or other existing arguments. Often, the goal is not to find the arguments that can be accepted together but to evaluate how accepted is the topic argument. But more generally, when one faces many arguments, having a more detailed evaluation of arguments than the binary accepted/rejected obtained with extension-based semantics may be useful. Leite and Martins (2011) emphasize the limitations of classical acceptability semantics for this kind of application. In addition, to accurately represent the opinions of thousands of users, it could be more appropriate to evaluate arguments using degrees of acceptability or gradual acceptability. With ranking-based semantics, we can precisely obtain a very detailed evaluation of the strength of each argument. This can be useful for these debate platforms, but also to select the best arguments in all kinds of debates (persuasion, deliberation, etc.).

On the other hand, we can see as a drawback the fact that the evaluation of each argument is not linked at all with its acceptance status: being an argument with a good evaluation does not mean that this argument should be accepted (under extension-based semantics), and even if we define "acceptance" with respect to the ranking, there is no natural threshold to make a distinction between accepted and non-accepted argument. Defining a ranking-based semantics that is compatible with the acceptance status of an extension-based semantics would be a solution. So we propose to build this kind of semantics by refining ranking-based semantics using extension-based semantics.

Conversely, a drawback of extension-based semantics is that they do not allow a very detailed evaluation of arguments. It is for instance impossible to give a better evaluation to an unattacked argument than to all the arguments that this argument defends, whereas the acceptability of the latter depends on the acceptability of the unattacked argument. So one can use the detailed evaluation of arguments to modify extension-based semantics, for instance by selecting only the best extensions for this evaluation.[13]

2.6.1 Improving Ranking-based semantics using Extension-based semantics

Refining ranking-based semantics using acceptance status

The first idea is to constrain the rankings to be compatible with the acceptance status of the arguments. We lexicographically combine a ranking denoting the acceptance status of the arguments given by an extension-based semantics and the ranking given by a ranking-based semantics.

Definition 2.25 (Refinement of a ranking)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework. Let $\succeq^1_{\mathsf{AF}}$ and $\succeq^2_{\mathsf{AF}}$ be two rankings on $\mathcal{A}$. The (lexicographical) refinement of $\succeq^2_{\mathsf{AF}}$ by $\succeq^1_{\mathsf{AF}}$ gives a new ranking $\succeq^{1,2}_{\mathsf{AF}}$ such that $\forall x, y \in \mathcal{A}$,

$$x \succeq^{1,2}_{\mathsf{AF}} y \text{ if and only if } (x \succ^2_{\mathsf{AF}} y) \text{ or } (x \simeq^2_{\mathsf{AF}} y \text{ and } x \succeq^1_{\mathsf{AF}} y)$$

The following definition allows to build a ranking from the acceptance status given by an extension-

[13] This work was originally published in (Bonzon et al., 2018)

based semantics[14]: an argument skeptically accepted is more acceptable than an argument credulously accepted which is more acceptable than a rejected argument.

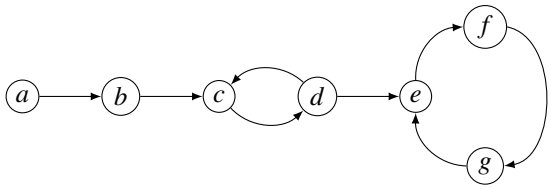
Definition 2.26 (Ranking from extension based-semantics)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and $\sigma \in \{co, pr, st, gr\}$. Let $\succeq^{\sigma}_{\mathsf{AF}}$ be a ranking on $\mathcal{A}$ such that $\forall x, y \in \mathcal{A}$, $x \succeq^{\sigma}_{\mathsf{AF}} y$ if and only if one of the following conditions is satisfied:

1. $x \in sa_{\sigma}(\mathsf{AF})$,
2. $x \in ca_{\sigma}(\mathsf{AF}) \setminus sa_{\sigma}(\mathsf{AF})$ and $y \notin sa_{\sigma}(\mathsf{AF})$,
3. $x, y \notin ca_{\sigma}(\mathsf{AF})$

Definition 2.27 (Acceptance-based ranking semantics)
Let σ_1 be a ranking-based semantics and $\sigma_2 \in \{co, pr, st, gr\}$. The **acceptance-based ranking semantics** $\mathtt{ARS}_{\sigma_1,\sigma_2}$ associates to any $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ a ranking $\succeq^{\sigma_1,\sigma_2}_{\mathsf{AF}}$ on $\mathcal{A}$ which is the refinement of $\succeq^{\sigma_2}_{\mathsf{AF}}$ by $\succeq^{\sigma_1}_{\mathsf{AF}}$.

I will focus on the categorizer-based ranking semantics to illustrate our method (see Section 2.4 on page 24).

Example 2.13 Let us first compute the arguments skeptically (sa) and credulously (ca) accepted w.r.t. the complete semantics on the following AF:



$\mathcal{E}_{co}(\mathsf{AF}) = \{\{a\}, \{a,c\}, \{a,d,f\}\}$, so $sa_{co}(\mathsf{AF}) = \{a\}$ and $ca_{co}(\mathsf{AF}) = \{a,c,d,f\}$. We obtain thus : $a \succ^{co}_{\mathsf{AF}} c \simeq^{co}_{\mathsf{AF}} d \simeq^{co}_{\mathsf{AF}} f \succ^{co}_{\mathsf{AF}} b \simeq^{co}_{\mathsf{AF}} e \simeq^{co}_{\mathsf{AF}} g$
The ranking returned by the categorizer-based ranking semantics is the following:
$a \succ^{\mathrm{Cat}}_{\mathsf{AF}} f \succ^{\mathrm{Cat}}_{\mathsf{AF}} d \succ^{\mathrm{Cat}}_{\mathsf{AF}} g \succ^{\mathrm{Cat}}_{\mathsf{AF}} b \succ^{\mathrm{Cat}}_{\mathsf{AF}} c \succ^{\mathrm{Cat}}_{\mathsf{AF}} e$.
Thus, when we combine the two rankings, the refinement-based ranking semantics returns the following ranking: $a \succ^{\mathrm{Cat},co}_{\mathsf{AF}} f \succ^{\mathrm{Cat},co}_{\mathsf{AF}} d \succ^{\mathrm{Cat},co}_{\mathsf{AF}} c \succ^{\mathrm{Cat},co}_{\mathsf{AF}} g \succ^{\mathrm{Cat},co}_{\mathsf{AF}} b \succ^{\mathrm{Cat},co}_{\mathsf{AF}} e$

■

One can see in this example that g has quite a good evaluation for the ranking-based semantics $\succ^{\mathrm{Cat}}_{\mathsf{AF}}$, whereas it is a rejected argument. In particular, it has a better evaluation than c that is credulously accepted (for the complete semantics). The combined ranking-based semantics $\succ^{\mathrm{Cat},co}$ allows to force c to be better than g.

So now the question is to know whether these modifications change the "rationality" of the ranking-based semantics, i.e. do these combined semantics satisfy less logical properties than the original ranking-based semantics?

Proposition 2.18 Let σ_1 be a ranking-based semantics and $\sigma_2 \in \{co, pr, st, gr\}$. Let α be any property among Abs, In, VP, DP, DDP, SC, $\oplus$DB, +DB, +AB, $\uparrow$AB, $\uparrow$DB, Tot, NaE.
If σ_1 satisfies the property α, then the semantics $\mathtt{ARS}_{\sigma_1,\sigma_2}$ satisfies the property α. The semantics $\mathtt{ARS}_{\sigma_1,gr}$ and $\mathtt{ARS}_{\sigma_1,st}$ satisfy QP, CT and SCT. The semantics $\mathtt{ARS}_{\sigma_1,\sigma_2}$ satisfies the property AvsFD and does not satisfy CP.

[14]Please note that, a priori, any extension-based semantics can be used in our method but for our properties, we choose to only focus on the four classical semantics (see Definition 2.4 on page 10).

It is interesting to note that, except for AvsFD and CP, the semantics satisfies the property if the original ranking-based semantics satisfies the properties. Thus, the compliance of the ranking-based semantics for these properties is preserved using the refinement with the extension-based semantics. Better than that, it allows the enforcement of AvsFD that few semantics satisfy (Section 2.4.2 on page 30). So it is an easy way to obtain new semantics satisfying AvsFD from standard semantics from the literature.

Refining ranking-based semantics using justification status

Instead of focusing on the acceptability status of the arguments, we can also build a ranking from the labeling-based justification status of the arguments, that offers a more fined-gained distinction of the arguments with respect to the labelings/extensions. However, the definition from Wu and Caminada (2010) (see Definition 2.9 on page 12) only concerns the complete semantics. It is why we propose to extend the definition to Dung's semantics.

Definition 2.28 (Extended justification status)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, $\sigma \in \{co, pr, st, gr\}$ and $x \in \mathcal{A}$. The extended justification status of x is the outcome yielded by the function $\mathcal{JS} : \mathcal{A} \rightarrow 2^{\{\texttt{in},\texttt{out},\texttt{undec}\}}$ s.t. $\mathcal{JS}_\sigma(x) = \{\mathcal{L}_\sigma(x) \mid \mathcal{L} \in \mathcal{L}_\sigma(\langle \mathcal{A}, \mathcal{R} \rangle)\}$.

In addition to the 6 statuses $\{\texttt{in}\}$, $\{\texttt{out}\}$, $\{\texttt{undec}\}$, $\{\texttt{in},\texttt{undec}\}$, $\{\texttt{out},\texttt{undec}\}$ and $\{\texttt{in},\texttt{out},\texttt{undec}\}$, we must add the status $\{\texttt{in},\texttt{out}\}$, that could not appear for the complete semantics, but which may be obtained, for instance with the preferred semantics.
With the graph depicted in Figure 2.10, we include the status $\{\texttt{in},\texttt{out}\}$ in the hierarchy of the justification statuses.

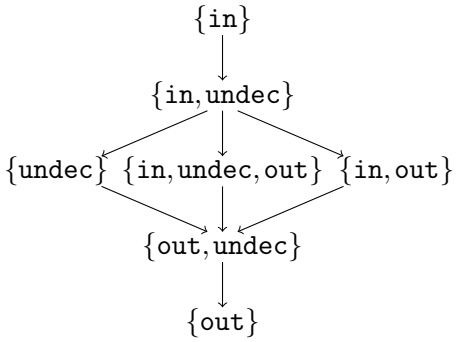


Figure 2.10: The hierarchy of the extended justification statuses

Thus, we can classify the statuses with the following ranking: $\{\texttt{in}\} \succ_{js} \{\texttt{in},\texttt{undec}\} \succ_{js} \{\texttt{undec}\} \simeq \{\texttt{in},\texttt{out},\texttt{undec}\} \simeq \{\texttt{in},\texttt{out}\} \succ_{js} \{\texttt{out},\texttt{undec}\} \succ_{js} \{\texttt{out}\}$. According to this classification, we can say that an argument is more acceptable than another one if it has a better status.

Definition 2.29 (Justification-based ranking)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and $\sigma \in \{co, pr, st, gr\}$. Let $\succeq_{\mathsf{AF}}^{\mathcal{JS}_\sigma}$ be a ranking on $\mathcal{A}$ such that $\forall x, y \in \mathcal{A}$,

$$x \succeq_{\mathsf{AF}}^{\mathcal{JS}_\sigma} y \text{ if and only if } \mathcal{JS}_\sigma(x) \succeq_{js} \mathcal{JS}_\sigma(y)$$

Definition 2.30 (Justification-based ranking semantics) Let σ_1 be a ranking-based semantics and $\sigma \in \{co, pr, st, gr\}$. The **justification-based ranking semantics** $\mathtt{JRS}_{\sigma_1,\sigma}$ associates to any $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ a ranking $\succeq_{\mathsf{AF}}^{\sigma_1,\mathcal{JS}_\sigma}$ on $\mathcal{A}$ which is the refinement of $\succeq_{\mathsf{AF}}^{\mathcal{JS}_\sigma}$ by $\succeq_{\mathsf{AF}}^{\sigma_1}$.

Example 2.13 **(continuing from p. 47)** This argumentation framework depicted has three complete labelings $\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_3$ with:
$\mathtt{in}(\mathcal{L}_1) = \{a\}, \mathtt{out}(\mathcal{L}_1) = \{b\}, \mathtt{undec}(\mathcal{L}_1) = \{c,d,e,f,g\}$
$\mathtt{in}(\mathcal{L}_2) = \{a,c\}, \mathtt{out}(\mathcal{L}_2) = \{b,d\}, \mathtt{undec}(\mathcal{L}_2) = \{e,f,g\}$
$\mathtt{in}(\mathcal{L}_3) = \{a,d,f\}, \mathtt{out}(\mathcal{L}_3) = \{b,c,e,g\}, \mathtt{undec}(\mathcal{L}_3) = \emptyset$
From these different complete labelings, the labeling-based justification statuses of each argument in the AF is $\mathcal{JS}_{co}(a) = \{in\}$, $\mathcal{JS}_{co}(b) = \{out\}$, $\mathcal{JS}_{co}(c) = \mathcal{JS}_{co}(d) = \{in, out, undec\}$, $\mathcal{JS}_{co}(e) = \mathcal{JS}_{co}(g) = \{undec, out\}$ and $\mathcal{JS}_{co}(f) = \{in, undec\}$. So we obtain the following ranking:

$$a \succ_{\mathsf{AF}}^{\mathrm{JS}_{co}} f \succ_{\mathsf{AF}}^{\mathrm{JS}_{co}} c \simeq_{\mathsf{AF}}^{\mathrm{JS}_{co}} d \succ_{\mathsf{AF}}^{\mathrm{JS}_{co}} e \simeq_{\mathsf{AF}}^{\mathrm{JS}_{co}} g \succ_{\mathsf{AF}}^{\mathrm{JS}_{co}} b$$

Combined with the ranking returned by the categorizer-based ranking semantics, we obtain the following ranking:

$$a \succ_{\mathsf{AF}}^{\mathrm{Cat,JS}_{co}} f \succ_{\mathsf{AF}}^{\mathrm{Cat,JS}_{co}} d \succ_{\mathsf{AF}}^{\mathrm{Cat,JS}_{co}} c \succ_{\mathsf{AF}}^{\mathrm{Cat,JS}_{co}} g \succ_{\mathsf{AF}}^{\mathrm{Cat,JS}_{co}} e \succ_{\mathsf{AF}}^{\mathrm{Cat,JS}_{co}} b$$

■

Proposition 2.19 Let σ_1 be a ranking-based semantics and $\sigma_2 \in \{co, pr, st, gr\}$. Let α be any property among Abs, In, VP, DP, DDP, $\oplus$DB, +DB, +AB, $\uparrow$AB, $\uparrow$DB, Tot, NaE.
If σ_1 satisfies the property α, then the semantics $\mathtt{ARS}_{\sigma_1,\mathrm{JS}_{\sigma_2}}$ satisfies the property α. The semantics $\mathtt{ARS}_{\sigma_1,\mathrm{JS}_{gr}}$ and $\mathtt{ARS}_{\sigma_1,\mathrm{JS}_{st}}$ satisfy QP, CT and SCT. The semantics $\mathtt{ARS}_{\sigma_1,\mathrm{JS}_{\sigma_2}}$ satisfies the property AvsFD and does not satisfy CP, SC.

One can remark that the difference of properties satisfied between this semantics and the previous one is minor. Indeed, the only difference concerns the Self-Contradiction (SC) property and can be explained by the fact that an argument that attacks itself is labeled `undec` if it is not attacked by other arguments. Thus, this argument is more acceptable than an argument directly attacked by a non-attacked argument while the skeptical and credulous inference functions always consider the two arguments as rejected.

2.6.2 Refining Propagation semantics using acceptance and justification status

We proposed to adapt the propagation principle introduced in Section 2.3 on page 19 by using acceptance status and justification status to allow a more fine-grained initial evaluation.
Recall that the first step of the propagation principle is ensured by the valuation function v_ε, which takes, for the propagation semantics, only two values (1 is assigned to the non-attacked arguments and ε for the attacked arguments). The idea is to propose more complex functions to assign values to arguments at the beginning of the propagation. Let us first define a valuation function that takes into account the level of acceptability of arguments.

Definition 2.31 (Valuation function based on acceptance)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and $\vec{z}_\sigma = \langle \alpha, \beta, \gamma, \delta \rangle$ be a vector of real number linked to the semantics $\sigma \in \{co, pr, st, gr\}$, with $1 \geq \alpha > \beta > \gamma > \delta \geq 0$. The valuation

function $v_{\vec{z}_\sigma} : \mathcal{A} \to [0,1]$ is defined as $\forall x \in \mathcal{A}$,

$$v_{\vec{z}_\sigma}(x) = \begin{cases} \alpha & \text{if } \mathcal{R}_1^-(x) = \emptyset \\ \beta & \text{if } x \in sa_\sigma(\mathsf{AF}) \text{ and } \mathcal{R}_1^-(x) \neq \emptyset \\ \gamma & \text{if } x \in ca_\sigma(\mathsf{AF}) \setminus sa_\sigma(\mathsf{AF}) \\ \delta & \text{if } x \notin ca_\sigma(\mathsf{AF}) \end{cases}$$

So a preference is given to non-attacked arguments, then to skeptically accepted arguments, then to credulously accepted ones, and the worst ones are rejected arguments.

Example 2.13 (continuing from p. 47) Applying the complete semantics on the AF depicted in this example allows to say that a is the only argument which is not attacked while c, d and f are credulously (and not skeptically) accepted and b, e and g are considered as rejected.
With $\vec{z}_{co} = \langle 1, 0.7, 0.3, 0 \rangle$, we obtain the following table which sums up the valuation of each argument at each step:

$\mathcal{V}_i^{\vec{z}_{co}}$	a	b	c	d	e	f	g
0	1	0	0.3	0.3	0	0.3	0
1	1	-1	0	0	-0.3	0.3	-0.3
2	1	-1	1.3	0.3	0.3	0.6	-0.3

When we lexicographically compare the propagation vector of each argument, we obtain the following ranking:

$$a \succ_{\mathsf{AF}}^{\mathcal{V}^{\vec{z}_{co}}} f \succ_{\mathsf{AF}}^{\mathcal{V}^{\vec{z}_{co}}} c \succ_{\mathsf{AF}}^{\mathcal{V}^{\vec{z}_{co}}} d \succ_{\mathsf{AF}}^{\mathcal{V}^{\vec{z}_{co}}} e \succ_{\mathsf{AF}}^{\mathcal{V}^{\vec{z}_{co}}} g \succ_{\mathsf{AF}}^{\mathcal{V}^{\vec{z}_{co}}} b$$

■

Let us check which properties are satisfied:

Proposition 2.20 Let $\sigma \in \{co, pr, st, gr\}$. The semantics $\text{Propa}^{\vec{z}_\sigma}$ satisfies Abs, In, VP, DP, ↑AB, ↑DB, +AB, Tot, NaE and AvsFD. The other properties are not satisfied.

The set of properties satisfied by $\text{Propa}^{\vec{z}_\sigma}$ is close to the ones satisfied by the semantics Propa_ε. The differences come from the AvsFD property that is satisfied by $\text{Propa}^{\vec{z}_\sigma}$ thanks to the distinction done between attacked arguments in the initial evaluation, and from the fact that SCT and CT are not satisfied.

We also define another valuation function by considering the extended justification status.

Definition 2.32 (Valuation function based on justification)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and $\vec{\text{js}}_\sigma = \langle \alpha, \beta, \gamma, \delta, \varepsilon, \omega \rangle$ be a vector of real number linked to the semantics $\sigma \in \{co, pr, st, gr\}$, with $1 \geq \alpha > \beta > \gamma > \delta > \varepsilon > \omega \geq 0$.
The valuation function $v_{\vec{\text{js}}_\sigma} : \mathcal{A} \to [0,1]$ is defined as $\forall x \in \mathcal{A}$,

$$v_{\vec{\text{js}}_\sigma}(x) = \begin{cases} \alpha & \text{if } \mathcal{R}_1^-(x) = \emptyset \\ \beta & \text{if } \mathcal{JS}_\sigma(x) = \{in\} \text{ and } \mathcal{R}_1^-(x) \neq \emptyset \\ \gamma & \text{if } \mathcal{JS}_\sigma(x) = \{in, undec\} \\ \delta & \text{if } \mathcal{JS}_\sigma(x) \in \{\{undec\}, \{in, out\}, \{in, undec, out\}\} \\ \varepsilon & \text{if } \mathcal{JS}_\sigma(x) = \{undec, out\} \\ \omega & \text{if } \mathcal{JS}_\sigma(x) = \{out\} \end{cases}$$

Example 2.13 **(continuing from p. 47)** Let us recall the justification status labeling of each argument: $\mathcal{JS}_{co}(a) = \{in\}$, $\mathcal{JS}_{co}(b) = \{out\}$, $\mathcal{JS}_{co}(c) = \mathcal{JS}_{co}(d) = \{in, out, undec\}$, $\mathcal{JS}_{co}(e) = \mathcal{JS}_{co}(g) = \{undec, out\}$ and $\mathcal{JS}_{co}(f) = \{in, undec\}$. So, with $\vec{\jmath s}_{co} = \langle 1, 0.8, 0.6, 0.4, 0.2, 0\rangle$, we have:

$\mathcal{V}_i^{\vec{\jmath s}_{co}}$	a	b	c	d	e	f	g
0	1	0	0.4	0.4	0.2	0.6	0.2
1	1	-1	0	0	-0.4	0.4	-0.4
2	1	-1	1.4	0.4	0.6	1	-0.2

And, consequently, we obtain the following ranking:

$$a \succ^{\mathcal{V}}_{\mathsf{AF}} {}^{\vec{\jmath s}_{co}} f \succ^{\mathcal{V}}_{\mathsf{AF}} {}^{\vec{\jmath s}_{co}} c \succ^{\mathcal{V}}_{\mathsf{AF}} {}^{\vec{\jmath s}_{co}} d \succ^{\mathcal{V}}_{\mathsf{AF}} {}^{\vec{\jmath s}_{co}} e \succ^{\mathcal{V}}_{\mathsf{AF}} {}^{\vec{\jmath s}_{co}} g \succ^{\mathcal{V}}_{\mathsf{AF}} {}^{\vec{\jmath s}_{co}} b$$

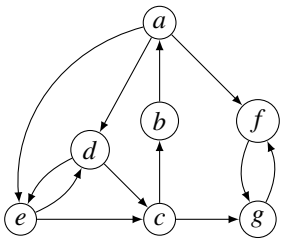
■

As shown in the following proposition, increasing the number of distinctions between the attacked argument does not change the properties satisfied by the propagation semantics.

Proposition 2.21 Let $\sigma \in \{co, pr, st, gr\}$. The semantics $\text{Propa}^{\vec{\jmath s}_\sigma}$ satisfies Abs, In, VP, DP, ↑AB, ↑DB, +AB, Tot, NaE and AvsFD. The other properties are not satisfied.

2.6.3 Improving Extension-based using Ranking-based semantics

In this section, we propose three different ways to use ranking-based semantics to modify the results obtained by extension-based semantics. The first two methods aim to decrease the number of extensions, to allow more inferences, thanks to ranking-based semantics. The last method disregards the attacks that come from bad arguments with respect to the ranking-based semantics.



$$\mathcal{E}_{gr}(\mathsf{AF}) = \{\}$$
$$\mathcal{E}_{pr}(\mathsf{AF}) = \{\{a,c\},\{b,d,f\},\{b,e,f\},\{b,d,g\},\{b,e,g\}\}$$
$$\mathcal{E}_{st}(\mathsf{AF}) = \{\{a,c\},\{b,d,f\},\{b,e,f\},\{b,d,g\},\{b,e,g\}\}$$
$$\mathcal{E}_{co}(\mathsf{AF}) = \{\emptyset,\{a,c\},\{b,d,f\},\{b,e,f\},\{b,d,g\},\{b,e,g\}\}$$
$$Cat(\mathsf{AF}) = b \succ^{\text{Cat}}_{\mathsf{AF}} a \succ^{\text{Cat}}_{\mathsf{AF}} c \succ^{\text{Cat}}_{\mathsf{AF}} g \succ^{\text{Cat}}_{\mathsf{AF}} d \simeq^{\text{Cat}}_{\mathsf{AF}} e \succ^{\text{Cat}}_{\mathsf{AF}} f$$

Figure 2.11: An AF with many extensions

As shown with the argumentation framework depicted in Figure 2.11, when many cycles with even lengths exist, extension-based semantics return several extensions (except for the grounded semantics which always return a unique extension). As discussed in Konieczny et al. (2015), in this case, selecting the arguments skeptically and credulously accepted can be problematic. Indeed, if there are many extensions, using skeptical inference can give almost no information and the credulous inference can give too many arguments. There exist works where additional information (*e.g.* weight on the attacks, preferences) are used to reduce the number of extensions in

a given argumentation framework (*e.g.* (Amgoud and Vesic, 2014; Coste-Marquis et al., 2012)). However, our goal is to reduce the set of extensions without any additional information in the argumentation framework. While in Konieczny et al. (2015), the attack relation is taken into account to discriminate some extensions, we propose here to consider the ranking returned by ranking-based semantics to select the "best" extensions. For this purpose, we propose two approaches.

Select the best extensions: comparing the arguments' ranks

The first criterion we consider is the rank that an argument has in a ranking of arguments returned by ranking-based semantics.

Definition 2.33 (Rank)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework. Given a ranking-based semantics σ, the rank of $x \in \mathcal{A}$ w.r.t. $\succeq^{\sigma}_{\mathsf{AF}}$, denoted by $r_\sigma(x)$, is the level in which it belongs in the ordered sequence of equivalence classes of $\mathcal{A}$ with respect to $\succeq^{\sigma}_{\mathsf{AF}}$. So $r_\sigma(x) = i$ where i is the longest path $x_1 \succ^{\sigma}_{\mathsf{AF}} \ldots \succ^{\sigma}_{\mathsf{AF}} x_i \succ^{\sigma}_{\mathsf{AF}} x$; and $r_\sigma(x) = 0$ if $\nexists y \in \mathcal{A}$ s.t. $y \succ^{\sigma}_{\mathsf{AF}} x$.

Example 2.14 If we consider the ranking returned by the categorizer semantics on the AF depicted in Figure 2.11 on the previous page, the rank of each argument is $r_{Cat}(a) = 1$, $r_{Cat}(b) = 0$, $r_{Cat}(c) = 2$, $r_{Cat}(d) = 4$, $r_{Cat}(e) = 4$, $r_{Cat}(f) = 5$ and $r_{Cat}(g) = 3$. ∎

Given a ranking-based semantics, the rank multiset of an extension obtained from an extension-based semantics is composed of the rank of each of its arguments.

Definition 2.34 (Rank multiset) Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, σ_1 be an extension-based semantics and σ_2 be a ranking-based semantics. For an extension $\mathcal{E} \in \mathcal{E}_{\sigma_1}(\mathsf{AF})$, with $\mathcal{E} = \{x_1, \ldots, x_n\}$, we define its rank multiset as $rv_{\sigma_2}(\mathcal{E}) = (r_{\sigma_2}(x_1), \ldots, r_{\sigma_2}(x_n))$.

Example 2.14 **(continuing from p. 52)** Focusing on the preferred extension $\{b, d, f\}$ and the categoriser-based semantics, we have $rv_{Cat}(\{b, d, f\}) = (0, 4, 5)$. ∎

We use an aggregation function to aggregate the values belonging to the same rank multiset.

Definition 2.35 (Aggregation function)
We say that $\oplus$ is an aggregation function if for every $n \in \mathbb{N}$, $\oplus$ is a mapping from $\mathbb{N}^n$ to $\mathbb{N}$ such that:

- if $x_i \geq x'_i$, then $\oplus(x_1, \ldots, x_i, \ldots, x_n) \geq \oplus(x_1, \ldots, x'_i, \ldots, x_n)$
- $\oplus(x_1, \ldots, x_n) = 0$ iff for every i, $x_i = 0$
- $\oplus(x) = x$.

Many typical examples of aggregation functions exist such as sum, max, min, leximax, leximin, etc. The goal is now to compare the score assigned to each extension to select the best ones with respect to the chosen criterion.

Definition 2.36 (Rank-based extensions)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, σ_1 be an extension-based semantics, σ_2 be a ranking-based semantics and $\oplus$ be an aggregation function.
The set of rank-based extensions (RBE) is defined as $\mathtt{RBE}^{\oplus}_{\sigma_1,\sigma_2}(\mathsf{AF}) = \underset{\mathcal{E} \in \mathcal{E}_{\sigma_1}(\mathsf{AF})}{\operatorname{argmin}} \oplus(rv_{\sigma_2}(\mathcal{E}))$

The resulting set of extensions depends on the chosen aggregation function. Indeed, using the

average favors the extensions with few arguments but which have a good rank (even if these are not the best ranks) while when the leximin is used, the number of arguments has no impact because the rank of the best argument in each extension is first compared and in case of a tie for some extensions, we compare the second best rank and so on.

Proposition 2.22 For every $\oplus$, for every semantics σ_1 and σ_2, for every $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$, for every $x \in \mathcal{A}$,

- $\mathtt{RBE}^{\oplus}_{\sigma_1,\sigma_2}(\mathsf{AF}) \subseteq \mathcal{E}_{\sigma_1}(\mathsf{AF})$
- $sa_{\sigma_1}(\mathsf{AF}) \subseteq \bigcup_{\mathcal{E} \in \mathtt{RBE}^{\oplus}_{\sigma_1,\sigma_2}(\mathsf{AF})} \mathcal{E} \subseteq ca_{\sigma_1}(\mathsf{AF})$

Baroni and Giacomin (2007) pointed out a set of properties for extension-based argumentation semantics: I-maximality, Admissibility, Strong Admissibility, Reinstatement, Weak Reinstatement and CF-Reinstatement.

Proposition 2.23 Let $\oplus$ be an aggregation function and σ_2 be a ranking-based semantics. Let α be any property among I-maximality, Admissibility, Strong Admissibility, Reinstatement, Weak Reinstatement and CF-Reinstatement.
If the semantics σ_1 satisfies the property α, then the semantics $\mathtt{RBE}^{\oplus}_{\sigma_1,\sigma_2}$ satisfies the property α.

So RBE satisfies the same properties as the underlying extension-based (or labeling-based) semantics they are built from, except for the directionality property, just like in Konieczny et al. (2015).

Select the best extensions: pairwise comparison

Our second approach consists in comparing all pairs of extensions based on the number of arguments in one extension which are more acceptable than the arguments in another extension.

Definition 2.37 (Acceptability number)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework. Let σ_1 be an extension-based semantics, σ_2 be a ranking-based semantics and $\mathcal{E}, \mathcal{E}' \in \mathcal{E}_{\sigma_1}(\mathsf{AF})$. We have

$$\mathcal{N}_{\sigma_2}(\mathcal{E}, \mathcal{E}') = |\{(x,y) \text{ s.t. } x \succ^{\sigma_2}_{\mathsf{AF}} y \text{ with } x \in \mathcal{E} \text{ and } y \in \mathcal{E}'\}|$$

Example 2.15 Let us consider the set of extensions returned by the preferred semantics and the ranking returned by the categorizer-based semantics on the AF depicted in Figure 2.11 on page 51. For example, $\mathcal{N}_{Cat}(\mathcal{E}_1, \mathcal{E}_4) = 4$ because $c \succ^{\text{Cat}} d$, $c \succ^{\text{Cat}} g$, $a \succ^{\text{Cat}} d$ and $a \succ^{\text{Cat}} g$. So, following the same reasoning, when we compare all the extensions, we obtain the following table:

$\mathcal{N}_{Cat}$	$\mathcal{E}_1$	$\mathcal{E}_2$	$\mathcal{E}_3$	$\mathcal{E}_4$	$\mathcal{E}_5$
$\mathcal{E}_1 = \{a,c\}$	$\times$	4	4	4	4
$\mathcal{E}_2 = \{b,d,f\}$	2	$\times$	0	0	0
$\mathcal{E}_3 = \{b,e,f\}$	2	0	$\times$	0	0
$\mathcal{E}_4 = \{b,d,g\}$	2	1	1	$\times$	0
$\mathcal{E}_5 = \{b,e,g\}$	2	1	1	0	$\times$

■

The approach consists in counting how many extensions are defeated by a given extension and selecting the extension(s) that obtain the best score.

Definition 2.38 (Acceptability-based extensions)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework, σ_1 be an extension-based semantics and σ_2 be a ranking-based semantics. The set of acceptability-based extensions (ABE) is defined as follows:

$$\mathtt{ABE}_{\sigma_1,\sigma_2}(\mathsf{AF}) = \underset{\mathcal{E} \in \mathcal{E}_{\sigma_1}(\mathsf{AF})}{\operatorname{argmax}} |\{\mathcal{E}' \in \mathcal{E}_{\sigma_1}(\mathsf{AF}) : \mathcal{N}_{\sigma_2}(\mathcal{E}, \mathcal{E}') > \mathcal{N}_{\sigma_2}(\mathcal{E}', \mathcal{E})\}|$$

Example 2.15 **(continuing from p. 53)** We can see that the extension $\mathcal{E}_1$ defeats all the other extensions, so, $\mathtt{ABE}_{pr,Cat}(\mathsf{AF}) = \{\mathcal{E}_1\}$. ∎

Proposition 2.24 For every semantics σ_1 and σ_2, for every $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$, for every $x \in \mathcal{A}$,

- $\mathtt{ABE}_{\sigma_1,\sigma_2}(\mathsf{AF}) \subseteq \mathcal{E}_{\sigma_1}(\mathsf{AF})$
- $sa_{\sigma_1}(\mathsf{AF}) \subseteq \bigcup_{\mathcal{E} \in \mathtt{ABE}_{\sigma_1,\sigma_2}(\mathsf{AF})} \mathcal{E} \subseteq ca_{\sigma_1}(\mathsf{AF})$

Although the obtained semantics are different from the RBE semantics, ABE semantics exhibit the same properties.

Proposition 2.25 Let σ_2 be a ranking-based semantics. Let α be any property among I-maximality, Admissibility, Strong Admissibility, Reinstatement, Weak Reinstatement and CF-Reinstatement.
If the semantics σ_1 satisfies the property α, then the semantics $\mathtt{ABE}_{\sigma_1,\sigma_2}$ satisfies the property α.

Removing attacks

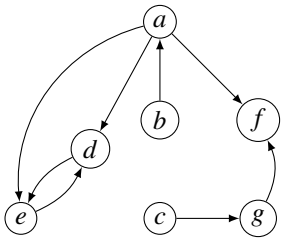
As a last possible modification of extension-based semantics with ranking based-semantics we will look at a more drastic modification, where we put more emphasis on the ranking-based semantics. The idea here is to give strong priority to strong arguments with respect to the ranking-based semantics, by not considering attacks from weaker arguments to stronger ones. So we will use the preference-based argumentation framework of Amgoud and Cayrol (2002a), by considering the ranking obtained by the ranking-based semantics as the preference relation between arguments. Amgoud and Cayrol (2002a) redefine the attack relation in saying that an argument x **defeats** an argument y if and only if there exists an attack from x to y and y is not preferred to x with respect to the preference relation.

Definition 2.39 (Ranking-based argumentation framework)
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework and σ be a ranking-based semantics. An AF_σ is a triplet $\langle \mathcal{A}', \mathcal{R}', \succeq^{\sigma}_{\mathsf{AF}} \rangle$ where:

- $\mathcal{A}' = \mathcal{A}$
- $\mathcal{R}' = \{(a,b) \mid (a,b) \in \mathcal{R} \text{ and } a \succeq^{\sigma}_{\mathsf{AF}} b\}$
- $\succeq^{\sigma}_{\mathsf{AF}}$ is the ranking on $\mathcal{A}$ returned by the ranking-based semantics σ from AF

The acceptability of the arguments is then defined in the standard way from this new argumentation framework.

Example 2.16 Let us focus on the argumentation framework depicted in Figure 2.11 on page 51 and on the categorizer-based ranking semantics. Following the definition, we must remove the attacks (c,b) (because $b \succ^{\text{Cat}}_{\mathsf{AF}} c$), (d,c) (because $c \succ^{\text{Cat}}_{\mathsf{AF}} d$), (e,c) (because $c \succ^{\text{Cat}}_{\mathsf{AF}} e$) and (f,g) (because $g \succ^{\text{Cat}}_{\mathsf{AF}} f$). Thus, we obtain the following AF_{Cat} and its extensions:



$\mathcal{E}_{gr}(\langle \mathcal{A}, \mathcal{R} \rangle_{Cat}) = \{b, c, f\}$
$\mathcal{E}_{pr}(\langle \mathcal{A}, \mathcal{R} \rangle_{Cat}) = \{\{b, c, e, f\}, \{b, c, d, f\}\}$
$\mathcal{E}_{st}(\langle \mathcal{A}, \mathcal{R} \rangle_{Cat}) = \{\{b, c, e, f\}, \{b, c, d, f\}\}$
$\mathcal{E}_{co}(\langle \mathcal{A}, \mathcal{R} \rangle_{Cat}) = \{\{b, c, f\}, \{b, c, e, f\}, \{b, c, d, f\}\}$ ∎

One can remark that the computed extensions are not necessarily conflict-free with respect to the original AF due to the removal of some attacks. That is perfectly natural since we consider that these attacks are not legitimate ones, so conflict-freeness has to be considered with respect to the defeat relation, not the attack one.

But, in the case where one wants a conflict-free result with respect to the original argumentation framework, we propose now a method to "rationalize" these extensions by extracting conflict-free subsets. For this purpose, for a given set of (potentially not conflict-free) arguments, we select the subsets of arguments (maximal w.r.t. $\subseteq$) which are conflict-free and which respect the constraint saying that when a conflict exists between two arguments, the most acceptable argument (w.r.t. the ranking-based semantics used to define AF_σ) is selected.

2.7 Similarity between Arguments

Another question of interest concerning semantics in argumentation theory concerns the notion of similarity between arguments. Similarity is related to commonality, in that the more commonality two arguments share, the more similar they are. Similar arguments appear frequently either in texts or debates. While the importance of detecting similar arguments (*paraphrases*, or *rephrased* arguments) has been identified as an important challenge in argument mining (Konat et al., 2016; Stein, 2016), this relation is always taken in a strong sense (*"two text units are in the relation of rephrase when substitution of one unit for another preserves the argument structure"* (Konat et al., 2016)). When such a relation exists, rephrased arguments can simply be considered as a single argument for the evaluation phase. The same holds for the equivalence of arguments studied in (Amgoud et al., 2014; Wooldridge et al., 2006).

There is however to the best of our knowledge no work investigating more generally the impact of partially similar arguments at the level of their global evaluation. Consider the example of Figure 2.12 on the following page, extracted from a debate held on the `arguman` platform.[15]

Arguments *a* and *c* are similar (in short, they take a probabilistic frequentist scheme). However, both also add subtle elements: *c* mentions that the number of people currently alive is negligible wrt. the number of people who died in history. On the other hand, argument *a* mentions (*"until dramatic medical discoveries [...] are made..."*). That is, while they certainly share some elements and should not be simply considered as two arguments, it would also be too simplistic to merely regard them as a single argument resulting from the merging of these individual arguments. One obvious problem with this solution would be that a counter-attack against a sub-part of the argument would transfer to the whole argument. In our example, *d* attacks the statement made more precise in *c* that the probability becomes negligible, but it would not attack the statement of *a* (*"we should not expect..."*).

Our research question is thus: *Can we design gradual semantics taking into account the degree of similarity among arguments?*

[15] `http://en.arguman.org/`

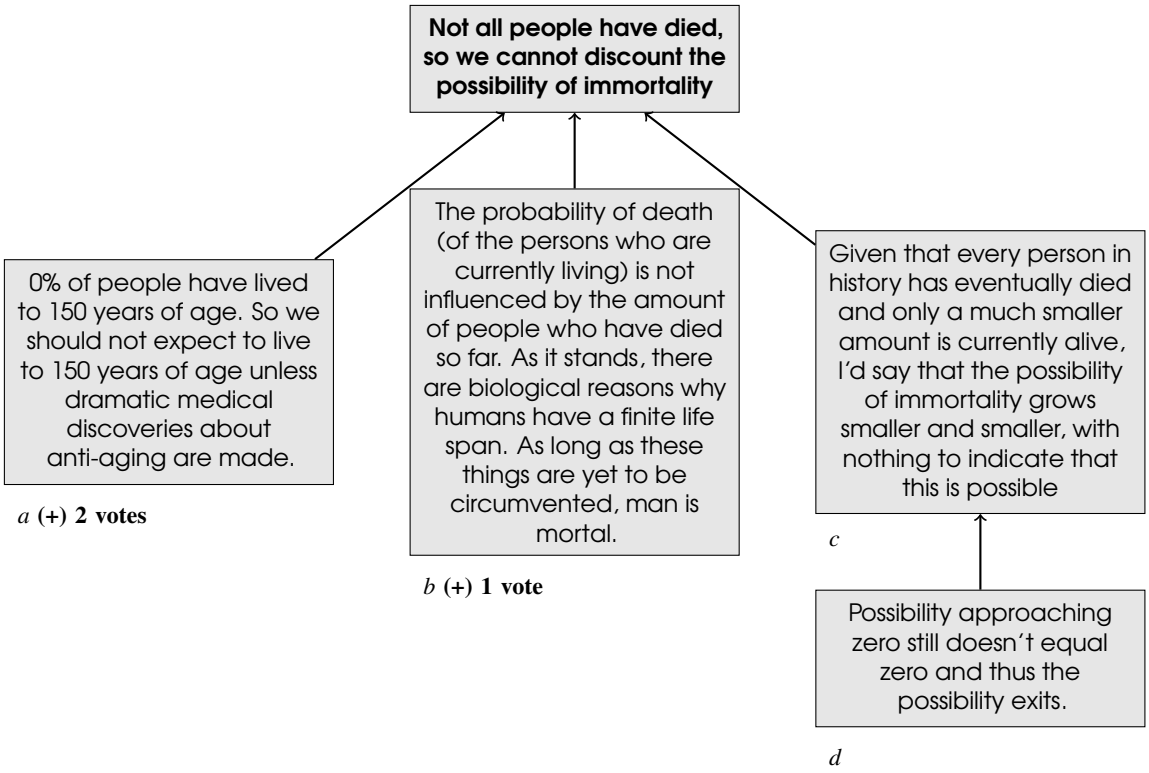


Figure 2.12: Excerpt of an Argüman debate.

First and foremost, it is important to stress that our perspective is *normative*. Surely, we do not deny that repeating the same argument, again and again, can be, in some contexts, an efficient strategy in a debate. However, ideally, this should not be the case. For instance, in the context of online debates, the objective of the owner of the platform may precisely be to choose an evaluation method resistant to some sockpuppet behavior (Kumar et al., 2017) consisting in repeating similar arguments under false alternative identities.
The basic idea of our approach is thus that the overall strengths of an argument's attackers should not be fully considered if the attackers are similar. Similarity is usually simply defined among *pairs* of entities. In the context of argumentation though, we shall defend the view that it sometimes makes sense to consider more generally the similarity between an individual argument and a full *set* of other arguments. This does not rule out the use of classical pairwise measures, and indeed some of our methods simply require an input pairwise similarity.

2.7.1 Weighted Argumentation Graph with Similarity

Similarity has been studied in many domains, like recognition, classification and clustering, where it is necessary to compare two objects. For that purpose, numerous similarity measures were defined (see (Lesot et al., 2009) for a survey). They are generally suited for objects of the same size and are defined as functions assigning to any pair of objects a value in the unit interval $[0, 1]$. The greater the value, the more similar the objects. In the context of argumentation, it will sometimes be useful to define the similarity of an argument with respect to a set of arguments.
Existing measures share two main properties: *maximality* capturing the idea that two identical objects are fully similar, and *symmetry* meaning that similarity is symmetric in nature.[16]
Let us now introduce the class of argumentation graphs we are interested in. We focus here on argumentation frameworks where each argument has a basic weight representing different issues.

[16]A formal definition of similarity measure is given in (Amgoud et al., 2018).

We assume that similarities between individuals and subsets of arguments are given by a similarity measure.

Definition 2.40 (Weighted argumentation graph with similarity)
A **weighted argumentation framework with similarity** is a tuple $\mathsf{G} = \langle \mathcal{A}, w, s, \mathcal{R} \rangle$ where

- $\mathcal{A} \subseteq_f \mathcal{A}$,
- w is a function from $\mathcal{A}$ to $[0,1]$,
- s is a similarity measure on $\mathcal{A}$,
- $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$.

Intuitively, $w(a)$ represents the basic strength of argument a, and $s(a,X)$ is the degree of similarity between a and the set X of arguments.
Some semantics for weighted argumentation systems have been proposed in (Amgoud and Ben-Naim, 2016; Amgoud et al., 2017a).

Definition 2.41 (Semantics for weighted argumentation frameworks)
A **semantics** is a function **S** assigning to any $\mathsf{G} = \langle \mathcal{A}, w, s, \mathcal{R} \rangle$ a function $\mathtt{Deg}^{\mathbf{S}}_{\mathsf{G}} : \mathcal{A} \to [0,1]$. $\mathtt{Deg}^{\mathbf{S}}_{\mathsf{G}}(a)$ represents the *overall strength* of a.

The overall strength of an argument should depend on its basic weight, the similarities between its attackers, and the overall strengths of those attackers. It is also worth mentioning that fully similar arguments do not necessarily get the same overall strength. This is mainly because these arguments may have different basic strengths. For instance, if the basic strength of an argument represents the importance degree of its source, then it may be the case that two equivalent arguments are given by two sources, one of which is reliable and the other not. Similarly, in the context of online debate where weights may represent the votes that users have expressed, it may be the case that one argument has attracted many more votes than another (similar) one. This is illustrated in Figure 2.12 on the preceding page where a received two votes, while the similar argument c did not receive any.

2.7.2 Rationality Principles

The principles introduced in Section 2.2.1 on page 13 are based on the strong *implicit* assumption that arguments in a graph are not fully similar. While they can assume that fully redundant arguments are removed from argumentation graphs, they do not account for partially similar ones. Moreover, they are conceived for *flat* argumentation system, that is argumentation frameworks without weights.
Thus, we take inspiration from the list of principles of Amgoud et al. (2017a), and redefine two of them by considering similarities between attackers (reinforcement and counting). The definitions of all the others do not change since similarity is not involved in them.
Recall that *Independence* states that the evaluation of an argument should be independent of any argument that is not related to it by a path.

Property 23 — Directionality. The fact that an argument attacks other arguments should not affect its own evaluation.

Property 24 — Maximality. Non-attacked arguments receive their basic strengths.

Property 25 — Monotony. The overall strength of an argument cannot be improved by an attack.

Property 26 — Proportionality. The greater the basic strength of an argument, the more resilient the argument to attacks.

When the principles of monotony and proportionality are satisfied, it follows that two similar arguments are equally strong if and only if they have the same basic weights. This holds when the

attack relation satisfies the natural property stating that similar arguments receive the same attacks. Let us now switch to the two principles that are affected by similarity. *Reinforcement* states that increasing the quality of an attacker leads to a decrease in the overall strength of its target.

Property 27 — S-Reinforcement. Increasing the quality of an attacker leads to a decrease in the overall strength of its target when the improved attacker is not impeded by similarities with the other attackers of its target.

Counting states that each attacker with a strictly positive overall strength should have an impact on its target. This principle gives birth to two separate principles:

Property 28 — S-Counting. Each attacker with a strictly positive overall strength should have an impact on its target if it brings some novelty to the attackers which are at least as strong as it is.

Recall that similarity can lead to ignoring attackers or considering parts of their strengths. The question of which argument to be sacrificed raises then naturally. Thus, the weakest arguments are disadvantaged. If for instance, two attackers a and b are fully similar but a is stronger than b, the semantics ignores b and fully considers a.

Property 29 — Redundancy Freeness. If an attacker of an argument does not bring any novelty with respect to the other (stronger) attackers of the argument, then it should be discarded.

The last principle, called *Sensitivity to Similarity*, is new and has no counterparts in Amgoud et al. (2017a).

Property 30 — Sensitivity to Similarity. The more similar the attackers of an argument, the stronger the argument.

These four principles are compatible, that is, there exists at least one semantics satisfying all of them. They are also compatible with all the cited ones.

2.7.3 Extending Weighted h-Categorizer Semantics

In (Amgoud et al., 2017a), the semantics that deals with weighted argumentation graphs were analyzed. The results show that there are essentially two categories of semantics: the first one considers only one attacker (the strongest one) among all the attackers of an argument. Examples of such semantics are extension-based ones (Dung, 1995), Trust-based semantics (da Costa Pereira et al., 2011), Iterative schema (Gabbay and Rodrigues, 2015) and Top-based semantics (`Mbs`) (Amgoud et al., 2017a). The second category of semantics takes into account all the attackers of an argument. There are also several such semantics in the literature: weighted h-Categorizer (`Hbs`) and cardinality-based (`Cbs`) proposed both in (Amgoud et al., 2017a), (DF-)QuAD proposed in (Baroni et al., 2015; Rago et al., 2016), as well as the propagation-based semantics.
Similarity comes into play when all attackers may be taken into account. Thus, only the second category of semantics is concerned with the problem of not dealing with similarities. In what follows, we extend `Hbs` for accounting for similarities between attackers. Please note that `Cbs` can be extended exactly in the same way.
The weighted h-categorizer semantics (`Hbs`) takes as input a weighted argumentation system and assigns for each argument an overall strength, which depends on the basis weight of the argument and the sum of the overall strengths of its attackers.

Definition 2.42 (Weighted h-categorizer semantics (`Hbs`))
Let $\mathsf{WF} = \langle \mathcal{A}, w, \mathcal{R} \rangle$ be a weighted argumentation system, and $a \in \mathcal{A}$.

$$\mathtt{Deg}^{\mathtt{Hbs}}_{\mathsf{WF}}(a) = \frac{w(a)}{1 + \sum\limits_{b \in \mathcal{R}^-(a)} \mathtt{Deg}^{\mathtt{Hbs}}_{\mathsf{WF}}(b)}$$

A generalization of some of the semantics (including the weighted h-categorizer semantics), called the "local approach", has been proposed by Cayrol and Lagasquie-Schiex (2005). Indeed, the value of an argument is obtained by combining two functions: one to aggregate the score of its direct attackers (Hbs uses the sum for example) and one attenuation function to apply the "negative" effect of its direct attackers (Hbs uses the reciprocal function for example). Our goal is to modify the function allowing to aggregate the score of the direct attackers in order to take into account the similarities which could exist between them.[17]

The readjustment method (Readjustment weighted h-categorizer (RHbs)). The first possibility consists in aggregating the overall score of the direct attackers, to find the "real" score of each direct attacker of an argument by taking into account the pairwise similarities that could exist with the other direct attackers.

The grouping by thresholds method (Grouping weighted h-categorizer (GHbs)). The second method allows to directly compute the score resulting from the aggregation of all the direct attackers of an argument. The rationale of the method is to do so by considering iteratively the different relevant thresholds of similarity, and by examining which sets of arguments can be assessed as sufficiently similar, in the sense that any pairwise similarity among arguments of the set is above the required threshold. Thus, it constitutes a sort of intermediate step between approaches based on pairwise similarities and approaches based on setwise similarities.

Method based on setwise similarity (Extended weighted h-categorizer (EHbs)). The basic idea of this last method is the following: instead of considering the whole overall strength of each attacker, we only consider a part that is proportional to the novelty of the argument with respect to stronger attackers. For instance, when two fully similar arguments have different overall strengths, we keep the whole value of the strongest argument and discard the value of the weakest one. The attackers should thus be rank-ordered from the strongest to the weakest ones.

Let us now check which principles, among the four principles introduced in this paper, are satisfied by these semantics:

Theorem 2.1

- The semantics RHbs and GHbs satisfy the properties S-Reinforcement and Sensitivity to Similarity. The properties S-Counting and Redundancy Freeness are not satisfied.
- The semantics EHbs satisfies the properties S-Reinforcement, Sensitivity to Similarity, S-Counting and Redundancy Freeness.

The reasons behind the fact that S-Counting and Redundancy Freeness are not satisfied by RHbs and GHbs are directly linked with the discussion on the Monotony principle. Indeed, RHbs and GHbs do not satisfy Monotony, consequently, adding a new attacker to an argument does not always harm this argument.
The semantics GHbs, RHbs and EHbs extend Hbs by similarities. When the arguments are all distinct (i.e., similarities are 0), the four semantics assign the same values to all arguments.

Theorem 2.2 For any $G = \langle \mathcal{A}, w, s, \mathcal{R} \rangle$, if for any $a \in \mathcal{A}$, for any $X \subseteq \mathcal{A} \setminus \{a\}$, $s(a,X) = 0$, then

$$\mathtt{Deg}_G^{\mathtt{GHbs}} \equiv \mathtt{Deg}_G^{\mathtt{RHbs}} \equiv \mathtt{Deg}_G^{\mathtt{EHbs}} \equiv \mathtt{Deg}_{G'}^{\mathtt{Hbs}}$$

[17] See (Amgoud et al., 2018) for details.

where $\mathsf{G}' = \langle \mathcal{A}, w, \mathcal{R} \rangle$.

The four semantics coincide also on the class of argumentation frameworks where each argument is attacked by at *most* one attacker. Indeed, our semantics only consider the similarity between the direct attackers of an argument. So if this argument has no direct attacker then its score is equal to its basic weight. And if this argument has only one direct attacker then we only consider the overall strength of this attacker without any modification.

Theorem 2.3 For any $\mathsf{G} = \langle \mathcal{A}, w, s, \mathcal{R} \rangle$, if for any $a \in \mathcal{A}$, $|\mathcal{R}_{\mathsf{G}}^{-}(a)| \leq 1$, then

$$\mathtt{Deg}_{\mathsf{G}}^{\mathtt{GHbs}} \equiv \mathtt{Deg}_{\mathsf{G}}^{\mathtt{RHbs}} \equiv \mathtt{Deg}_{\mathsf{G}}^{\mathtt{EHbs}} \equiv \mathtt{Deg}_{\mathsf{G}'}^{\mathtt{Hbs}}$$

where $\mathsf{G}' = \langle \mathcal{A}, w, \mathcal{R} \rangle$.

3. Interaction and Argumentation

Argumentation can be used by a single agent to reason about his beliefs, goals and possible actions. But, an even more complex and interesting form of argumentation is when arguments are coming from several different agents. In this context, we face a multi-agent setting where agents hold different (and possibly conflicting) views of the world. We assume that each agent is modeled as an argumentation system and that they want to achieve a common decision, either by aggregating their points of view or by debating, negotiating, and trying to convince others.

In this chapter, we are first interested in studying protocols for persuasion in such multi-agent systems, first in a context of 'classical semantics' a la Dung (Dung, 1995), where we will see how we can characterize the minimal changes necessary to achieve an argumentative goal in the debate. Then, we will turn our attention to the dynamical aspect of gradual semantics, and study the impact of such semantics on the debates and the agents' opinions. Then, we will focus on argumentative negotiation, and look at a protocol allowing agents to take into account a partial knowledge of their opponents. Finally, we will study how the diversity of views observed in such multi-agent systems is consistent with the assumption that every individual argumentation framework is induced by a combination of, first, some basic factual attack relation between the arguments and, second, the personal preferences of the agent concerned regarding the moral or social values the arguments under scrutiny relate to.

Note that part of this work has been done by Dionysios Kontarinis (Kontarinis, 2014) during his PhD thesis, co-supervised with Nicolas Maudet and Pavlos Moraitis.

3.1 Protocols for persuasion

Protocols for persuasion (Prakken, 2006) regulate the exchange of arguments to arbitrate among conflicting viewpoints. When the ambition is to regulate some interaction between different agents, it is often desirable that the outcome of the dialogue is not entirely predetermined from the initial situation (Loui, 2002; Parsons et al., 2003). This means that agents have a chance to influence the outcome of the game depending on how they play. Different properties of such protocols have been studied with the help of game-theoretical concepts (see (Rahwan and Larson, 2009) for a survey).

We are interested in the case of *multiparty argumentation*. In this context, a number ($n > 2$) of agents exchange arguments on a common gameboard. Such dialogues give rise to new types of questions: What types of protocols can coordinate multilateral dialogues? How can turn-taking be defined in multilateral debates, and how does it affect the debate? How can a group's power be measured in a debate? How can a group of agents coordinate to maximize their chances of winning? What should be the collective outcome?

We started this line of work by introducing a simple protocol in (Bonzon and Maudet, 2011).

3.1.1 A Relevance-based Protocol for Multiparty Persuasion

We consider a set N of n agents. We assume that each agent holds an argumentation system $\mathsf{AF}_i = \langle \mathcal{A}, \mathcal{R}_i \rangle$, sharing the same set of arguments $\mathcal{A}$, but with possible conflicting views on attack relations between arguments, coming for instance from different underlying preferences. Suppose we are in the presence of two mutually conflicting arguments, a and b. The agents may have different opinions regarding the respective credibility of a given source, hence in one argument system the attack would hold from a to b, while the symmetric relation would hold for the other agent. More generally, the situation may occur as the result of agents being equipped with preference-based (or value-based) argument systems (Amgoud and Cayrol, 2002b; Bench-Capon, 2002; Brewka, 2001) sharing the same arguments but diverging when it comes to the preferential value attached to arguments.

We assume that agents are *focused* (Rahwan and Tohmé, 2010), that is, they concentrate their attention on a specific (same for all) argument. This argument is referred to as the *issue* d of the debate (Prakken, 2006). Unsurprisingly, agents want to see the acceptability status (under the grounded semantics) of the issue coincide in the debate and in their individual systems. Thus we can see the debate as opposing two groups of agents: $\mathsf{CON} = \{a_i \in N | d \notin \mathcal{E}_{\mathbf{gr}}(\mathsf{AF}_i)\}$ and $\mathsf{PRO} = \{a_i \in N | d \in \mathcal{E}_{\mathbf{gr}}(\mathsf{AF}_i)\}$. If $X = \mathsf{PRO}$ (resp. CON), we have $\overline{X} = \mathsf{CON}$ (resp. PRO).

Merged Argumentation System

What should be the collective view in that case? To tackle this problem, we rely on the notion of a *merged* argumentation system (Coste-Marquis et al., 2007). In the specific case we discuss here, it turns out that a meaningful way to merge is to take the *majority argumentation system* where attacks supported by a majority of agents are kept (this corresponds to minimizing the sum of the edit distances between the AF_i and the merged system (see Coste-Marquis et al., 2007, Prop. 41)). Assuming, on top of that, that ties are broken in favor of the absence of an attack allows to ensure the existence of a single such merged argumentation system, that we denote MAS_N.

Definition 3.1 (Majority argumentation system)
Let N be a set of agents and $\langle \mathsf{AF}_1 \dots \mathsf{AF}_n \rangle$ be the collection of their argumentation systems. The **majority argumentation system** is $\mathsf{MAS}_N = \langle \mathcal{A}, \mathcal{M} \rangle$ where $\mathcal{M} \subseteq \mathcal{A} \times \mathcal{A}$ and $(x,y) \in \mathcal{M}$ when $|\{i \in N | (x,y) \in \mathcal{R}_i\}| > |\{i \in N | (x,y) \notin \mathcal{R}_i\}|$.

The corresponding *merged outcome* is denoted by $\mathcal{E}_{\mathbf{gr}}(\mathsf{MAS}_N)$.

A relevance-based protocol

Building upon (Prakken, 2006; Rahwan and Larson, 2008), we assume that agents' moves should immediately improve their satisfaction with respect to the current situation of the debate. However, no central computation of the whole system takes place, and no coordination between agents is assumed (even if they share the same view). The motivating applications we have in mind are for example online platforms allowing users to asynchronously modify the content of a collective debate.

As in those platforms, agents will exchange arguments via a common gameboard. The issue will be assumed to be present on this gameboard when the debate begins. The "common" argument system is, therefore, a *weighted argumentation system* (Dunne et al., 2009) where the weight is simply a number equal in the difference between the number of agents who asserted a given attack and the number of agents who opposed it. We denote by $(x,y) \in \mathcal{R}_\alpha$ the fact that the attack has a weight α. Let $\mathcal{A}_{\mathsf{GB}}$ be the set of all the arguments present on the gameboard. The collective outcome is obtained by applying the semantics used on the argumentation system $\langle \mathcal{A}_{\mathsf{GB}}, \mathcal{R}_{\mathsf{GB}} \rangle$ where $\mathcal{R}_{\mathsf{GB}} \subseteq \mathcal{A}_{\mathsf{GB}} \times \mathcal{A}_{\mathsf{GB}}$ and $\mathcal{R}_{\mathsf{GB}} = \{(x,y) \in \mathcal{R}_\alpha | \alpha > 0\}$. In words, we only retain attacks

supported by a (strict) majority of agents who have expressed their view on this relation. Observe that following our tie-breaking policy we require the number of agents supporting the relation to strictly overweight the number of agents who oppose it (*i.e* in case of a tie, the relation does not hold).

As said before, a move is acceptable if it immediately modifies the current status of the issue under discussion. Another important feature is that the protocol allows only one argument to be advanced at the same time. A permitted move is as follows: either a positive assertion of an attack $(x,y) \in \mathcal{R}$, or the contradiction of an (already introduced) attack.
When a (relevant) move is played on the gameboard, the following update operation takes place:

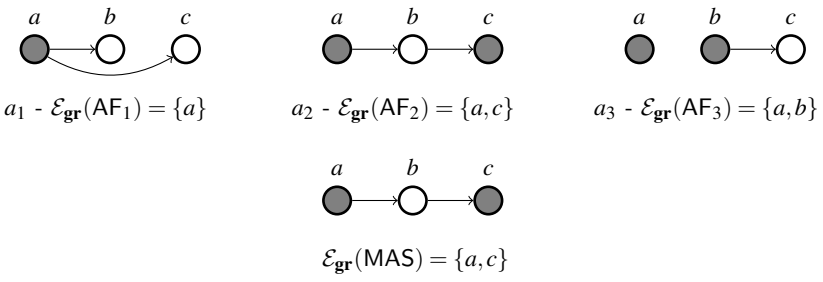
1. after an assertion $(x,y) \in \mathcal{R}$,
   - if $(x,y) \in \mathcal{R}_\alpha$, then $\alpha := \alpha + 1$
   - if $(x,y) \notin \mathcal{R}_\alpha$ and $x,y \in \mathcal{A}_{\mathsf{GB}}$, then the edge is created with $\alpha := 1$
   - otherwise (x is not present), then the node of the new argument is created and the edge is created with $\alpha := 1$
2. after a contradiction of $(x,y) \in \mathcal{R}$ we have $\alpha := \alpha - 1$

The reader may remark that the value of α is binary if agents obey this protocol.
All the ingredients are in place to describe the simple protocol which will regulate the multiparty debate.

1. Agents report their individual views on the issue to the central authority, which then assigns (privately) each agent to PRO or CON.
2. The first round starts with the issue on the gameboard and the turn given to CON.
3. Until a group of agents cannot move, we have:
   a. agents independently propose moves to the central authority;
   b. the central authority picks the first (or at random) relevant move from the group of agents whose turn is active, updates the gameboard, and passes the turn to the other group

When a group of agents cannot move, we say that the gameboard is stable and we refer to $\mathcal{E}_{\mathbf{gr}}(\mathsf{GB})$ as the *outcome of the debate*.

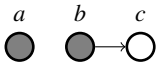
Example 3.1 Let three agents with their argumentation systems and the following merged argumentation framework:

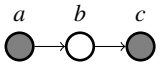


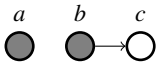
The issue of the dialogue is the argument c. We have $\mathsf{CON} = \{a_1, a_3\}$, $\mathsf{PRO} = \{a_2\}$.
A sequence of moves allowed by the protocol is the following:

a_1 plays for CON, adds $(a,c) \in \mathcal{R}_{\mathsf{GB}}$ 

a_2 plays for PRO, removes $(a,c) \in \mathcal{R}_{\mathsf{GB}}$ 

a_3 plays for CON, adds $(b,c) \in \mathcal{R}_{\mathsf{GB}}$ 

a_2 plays for PRO, adds $(a,b) \in \mathcal{R}_{\mathsf{GB}}$ 

a_3 plays for CON, removes $(a,b) \in \mathcal{R}_{\mathsf{GB}}$ 

a_2 cannot add c in the extension

The game board is stable, we obtain $\mathcal{E}_{\mathbf{gr}}(\mathsf{GB}) = \{a,b\}$. ∎

The first interesting thing to observe in this simple example is the fact that the status of an issue in the merged argumentation system can contradict the opinion of the majority. This is discussed in Coste-Marquis et al. (2007): if agents vote on extensions, the attack relations from which extensions are characterized are not taken into consideration, and a lot of significant information is not exploited.

Another important thing to note in this example is that PRO agents cannot ensure c in $\mathcal{E}_{\mathbf{gr}}(\mathsf{GB})$. It is then impossible to guarantee convergence to the status of the issue obtained in the merged argumentation system. This is because agent a_1 has no interest to play the attack relation (a,b), which appears in the MAS. As studied in a different context by Rahwan and Larson (2008), this can be seen as a strategic manipulation by withholding an argument or an attack between arguments. But is it even possible to reach the merged outcome in this case? We leave it to the reader to check that this is not the case here.

This example shows that the characterization of the result obtained by debates following this protocol is not as simple as one can believe at first glance.

Global arguments-control graph

To characterize the status of the issue obtained by our protocol we will need the notion of *global arguments-control graph* (ACG). The idea here is to gather the attacks of all agents in the same argumentation graph, and then determine which group, PRO or CON, has control over some path of this graph, and thus a possible way to reach its preferred outcome. To do so, we first need to define the notion of *control* over an attack relation:

Definition 3.2 (Control, playability)
Let N be a set of agents, $\langle \mathsf{AF}_1 \dots \mathsf{AF}_n \rangle$ be the collection of their argumentation systems, and $\mathcal{L} = \cup_{i \in 1\dots n} \mathcal{R}_i$ be the union of their attack relations. Let $X \in \{\mathsf{CON}, \mathsf{PRO}\}$. Finally, let $add_{(a,b)} = \{a_i \in N | (a,b) \subseteq \mathcal{R}_i\}$ be the set of agents being able to add the attack (a,b), and $rem_{(a,b)} = \{a_i \in N | (a,b) \not\subseteq \mathcal{R}_i\}$. be the set of agents being able to remove the attack (a,b).

- X has the **constructive control** of $(a,b) \in \mathcal{L}$, denoted by $X^+(a,b)$, iff $|add_{(a,b)} \cap X| > |rem_{(a,b)} \cap \overline{X}|$, that is if the number of agents in X who can add (a,b) is greater than the number of agents in $\overline{X}$ who can remove it.
- X has the **destructive control** of $(a,b) \in \mathcal{L}$, denoted by $X^-(a,b)$, iff $|rem_{(a,b)} \cap X| \geq |add_{(a,b)} \cap \overline{X}|$, that is if the number of agents in X who can remove (a,b) is greater or equal than the number of agents in $\overline{X}$ who can add it.

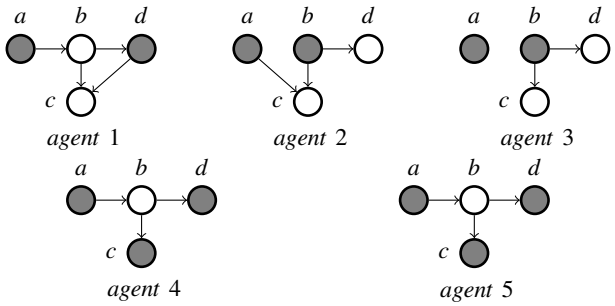
We will say that $(a,b) \in \cup_{1\dots n} \mathcal{R}_i$ is **playable** by a X, denoted by $X_\bullet(a,b)$, if and only if there is an $a_i \in X$ such that $(a,b) \in \mathcal{R}_i$.

For the sake of readability, we will only specify the information about playability when it is relevant.

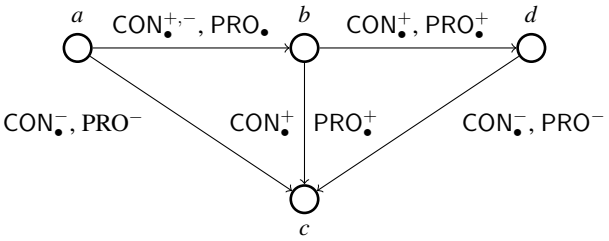
Definition 3.3 (Global arguments-control graph)
Let N be a set of agents and $\langle \mathsf{AF}_1 \dots \mathsf{AF}_n \rangle$ be the collection of their argumentation systems. The **global arguments-control graph** is $\mathsf{ACG}_N = \langle \mathcal{A}, L \rangle$ is constructed as follow:

1. $\mathcal{L} = \cup_{i \in 1 \dots n} \mathcal{R}_i$
2. Label each $(a,b) \in \mathcal{L}$ by the information about control and playability for each group $X \in \{\mathsf{PRO}, \mathsf{CON}\}$.

Example 3.2 Five agents have the following argumentation systems:



The issue of the dialogue is the argument c. We have $\mathsf{CON} = \{a_1, a_2, a_3\}$, $\mathsf{PRO} = \{a_4, a_5\}$. The global arguments-control graph is the following:



We easily see that the issue c is a **possible outcome** for the agents in CON: CON can attack c with b. Then, the only possible move for PRO is to attack b with a. However, CON can remove this attack, and PRO has no other move.
But c is also a **possible outcome** for PRO: CON can start by the attack (d,c), which is playable by a_1. Then, a_5 will defend with (b,d), and a_1 counter-attack with (a,d). If the next move of PRO is to remove (d,c), then CON has no other move left: it cannot add the attack (b,c), as it is defended by a; and it cannot remove the edge (a,b) as it does not drop c from the extension.
■

This example shows that our protocol leaves some degree of freedom to agents (strategy may make a difference) and that the issue of the debate is not predetermined from the initial situation.
The possibility for both teams to win the debate comes from the fact that the attack (a,b) is a *switch*.

Definition 3.4 (Switch)
An edge (x,y) on a path P is a **switch** for the issue d if

1. it is a defense for d on P
2. it is playable by CON
3. there exists an even-length path from y to d such that all the attack edges are playable by

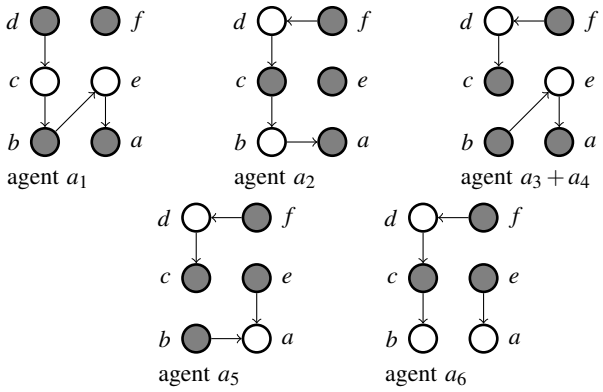
CON and all the defense edges are playable by PRO. So it is also a potential attack for d via a different path.

Essentially, what this definition says is that there is a possibility that this edge (x,y) may be played as an attack by CON.

Is it useful to allow reinforcement?

A natural extension is to consider that a move may also be relevant as long as it *reinforces* (or symmetrically *weakens*) an edge which, if deleted and all other things being equal, would change the status of the issue. Essentially, besides the relevant moves as defined in the previous section, this would allow agents to augment the weight of an existing attack, and we refer to this as a *reinforcement move*. Symmetrically, agents may weaken an attack even if hey don't directly delete it, and we refer to this as a *weakening move*. This extended protocol would allow any number of such relevant moves during a group's turn, but (as before) would only switch to the other side after a change in the current status of the issue. However, the following example tells us that it is not beneficial for an agent to play reinforcement moves. Worse, and rather counter-intuitively, it can be damaging for agents to do so.

Example 3.3 Let 6 agents with the following argumentation systems.



The issue of the debate is a. There are four agents PRO and two agents CON. The key to the analysis is to see that PRO agents initially hold constructive and destructive control on (c,b). Now compare the following sequences of moves. In the first one, a_5 plays (b,a). Then a_1 plays (c,b) and a_2 reinforces this move. At this point, PRO loses its destructive control on (c,b). Assume a CON agent plays (d,c). a_1 can remove (b,a). Then a_5 can play (e,a), a_3 can defend with (b,e). Now, with a CON agent playing (f,d), the debate is doomed with the issue *out*.
In the alternative case where no reinforcement is played, we are in the case discussed in the first protocol: PRO can remove the attack (c,b), and win the debate. ∎

This result tells us that in the absence of coordination, agents are better off employing directly relevant moves, hence adopting a "wait and see" approach. Still, using reinforcement moves may prove useful in practice, in contexts where the debate is limited: for instance, agents may be impressed by seemingly large majorities and avoid these issues to concentrate on some other ones.

This protocol, despite its simplicity, raises several interesting questions: What strategies can be followed by the agents? If several outcomes are possible depending on how agents play, how can we decide whether the outcome of the dialogue can be accepted by all the participants in the debate? We will first look at the former question, wondering how agents can select the "best" move to play.

3.1.2 Target sets

We are interested here in the study of the dynamics of argumentative debates: it is not clear how agent strategies could change the outcome of debates. In the previous section, we introduced a very simple dynamic, based on a direct notion of relevance inspired by (Prakken, 2005). We have shown in particular that in the absence of coordination and with myopic behavior, agents can play against their own interests. This justifies the fact that some "guidance" might be useful to agents, without assuming any sort of explicit coordination among agents. We thus turn our interest to the notion of *target sets*, that has been proposed in (Boella et al., 2011).
Roughly speaking, a target set specifies the minimal change necessary to achieve an argumentative goal in the debate. The intuition is that agents should be better off following their target set recommendations.

Basic definitions of target sets

A target set is a minimal set of *actions* on an argumentation system allowing to achieve a given argumentative goal.

Definition 3.5 (Action on a gameboard)
Let $\mathsf{GB} = \langle \mathcal{A}_{\mathsf{GB}}, \mathcal{R}_{\mathsf{GB}} \rangle$ be the gameboard at a given time. An **action on** GB is a tuple $((x,y),s)$ with $(x,y) \in \mathcal{A}_{\mathsf{GB}} \times \mathcal{A}_{\mathsf{GB}}$ and $s \in \{+,-\}$, which stands respectively for adding or removing an attack on the GB.
The **resulting** GB after playing a set of actions m, is denoted as $\Delta(\mathsf{GB}, m) = \langle \mathcal{A}_{\mathsf{GB}}, \mathcal{R}^m_{\mathsf{GB}} \rangle$.

In what follows, we will say that an argument $a \in \mathcal{A}$ is **credulously accepted** (resp. **skeptically accepted**) w.r.t. system AF under semantics σ, denoted $\sigma_\exists(a, \mathsf{AF})$ (resp. $\sigma_\forall(a, \mathsf{AF})$), if and only if a belongs to at least one (resp. to every) extension of AF under the σ semantics.[1] We will denote by $\sigma \in \{ad, co, pr\}$ the set of **ad**(missible), **co**(mplete) and **pr**(eferred) semantics. Moreover, as stated in (Dung, 1995), an argument $a \in \mathcal{A}$ belongs to the grounded extension if and only if it is skeptically accepted under the complete semantics. We will then denote in the following the grounded extension by $co_\forall(a, \mathsf{AF})$.
Let us now describe the types of goals that we focus on. We focus on the acceptance of a single argument $d \in \mathcal{A}$ called the *issue*, and we consider these two types of goals:

Definition 3.6 (Goals)
Let $X \in \{\exists, \forall\}$. A **positive goal** $g = \sigma_X(d)$ is **satisfied** in GB if and only if $\sigma_X(d, \mathsf{GB})$ holds. A **negative goal** $g = \neg\sigma_X(d)$ is **satisfied** in GB if and only if $\sigma_X(d, \mathsf{GB})$ does not hold.
A **goal** will denote either a positive or a negative goal.

If a specific (positive or negative) goal is not satisfied in GB, then we search for possible actions m on GB leading to a modified system $\Delta(\mathsf{GB}, m)$ in which that goal is satisfied.

Definition 3.7 (Successful set of actions, Target set)
m is a **successful set of actions for goal** g if and only if g is satisfied in $\Delta(\mathsf{GB}, m)$. We denote by $\mathbb{M}(\mathsf{GB})$ the set of all successful sets of actions.
m is a **target set for goal** g if and only if m is a $\subseteq$-minimal set of actions from $\mathbb{M}(\mathsf{GB})$. We denote by $\mathbb{T}(\mathsf{GB})$ the set of all target sets.

We now provide some properties of successful actions and target sets.

[1] Thus, $\sigma_\exists(a, \mathsf{AF})$ if and only if $a \in ca_\sigma(\mathsf{AF})$, and $\sigma_\forall(a, \mathsf{AF})$ if and only if $a \in sa_\sigma(\mathsf{AF})$ (see Section 2.1 on page 9).

Proposition 3.1 It holds that

$$\mathbb{M}^{co}_{\forall} \subseteq \mathbb{M}^{pr}_{\forall} \subseteq \mathbb{M}^{\sigma}_{\exists} \text{ and } \mathbb{M}^{\sigma}_{\neg\exists} \subseteq \mathbb{M}^{pr}_{\neg\forall} \subseteq \mathbb{M}^{co}_{\neg\forall}$$

Proposition 3.2
If m is a set of actions such that $m \in \mathbb{T}^{co}_{\forall}$ and $m \in \mathbb{T}^{\sigma}_{\exists}$, then $m \in \mathbb{T}^{pr}_{\forall}$ (1)
Moreover, if m is a set of actions such that $m \in \mathbb{T}^{\sigma}_{\neg\exists}$ and $m \in \mathbb{T}^{co}_{\neg\forall}$, then $m \in \mathbb{T}^{pr}_{\neg\forall}$ (2)

Figure 3.1 graphically represents the links between the set of successful sets of actions and the target sets for the positive and the negative goals.

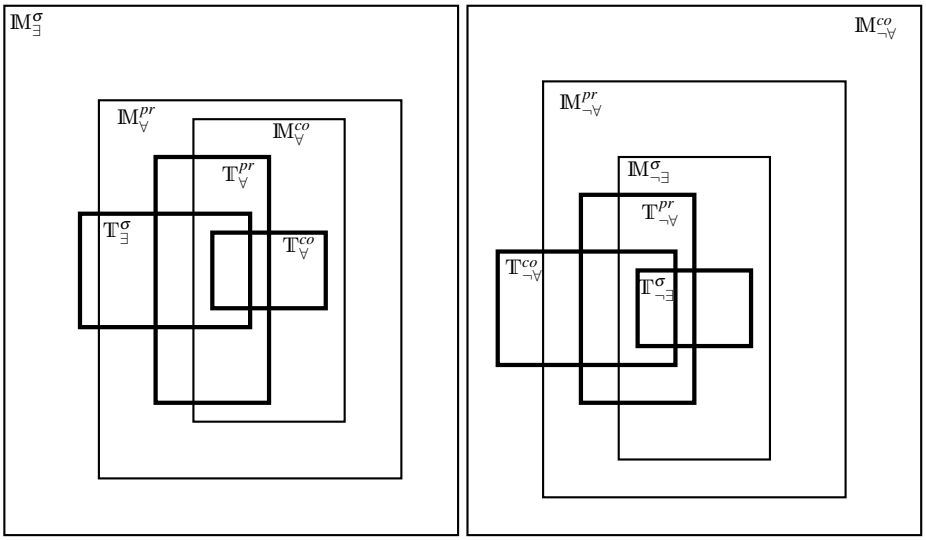


Figure 3.1: On the left: The sets of successful sets of actions and target sets for the positive goals. On the right: The sets of successful sets of actions and target sets for the negative goals.

Having highlighted some properties of the successful set of actions and of the target sets, we can now define a rewriting procedure that computes target sets.

Computing Target Sets

To compute all the target sets for some types of goals, we have provided a set of rewriting rules using the Maude[2] system (Clavel et al., 1999), which is based on rewriting logic.
Maude is both a declarative programming language and a system. A program in Maude is a logical theory, and a computation made by that program is a logical deduction using the axioms of the theory.
Our Maude program[3] is given as input a term which describes an argumentation system and contains either $\mathsf{PRO}(d)$ or $\mathsf{CON}(d)$, with $d \in \mathcal{A}$. If we want to ensure the (positive) goal of accepting argument d under some semantics, we start with $\mathsf{PRO}(d)$. Otherwise, if we want to ensure the (negative) goal of rejecting d, we start with $\mathsf{CON}(d)$. Maude starts from these atoms and, based on a set of rewriting rules and equations, rewrites the initial term, thus producing new terms, which are, in turn, rewritten. The system stops when all the computed terms are non-rewritable. Every term of the output corresponds to an action on the initial system. Their connection with the status of d is detailed in the following proposition.

[2] http://maude.cs.uiuc.edu
[3] An interested reader can find this Maude program in (Kontarinis et al., 2013).

Proposition 3.3 The following table illustrates for which goals the rewriting procedure is correct for successful actions and/or complete for target sets.

Goal	*Correctness for successful actions*	*Completeness for target sets*
$\sigma_{\exists}(d)$	Yes	Yes
$pr_{\forall}(d)$	No	No
$co_{\forall}(d)$	No	Yes
$\neg\sigma_{\exists}(d)$	No	No
$\neg pr_{\forall}(d)$	No	?
$\neg co_{\forall}(d)$	Yes	Yes

Note that the completeness regarding the goal $\neg pr_{\forall}(d)$ is left open so far. However, for the sake of readability, we draw Figure 3.1 assuming that the answer is "Yes".

Strategies and Target Sets

Then, we experimentally investigated how well strategies based on target sets behave. We studied several dynamics, of increasing complexity, where the notion of target set is thoroughly exploited. Our experimental results show in particular that the use of these sophisticated strategies provides an advantage to the side using it and that it shortens the length of debates.
To study different strategies, we defined a specific protocol. As in the protocol defined in Section 3.1.1 on page 62, the agents focus on the status (under the grounded semantics) of a single argument $d \in \mathcal{A}$, which is the issue of the debate. As the goal of an agent $i \in N$ is to have the issue's status be the same, on the GB and in her private system, at the end of the debate, we can once again distinguish two groups of agents: the agents of the group PRO (resp. CON) who have (resp. do not have) the issue in the grounded extension of their systems.
However, unlike the previously mentioned protocol, agents can *vote* for or against an attack (and not just add or remove it). Then, the *verdict* of an attack is `True` if the attack is present on the gameboard (the number of votes of the attack is strictly greater than the number of votes against it), `False` otherwise. To ensure the termination of our protocol, we assume that an agent cannot vote on the same attack twice.
The protocol is defined by the following:

- **Participants**: A finite set of agents N, each one being either PRO or CON, according to her opinion on the issue's status.
- **Turntaking**: Round-robin. The token is given to each agent, in turn, and comes back to the first agent once all agents have played.
- **Permitted moves**: Agent i at timestep t can either:
  - Vote for an attack, denoted $\langle (a,b), +, i \rangle$
  - Vote against an attack, denoted $\langle (a,b), -, i \rangle$
  - Play a **pass move** (giving the token to the next agent).
- **Stopping condition**: $|N|$ pass moves have been played in a row.
- **Winning condition**: Once the debate has stopped, all PRO (resp. CON) agents win if and only if the issue belongs (resp. does not belong) to the grounded extension of the argumentation system of GB.

When having the token, an agent can vote on any of the attacks under discussion, but which one should he choose? In general, a *strategy* states, for each agent, what move should be uttered next in the course of the debate. When a strategy returns a single move, we say it is *deterministic*.

Depending on the information required to take this decision, we can distinguish different kinds of strategies:

- *(k)-history-based strategies*: the strategy selects moves based on the last k moves uttered in the debate, noted *h(k)-strategies*. For instance:

  "If someone just attacked argument a, I will try to defend it."

- *(k)-state-based strategies*: the strategy selects moves based on the last k states of the gameboard, noted *s(k)-strategies*. For instance:

  "If $a \in \mathcal{E}_{gr}(\mathsf{GB})$, then I will utter the attack (d,a)."

We say that a strategy s has a richer information basis than a strategy s' (noted $s \rhd s'$) when it uses more information to select the next moves.
In what follows, we study a natural class of *s(1)-strategies*, as we define strategies based on the computation of the target sets of the last GB. We also assume that all agents from one side (PRO or CON) use the same strategy. This facilitates the analysis but constitutes of course a simplification. Moreover, we assume that the agents cannot disclose their private argumentation systems. Thus, agents do not have any knowledge of the other agents' private systems.
As said before, the analysis of target sets and their properties leads us naturally to think that agents would profit from focusing on attacks on target sets, as it is the fastest and most economical way to achieve a goal.

Dominance. One may wonder whether "playing within target sets" is a dominant strategy, that is, whether agents can never be better off playing a different strategy, whatever the strategy of the other party is. Note first that "playing within target sets" does not constitute a single strategy, but instead a class of strategies, in fact, a subclass of *s(1)-strategies*. So when say "a dominant strategy", we abuse language and mean *any* strategy belonging to this class. This turns out to be a too-demanding notion because the strategy of the other player can be of any kind, in particular, it may be such that moves played outside a target set will precisely be the moves required to lead to a winning result.

Symmetric Equilibrum. We may then ask whether a weaker property can be guaranteed: is it the case that, *if the other agent follows a strategy consisting of playing within target sets*, then agents of the other side will not have the incentive to play differently, ie. whether this constitutes a symmetric equilibrium. This is not the case either.[4]

All in all, playing in target sets looks intuitively like a good strategy, but it seems difficult to obtain theoretical guarantees. This leads us to study it experimentally.
To do so, we define 5 strategies, from the simpler to the more complex, mainly focusing on target sets. Strategy 0 is the exception, as it is a *random strategy*, which will allow us to assert that playing in the target sets is useful. Note that when there are no available moves for an agent (we remind that an agent cannot vote on the same attack twice), that agent obligatorily passes.

Strategy 0: This is a random strategy, where (1) if the agent is winning, then she plays pass. (2) otherwise, she votes randomly on an attack on the gameboard.

Strategy 1: The idea of this strategy is to allow only agents who are not satisfied with the current state of the gameboard to vote. Moreover, these agents can only vote if they can change the status of the issue (and thus, if they can change the verdict of an attack belonging to a target set of cardinality 1).[5] More precisely: (1) if the agent is winning, then she plays pass. (2)

[4] Counter-examples for these two properties are given in (Kontarinis et al., 2014a).
[5] Note that this strategy is the one studied in Section 3.1.1 on page 62.

otherwise, the agent can only vote on an attack if this vote allows changing the status of the issue.

Strategy 2: This strategy improves the previous one by allowing agents to vote on a target set of cardinality greater than 1: an agent can vote on an attack if she can change its verdict, but this vote does not have to change the status of the issue. More precisely: (1) if the agent is winning, then she plays pass. (2) otherwise, the agent can only vote on an attack if this attack belongs to a target set, and if this vote allows to change the verdict on this attack.

Strategy 3: This strategy allows an agent to vote on an attack belonging to a target set, even if she cannot change the verdict on this attack. More precisely: (1) if the agent is winning, then she plays pass. (2) otherwise, the agent can vote on any attack belonging to a target set (towards changing the verdict).

Strategy 4: This strategy improves the previous one by allowing a winning agent to play a move that renders the goal of the other team more difficult to be reached. More precisely: (1) if the agent is winning, then she can vote on an attack that belongs to a target set for the goal of the other team and "reinforce" it.[6] (2) otherwise, the agent can vote on any attack belonging to a target set (towards changing the verdict).

An agent can compute a set of possible votes, using any of the above strategies. Then, he can either randomly choose a vote among them, or use a more subtle heuristic. We have defined three heuristics that can be used for filtering the initial set of possible votes.

Heuristics A the agent randomly chooses a possible vote.

Heuristics B the agent filters out all possible votes on non-minimal (wrt. cardinality) target sets [7]. Then, she randomly chooses a vote.

Heuristics C the agent filters out all possible votes on non-minimal (wrt. cardinality) target sets. If she can change the verdict of an attack among the remaining ones, she filters out all the attacks she cannot change. Then, she randomly chooses a vote.

Coupling a strategy with a heuristics gives us a specific **strategy profile**. As Strategy 0 does not use target sets, it can not be coupled with any heuristics. Also, in Strategy 1 an agent can only vote on an attack if it belongs to a target set of cardinality 1 and she can change its verdict, so it does not make any sense to associate Strategy 1 with heuristics B or C. In the same way, in Strategy 2 an agent can only vote on an attack if she can change its verdict, so it does not make sense to couple Strategy 2 with heuristics C. We thus have the following strategy profiles to consider (the number indicates the strategy type and the capital letter the heuristics): $SP = \{0, 1, 2A, 2B, 3A, 3B, 3C, 4A, 4B, 4C\}$.
We assume that the agents of the same group (PRO or CON) are using the same strategy profile during a debate. This is done to draw more easily conclusions on how the strategy profiles fare against each other. We can thus introduce the notion of **debate profile**. A debate profile is defined as a couple $(SP_{\mathsf{PRO}}, SP_{\mathsf{CON}})$ with $SP_{\mathsf{PRO}}, SP_{\mathsf{CON}} \in SP$. It indicates that all agents in the PRO (resp. CON) group are using the strategy profile SP_{PRO} (resp. SP_{CON}). Since there are 10 strategy profiles, there exist $10 \times 10 = 100$ different debate profiles. In the following, we first examine Strategy 0, and then we turn our attention to the 9 other strategy profiles which use target sets (thus on their corresponding $9 \times 9 = 81$ debate profiles).
For the analysis of the results and the evaluation of the strategies and debate profiles, we considered three criteria:

[6] And thus making it more difficult for the other team to change the verdict on this attack.

[7] For example, if an agent can vote on two attacks, the first being in a target set of cardinality 1, and the second in a target set of cardinality 2, then he will filter out the second option.

- **Debate length**: the average number of rounds in the debate.
- **Happiness**: the percentage of coincidence between the debate's result and the majority result. Its interest is better understood from the perspective of the debate's central authority. For example, if the central authority chooses a strategy profile for both PRO and CON (eg. the same one), then it may wish to know which one would help the majority, and which one would offer more chances to the minority.
- **Rationality**: the percentage of coincidence between the result of the debate and the merged result.

We also want to find the "best" strategies, meaning the strategies that maximize a group's chances to win the debate.

We begin our analysis with the random strategy profile. As far as the maximization of a group's winning chances is concerned, the random strategy profile did not fare worse than the quite simple strategy profiles 1 and 2X. The reason is that its drawback (the fact that agents playing random attacks could harm their own group), was balanced by the drawback of profiles 1 and 2X, which can "block" a group normally able to change an attack by casting two or more votes. The winning percentage of a group increases if instead of the random profile an elaborated profile 3X and 4X is used: the winning percentage always increased, up to 25% in some cases (although less in others). A key disadvantage of the random profile is that, if a group uses it, then the number of rounds of the debate explodes. In most cases, when one group adopted the random profile, the number of rounds increased by a factor of 10 (eg. from 25 rounds to 250 rounds). Remember that in the profiles focusing on target sets, if an agent has no move in a target set, she plays pass. This is not the case in the random profile, where a group can play a lot of "dummy" moves before achieving its goal.

On the positive side, if a group uses the random strategy, then the percentage of agreement with the merged outcome is quite high (in almost all the cases we tested that percentage was bigger than 90%). Naturally, the reason behind this, is that the group using the random profile will cast a lot more votes during the debate, and as a result, the GB will resemble more to the merged system. This was even clearer when both groups used the random profile when that percentage went up to 97.6%.

Concluding, the fact that the number of rounds increases dramatically when a group uses the random strategy, as well as the fact that it fares worse (as far as winning the debate is concerned) than strategies 3X and 4X, lead us to not include the random profile in the following tests, where we just compare the 9 strategy profiles focusing on target sets.

We now turn our attention to the 9 strategy profiles focusing on target sets and their corresponding 81 debate profiles.

Each of the four graphics in Figure 3.1.2 on the facing page contains information on *all* debate profiles focusing on target sets. The top left shows the percentage of PRO wins (for every profile), the top right shows the average number of rounds of the debates, the bottom left shows the percentage of agreement between the results of the debates and the merged results, and the bottom right shows the percentage of agreement between the results of the debates and the majority results.

Let us first consider the criterion of *debate length* (top-right). The lowest number of rounds is found when both agents use strategy 4. A small number of rounds is also obtained in profiles (1,4X), (4X,1) (where X $\in \{A,B,C\}$) and (1,1). For the latter, the reason is that there are cases where a group cannot vote on an attack because no single agent can change it (and thus the debate stops). For profiles (4X,4Y) the reason debates are short is that agents are not forced to play (useless) pass-moves, as they can reinforce attacks on the GB while they are winning. This is not possible with profiles (3X,3Y) which gives the longest debates. Note that agents using the strategy profiles 4X have the incentive to give more information than with the other strategy profiles. That can be seen as a disadvantage for agents who wish to hide information. A last remark on debate length: If we concentrate only on rounds that do not contain pass moves (let us call them *no-pass rounds*),

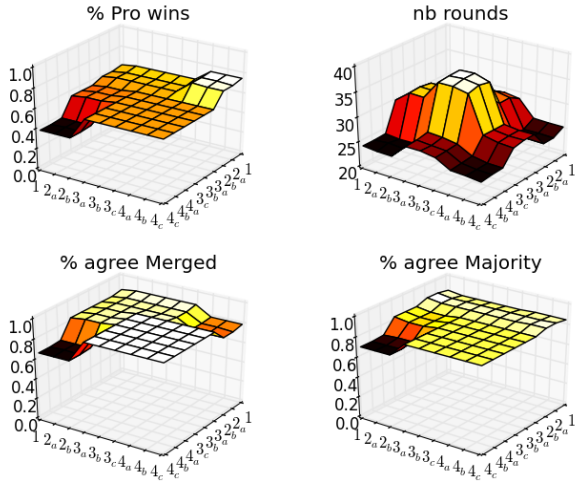


Figure 3.2: Top-Left: Percentage of wins by PRO. Top-Right: Number of rounds of the debates. Bottom-Left: Percentage of agreement with the merged result. Bottom-Right: Percentage of agreement with the majority result. PRO strategies are shown on the left side, and CON strategies on the right side of every graphic.

then the results of strategy profiles 3X and 4X are inverted. Strategy profiles 4X lead to more no-pass rounds, than profiles 3X (eg. (4C,4C) lead on average to 11.97 no-pass rounds, while (3C,3C) lead on average to 10.36 no-pass rounds). We see that when profiles 3X are used, many rounds involve pass moves, and this is the reason why profiles 3X have the biggest total number of rounds.

Let us now focus on *rationality* (bottom-left). The most "rational" outcomes (closer to the results of the merged system) are obtained when both groups use one of the strategies: 3A, 3B, 3C, 4A, 4B, 4C (the percentage of agreement being 0.88). The only cases where the results of the debates are farther from the merged results are when a group uses strategy profile 4X and the other group uses strategy profile 1 or 2X. So, we pull away from the merged result when a group uses the most advanced strategy (4X), while the other a simple one (1 or 2X). The smallest agreement is 0.66, at the profile (1,4X).

Similar results are obtained when we focus on *happiness* (bottom-right). Almost all profiles give a similar value of agreement with the majority (about 0.85). However, when PRO uses strategies 1, or 2X, and CON uses 4X, the debate's result starts to move away from the majority's opinion (its minimum value is 0.7).
Regarding the strategy which is most likely to win a debate, the most elaborated strategies 3 and 4 provide a clear advantage. PRO's best chance to win is when the profile (4X,1) is used (0.75 percentage of PRO winning). Similarly, CON's best chance to win is in profile (1,4X) (0.38 of PRO winning). In general, no matter what strategy a group is using, the other group increases its winning percentage if it uses strategy 3 or 4, instead of the simpler 2 and 1 (1 being the worst choice).

Finally, some remarks on the heuristics. Heuristics C which focuses on the smallest target sets, and prefers moves able to add/remove an attack, was expected to lead to the quickest debates. This was verified, although its results were not significantly better than the results of the simpler heuristics B and A. For example, the debate profile (4C,4C) lead to 23.88 rounds on average, while the profile (4A,4A) lead to 24.81. Also, the debate profile (3C,3C) leads to 35.29 rounds on average, while the

profile (3A,3A) leads to 36.29. We conjecture that, when heuristics C is used instead of B or A, the decrease in the number of rounds is small, because the randomly generated systems do not contain many target sets, and these target sets do not have great differences in size. We expect that in the case of master systems with target sets of considerably different sizes, heuristics C will lead to a more significant decrease in the number of rounds, compared to heuristics B and A.
To conclude, a general observation is that the more sophisticated strategy profiles (3X and 4X) are the best choices for the agents who want to win the debate. Their main difference lies in the average number of rounds, and the amount of information disclosed during the debate. Surprisingly, the simpler strategy profiles (1 and 2X) offer an interesting alternative, provided that the debate's central authority can ensure that both groups will use a simple strategy profile and that no group will switch to using a sophisticated one. It is worth noting that, in the above experiments, the probability that the winner is the same when either profile (1,1) or profile (3C,3C) is used, was almost 95%. Finally, the use of heuristics C shortens the length of the debates, though more tests are needed to evaluate its impact.

Playing outside Target Sets

As we have just seen, agents are better off following their target set recommendations. One challenge though is that target sets may prescribe multiple changes, and in general it is impossible to assume that agents have the opportunity to make all these changes. Thus, in this section, we will analyze the evolution of target sets when it cannot be assumed that an agent can play all the actions prescribed by a target set.[8]
We first turn our attention to what happens if we play actions that are not part of any target set for a given goal. Intuitively, we will not perform in this case an action allowing to get closer to the given goal. We will see that playing such an action could even lead us farther away from the goal, in the sense that we will need to perform more actions to obtain this goal.
Let m be a set of actions on a gameboard GB such that *m does not contain any action of any target set of* GB. After playing m, the goal remains unsatisfied in the resulting gameboard $\mathsf{GB}' = \Delta(\mathsf{GB}, m)$,[9] while the set of target sets changes and is denoted $\mathbb{T}(\mathsf{GB}')$.
The first property states that, for every new target set, there exists an old target set which is a subset of it.

Proposition 3.4 If m is a set of actions on GB such that $\mathsf{GB}' = \Delta(\mathsf{GB}, m)$, and no action of m belongs in a target set of GB, then $\forall t' \in \mathbb{T}(\mathsf{GB}')\ \exists t \in \mathbb{T}(\mathsf{GB})$ such that $t \subseteq t'$.

The second property states that, for every new target set, there is no old target set which is a strict superset of the new one. Thus, no target set "shrinks" when playing m.

Proposition 3.5 If m is a set of actions on GB such that $\mathsf{GB}' = \Delta(\mathsf{GB}, m)$, and no action of m belongs in a target set of GB, then $\forall t' \in \mathbb{T}(\mathsf{GB})\ \nexists t \in \mathbb{T}(\mathsf{GB})$ such that $t' \subset t$.

Moreover, for every old target set, there is at least one new target set which is a superset of the old one. Thus, no target set "disappears" when playing m.

Proposition 3.6 If m is a set of actions on GB such that $\mathsf{GB}' = \Delta(\mathsf{GB}, m)$, and no action of m belongs in a target set of GB, then $\forall t \in \mathbb{T}(\mathsf{GB})\ \exists t' \in \mathbb{T}(\mathsf{GB}')$ such that $t \subseteq t'$.

Finally, the cardinality of the new set of target sets is greater or equal to the cardinality of the previous set of target sets.

[8] This work was originally published in (Kontarinis et al., 2014b)

[9] Because if m was successful, then a subset of m would be a target set. Impossible since m contains no actions of any target set

Proposition 3.7 If m is a set of actions on GB such that $\mathsf{GB}' = \Delta(\mathsf{GB}, m)$, and no action of m belongs in a target set of GB, then $|\mathbb{T}(\mathsf{GB}')| \geq |\mathbb{T}(\mathsf{GB})|$.

Thus, if a set of actions m which does not contain any action of any target set of $\mathbb{T}(GB)$ is played, then according to Property 2, every new target set in $\mathbb{T}(GB')$ will be bigger than some old target set of $\mathbb{T}(GB)$. In that sense, it will become harder (or at least not easier) to satisfy the goal under consideration, even if the cardinality of the set of target sets may grow.

The previous results indicate that playing outside of target sets for a goal is not a good idea. But what happens if we play a set of actions m belonging to a target set? Let $t \in \mathbb{T}(GB)$ and let $m \subset t$. If we play m, then $t \setminus m$ will become a target set of the new gameboard $\Delta(GB, m)$. However, we are uncertain about the other target sets of GB: some may shrink too, but others may remain unchanged, or even grow. Moreover, the cardinality of $\mathbb{T}(GB)$ could decrease, remain unchanged or increase in $\Delta(GB, m)$.
Therefore, playing in a target set shrinks (at least) that target set, regardless of what happens to the other target sets. In that sense, at least one "path" toward the satisfaction of the goal becomes shorter, whereas this is not the case if we play outside target sets.

3.1.3 Expertise in Argumentative Debates

We now turn to another question, regarding how the agents who participate in a debate are likely to *accept* the outcome of this debate.
Once a debate is over, participating agents may not be entirely satisfied with the procedure's final outcome. First, of course, they may not be satisfied with the outcome itself (Caminada and Pigozzi, 2011; Rahwan and Larson, 2008). In this section[10] we concentrate on a different issue though, namely, the fact that the obtained result may in some way be *controversial*. In particular, in our context, this may result from two (distinct, but related) situations:

- *argumentative controversy*: when argumentation theory does not provide a clean-cut decision: this is the case in particular when several (conflicting) acceptable outcomes are returned;
- *voting controversy*: when voting does not offer a clear majority to support the fact that an attack should be taken into account (or not).

Our objective in this work is to set up a framework where these issues can be formally studied. Once a debate is obtained, we discuss how the choice of an additional expert should be made to make the result less controversial. We emphasize that our work is *not* dependent on a specific protocol. For what matters, the resulting debate may be the outcome of a multilateral protocol like the one proposed in Section 3.1.1 on page 62, or of a merging process (Cayrol and Lagasquie-Schiex, 2011; Coste-Marquis et al., 2007). Instead, we study how the different types of expertise of agents should be modeled, and how the additional expert may affect the current debate.
In order to define the expertise of agents over arguments or attacks, we need to attach to arguments some keywords, specifying which topics this argument is about. It is common practice in such systems (Toni and Torroni, 2011). In this work, we assume that the set of potential *topics*, denoted T, is known and fixed a priori by the system. Attached to each argument is a set of topics that, in principle, can be empty or contain as many topics as wished. From topics attached to arguments, we deduce how topics are attached to potential attacks.

Definition 3.8 (Topics)
Let T be the set of topics. The set of **topics of an argument** $a \in \mathcal{A}$ is given by function $top(a) \subseteq T$.
The set of **topics of a (potential) attack** $(a,b) \in \mathcal{R}$ is given by the function[a] $top(a,b) =$

[10] This work was originally published in (Kontarinis et al., 2012)

$top(a) \uplus top(b) \subseteq T \uplus T$.

[a] $\uplus$ indicates the multiset union

For an attack, it is thus possible to distinguish three levels of "relevance" for topics: *prominent* topics (attached to both arguments) denoted $prom(a,b) \subseteq T$, *relevant* topics (attached to one argument) denoted $rel(a,b) \subseteq T$, and *irrelevant* topics (not attached to either argument) denoted $irr(a,b) \subseteq T$.

When the agents express some opinion regarding an attack (in our context by voting, or initially stating the attack), a weight will be attached to that vote. Note that weights are not assigned to agents but to pairs agents-attacks depending on their expertise.

Definition 3.9 (Expertise of an agent)
The **expertise of agent** i is given by a function $exp(i) \subseteq T$.

Experts express their opinions on attacks by casting positive or negative votes.

Definition 3.10 (Vote)
A **vote** is a tuple $\langle (a,b), s, i \rangle$ where $(a,b) \in \mathcal{R}$ is the attack concerned by the vote, $s \in \{-1,+1\}$ is the polarity (sign) of the vote, and i is the voter.

The impact of the vote of an expert i for or against an attack depends on her expertise over the topics of this attack. Intuitively, the opinion of an expert on the topics of an attack should have more importance than the opinion of a non-expert on the same attack. However, this general principle needs to be made much more precise. Some key assumptions need to be made explicit:

- *Independence of expertise*: should two experts in one topic each have the same impact as one expert in both topics voting once?
- *Compensation among topics*: should we allow compensation among levels of topics? For instance, should two votes on a relevant topic be as important as one vote on a prominent one? Should irrelevant topics be considered in the first place?

Here, we make the following simple choices: we suppose that independence of expertise holds, we allow compensation among topics (considering prominent topics to be twice as important as relevant topics) and we disregard votes on irrelevant topics.

We thus propose the following definition of impact, which has two advantages. First, the more topics of an attack an agent is expert in, the greater her impact is on the attack. Second, expertise in prominent topics of an attack leads to a greater impact than expertise in its relevant topics.

Definition 3.11 (Impact)
Let i be an agent. The **impact of** i **on** $(a,b) \in \mathcal{R}$ is denoted $imp_i(a,b)$ and defined by $imp_i(a,b) = 2 \times |exp(i) \cap prom(a,b)| + |exp(i) \cap rel(a,b)|$. If $imp_i(a,b) = 0$ we say that i is a **dummy voter** on (a,b).

Definition 3.12 The **evaluation vector of** $(a,b) \in \mathcal{R}$ is denoted $v(a,b) = \langle w(a,b), mw(a,b) \rangle$, where $w(a,b)$ (called the **weight** of (a,b)) is the aggregated impact of all the experts who have voted for or against (a,b), and $mw(a,b)$ (called the **max-weight** of (a,b)) is the aggregated impact of all the voters on (a,b), assuming they had all voted in favor of it.

Once the experts have expressed their points of view on a subset of attack relations, we obtain an *aggregated argumentation system with weighted attacks (*WAS*)*. To use acceptability semantics of abstract argumentation (Dung, 1995), we need to define the notion of *non-weighted counterpart* AF *of a* WAS. To do so, we simply chose to remove all attacks with non-positive weights.

The procedure we propose then proceeds in three phases. The first phase consists of the aggregation of the different opinions of the agents and allows to obtain an aggregated WAS. Recall that we do not commit to any specific protocol here. Then comes an evaluation phase which allows to determine how controversial the aggregated WAS is. If required, this second phase leads to a third phase where we chose an expert to make the aggregated WAS less controversial.

Phase 1: Experts express their opinions

In this first phase, the agents place their arguments on the board and express their opinions by voting on the attack relations.

Phase 2: Evaluation of the aggregated WAS

Once the agents have expressed their opinion, we obtain an aggregated WAS. The question is then if this WAS is controversial, and to what extent. We want the WAS to be as uncontroversial as possible, to avoid discussions and discontentment about the outcome of the procedure. These disagreements could appear if the majority is not clearly defined, or if the final outcome can be interpreted in several ways.
Several criteria can assess how controversial a WAS is. In the following, we focus on two such criteria:

- *Stability of attacks*: an attack relation can be seen as *stable* if it is difficult to question it, more specifically, if a single expert cannot change its sign.
- *Persistence of arguments' labels*: an argument can be seen as *persistent* if its label (`in`, `out` or `undec`, under a chosen semantics) does not depend on unstable (controversial) attacks, that is if a single expert cannot change its label if she changes some signs of these attacks.

For the first criterion, we consider a natural qualitative scale and introduce three types of attacks. The *beyond any doubt* attacks are the ones on which sufficiently many agents agree (or, as a particular case, no agent has stated them). The *strong* attacks are defined as the attacks which are not beyond any doubt, but such that a single expert cannot change the sign of their weights. Finally, the *weak* attacks are neither beyond any doubt nor strong, thus they are the most controversial ones.
For the second criterion, we consider that a *persistent argument* is an argument whose label remains unchanged, regardless of any changes in the weak attacks. Thus, a persistent argument is an argument whose label cannot be changed by the vote of a single expert.
These two (related) notions allow us to determine to what extent the aggregated WAS is controversial. The next phase of the procedure depends on the result of this analysis. If the debate is too controversial at this point, it could be useful to know how to choose an expert to stabilize the aggregated WAS.

Phase 3: Asking the opinion of an expert

The main difficulty at this phase is that the choice of an expert depends on her expertise, but the decision-maker cannot know the expert's *opinion*. So, we consider an expert i who has not taken part in the discussion so far and ask for her opinion on the WAS. We assume that we know i's topics of expertise, but we do not know *a priori* her opinion on the attacks. We will not ask i's opinion on beyond-any-doubt attacks, as we are certain about them, but only on strong and weak attacks a strong attack can become weak after the expert's vote.
The main difficulty now lies in the comparison of the available experts, to choose the one who can make the WAS as uncontroversial as possible. In particular, we observe that it may not be a good heuristic to select the expert with the highest number of topics of expertise, because these topics may not be the most relevant ones. More surprisingly, we also observe that it may not be appropriate to always prefer an expert who declares a strict superset of topics over another expert, because

the additional impact provided by the extra topics may jeopardize an attack that was considered "strong" before.
It is important at this point to observe that a difficulty we face here is that, when comparing experts, we do not compare two WAS, but two *sets* of possible WAS (those that can be obtained when questioning the experts). This leads to various natural definitions of (strict, easily adapted to weak) dominance:

- *i necessarily dominates j* if any WAS that can be reached by i is "better" than any WAS that can be reached by j.
- *i possibly dominates j* if there exists a WAS that can be reached by i which is "better" than a WAS that can be reached by j.
- *i optimistically dominates j* if the best WAS that can be reached by i is "better" than the best WAS that can be reached by j.
- *i pessimistically dominates j* if the worst WAS that can be reached by i is "better" than the worst WAS that can be reached by j.

Observe that while the necessary dominance guarantees that the WAS obtained will be better, the optimistic and pessimistic dominance do not. However, they provide good reasons to prefer an expert over another one. By "better" we essentially mean in the sense of Pareto.

3.2 Multi-agent Dynamics

We are now interested in the dynamical aspect of *gradual* (or *scoring*) semantics. Please recall that scoring semantics assigns a numerical acceptability degree to each argument and that if one defines a grading-based semantics, then this straightforwardly induces a corresponding ranking-based semantics (see Section 2.2 on page 13). Interestingly, from their very inception, these semantics have been promoted as more natural in contexts, such as online debates (Leite and Martins, 2011). Many gradual semantics have been proposed recently, and their formal properties have been studied (in particular, their axiomatics and their computational properties (Amgoud et al., 2017a)). Our research question in this section is thus the following:[11]

> *If agents indeed reason and interact using some gradual semantics, together with some protocol, how will debates and agents' opinions evolve?*

This is a very general question, and there are a number of assumptions that we wish to make explicit upfront for the sake of clarity:

1. *agent-system coherence*: we assume that the agents' opinions and the system evaluation of the debate are based on the same argumentation semantics;
2. *agreement on the argumentative structure*: while agents may have different opinions because they hold different sets of arguments, they agree on attack relations among those arguments;
3. *independence of agents*: each agent behaves independently of the others, we shall not consider issues of coalitions, communication or influence directly among agents.

Of course, all these assumptions could be discussed. We believe though they constitute a natural starting point for the study of such dynamics — and a sort of minimal relevance test for such semantics in multi-agent settings.
While motivated by naturally occurring debates, our work is normative by nature. We do not claim that agents do indeed use (a variant of) such semantics in practice. We instead study how a system would evolve if agents were designed/enforced to follow such principles. Whether this corresponds to what is observed in real online platforms for instance is an interesting but difficult question that we leave for future work. What we are after instead are findings that could help to design a better

[11] This work has been originally published in (Dupuis de Tarlé et al., 2022)

platform, and at least provide some partial validation of the relevance of using such semantics in that context.

3.2.1 A multi-agent protocol using gradual semantics

As the protocols presented previously, we assume that a specific argument (the issue) plays the role of the main question of the debate, and all arguments must be connected to this issue. We assume that each debate will be characterized by a unique issue-oriented argumentation graph UG, the *universe graph*, containing every argument relevant to the issue of the debate. Each agent is equipped with a private argumentation graph, composed of a subset of nodes of UG, called her opinion graph, and representing her own view of the world. Therefore, all agents agree on the attack relations between the arguments they know of, and all the graphs share the same issue.
We assume that agents evaluate arguments using a grading semantics, namely the *h-categorizer* semantics (Besnard and Hunter, 2001) in its weighted variant proposed by Amgoud et al. (2017a), which is known to satisfy several desirable axioms.

Definition 3.13 (Weighted h-categorizer semantics)
The **weighted h-categorizer** is defined as:

$$Hbs(a) = \frac{w(a)}{1 + \sum_{b \in Att(a)} Hbs(b)}$$

When dealing with non-weighted graphs, we shall simply assume that the weights are 1 for all the arguments. In this case, we retrieve the classical *h*-categorizer definition (see Section 2.4 on page 24).
All agents play simultaneously, *i.e.*, at each step all agents play a move. An agent's move consists in adding to the current public graph (or gameboard) an argument, and all the attacks between this new argument and the ones already present in the public graph. Agents can only play arguments that directly attack an argument already on the gameboard.
Intuitively, this corresponds to the behavior of agents seeing the state of the online debate and adding a direct response to one of the published arguments. This is a mild constraint on the relevance of the moves (Prakken, 2006), allowing to backtrack to any previously stated arguments, although not to construct lines of argumentation which would require to state arguments not explicitly related to the debate in the first place. Each agent can perform only one operation on the state of the game at each step.

Dynamics and agents' strategies

To properly define the rational behavior of the agents, we need to clarify how an agent evaluates the current state of the debate, relative to her own private opinion. It would be too demanding to assume that agents require the value of the debate to be *exactly* as their personal opinion. Instead, we assume there is an interval around this value (the *comfort zone*) that makes them happy with the current outcome of the debate. The size of the comfort zone allows to model to what extent an agent is ready to compromise with her own value.
For every argument a of their own argumentation system, each agent can compute a hypothetical value $H_P(a)$ which corresponds to the value of the issue of the gameboard when adding argument a and all relevant attack relations. Using this hypothetical value, they can evaluate every argument that they know of and determine which of them would (theoretically) improve their satisfaction.
Their possible strategies are then the following:

- if an agent k is not comfortable, she can play any argument present in her argumentation graph, which directly attacks at least one argument of the gameboard and whose hypothetical value is closer to her opinion than the current public graph value.

- if an agent k is comfortable, she can play any argument present in her opinion graph, which directly attacks at least one argument of the gameboard and whose hypothetical value is still contained in her comfort zone.

Note the difference between both situations here: while an agent follows a simple better-response approach when she is not comfortable, we assume when she is that she may continue to exchange arguments as long as this does not make her uncomfortable.
In the end, to select which argument to play, the agents choose randomly among the possible strategies. If the set of possible strategies is empty, the agent does not play. At the end of the turn, every argument that was selected by an agent is added to the public graph, along with all relevant attacks
The game stops when every agent's strategy is to do nothing. As the number of arguments known by the agents is finite, the game trivially always finishes.

Learning process

After a turn, every agent has the possibility of learning the arguments that were played by others and are unknown to her. Learning an argument a means adding a to the set of arguments of the argumentation graph of the agent, adding all the attack relations between a and the arguments of her argumentation graph as they appear in the universe graph, and therefore updating the agent's opinion.
We chose to model the learning process to represent *confirmation bias*. Confirmation bias is a cognitive bias that consists in manifesting a preference towards the information which confirms preconceived ideas and grants less weight to the assumptions that challenge them (Poletiek, 2013). At the end of each turn, the agents have access to the new argument's impact on the public graph, that is the difference in value induced by each argument added on the gameboard. The probability for an agent to learn a new argument is related to the dissatisfaction[12] brought by this argument. Agents have a greater probability p_{favor} to learn arguments that favored their own opinion: those are the arguments whose impact on the public graph was to bring its value closer to the agent's opinion, and thus the effect of these arguments on the public graph decreased the agent's dissatisfaction. Conversely, if the arguments' impact on the public graph was to bring its value further away from the agent's opinion, and thus increased the agent's dissatisfaction, the agent has a probability $p_{against}$ to learn it, with $p_{against} \leq p_{favor}$ to account for the confirmation bias.

Experimental results

We now enumerate several hypotheses that we intuitively expect from our protocol and proceed to check that they are supported experimentally.[13]
While our setting is well-defined for argumentation graphs, most of the debates, as they can be seen in real life or online debate platforms[14] are in the form of trees, with the debated issue at the root. We thus decided to study, especially *issue oriented argumentation trees* (that is, for every argument of the graph, there is one and only one path toward the issue i) to present results specifically relevant for debates.
We hypothesize that the following properties are verified by our protocol :

H1 - Outcome: For a given debate, if the learning probabilities increase, the outcome gets closer to the merged value.

H2 - Flexibility: Increasing the size of the comfort zone increases the agent's satisfaction.

[12] Intuitively, the dissatisfaction captures how well the outcome of the debate matches an agent's personal views.
[13] All the material used for this work is available at https://github.com/LouiseDupuis/ArgumentationProject.
[14] See *e.g.* Debategraph (debategraph.org/home)

H3 - Open Mind: If the learning probability of an agent increases, she will be more satisfied at the end of the debate.

H4 - Strength of the Group: When many agents share the same initial information, they have a greater chance to be satisfied by the final result.

H5 - Power of Knowledge: Agents that know more arguments at the beginning of the game are more satisfied at the end.

H6 - Convergence of Views: The highest the learning probabilities, the lower the distance between the agent's final values.

Two different metrics can be used to assess the satisfaction of agents at the end of a game: the number of agents comfortable, and the average of their respective dissatisfaction.
In the special case of Hypothesis 4 (Strength of the Group), as we lacked a proper way to describe the similarity of groups of distinct agents, we proceeded by creating a certain number of "clones", agents which start the game with the same opinion graph and studied the average dissatisfaction of these clones with the variation of their number. We wanted to see whether big groups of clones had a better chance to sway the debate in their favor.
In the case of Hypothesis 6 (Convergence of Views), we chose to evaluate STD, the standard deviation of the agent's opinions at the end of the game, as a measure of the similarity of these opinions.
To test the effect of the learning process, in the case of Hypotheses 1, 3 and 6, we randomly select a learning probability p_{favor} and we fix $p_{against} = max(0, p_{favor} - 0.1)$. P_L designates the average of these two probabilities.

Each game simulation starts with the generation of the universe graph UG. We generate random issue-oriented argumentation trees using a Prüfer sequence. The profile of the game is built by selecting for each agent a random integer $S \in [2, A]$, the size of the agent's opinion graph, and then drawing S nodes from $\mathcal{A}$ and adding the edges corresponding to the relevant attacks.
For each hypothesis, we ran 1000 debates, with parameters $|N| = 7$ agents and $|\mathcal{A}| = 20$ arguments, and studied the correlation between two values of interest. We report the Pearson correlation coefficient R, as well as the p-value of the correlation p (see *e.g.* (Navarro, 2018)). We consider that $0.50 < |R| < 1$ corresponds to a high correlation, $0.30 < |R| < 0.49$ to a moderate correlation and $|R| < 0.29$ to a low correlation. We consider the null hypothesis (no correlation) to be successfully rejected when $p < 0.01$.
In the following table, we denote AD_{clones} the average dissatisfaction of the group of clones, $|Arg(DG_k)|$ and d_k respectively the number of arguments known at the beginning of the game by agent k and her dissatisfaction at the end of the game. The correlation level is the following: dark green = high, light green = moderate, and yellow = low.

	Variable 1	Variable 2	R	p value
H1	P_L	$\|V_F - V_M\|$	-0,55029	2,44E-80
H2	c_l	N_C	0,680451	4,1E-137
H3	P_L	AD	-0,70346	2,1E-150
H4	Nb of Clones	AD_{clones}	-0,28678	2,19E-20
H5	$\|Arg(DG_k)\|$	d_k	-0,40972	9,3E-38
H6	P_L	STD	-0,6683870	1,2764E-130

As many of the correlations we investigate are negative correlations, many of the R we obtain are negative: for instance, we expect the average dissatisfaction of the agents to *decrease* when the

learning probability increases. The signs of the correlation coefficient we obtain are all consistent with our hypotheses.
In every experiment, the null hypothesis is rejected with a p-value that is much lower than the threshold of 0.01. In the case of Hypotheses 1, 2, 3, 5 and 6 the correlation is high or moderate, and we conclude that these hypotheses are verified experimentally. Figure 3.3 presents the evaluation of the average dissatisfaction AD of the debates obtained when we vary P_L the learning probability of the agent. We observe a clear trend towards reduced agent dissatisfaction, however with a number of outliers indicating that this is not an exact law.

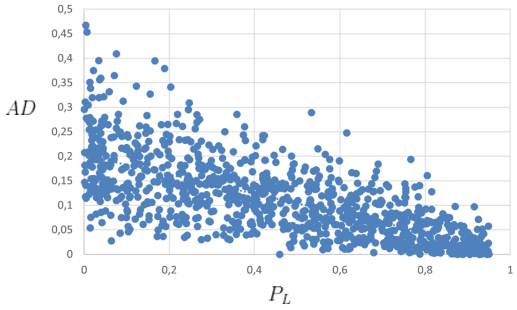


Figure 3.3: H3 - Open Mind: Average dissatisfaction of the agents as a function of learning probability.

In the case of Hypothesis 4, the effect of the presence of clones is not null but is not responsible for a large variation in the satisfaction of agents. Qualitative assessment of individual debates leads us to assume that this is because other factors play a larger role in the outcome of the debate, such as the number of arguments known to the clones. Indeed, as we do not take into account the number of people who play the same arguments, a group of ignorant clones act as a single ignorant agent and cannot prevent a knowledgeable opponent from swaying the game in her favor.
We conclude that our simple protocol empirically exhibits desirable properties. In the next Section we ask ourselves whether this empirical evidence is robust to a modification of the protocol where votes can be expressed by agents.

3.2.2 Adding votes

It is common for online debate platforms to allow their users to cast votes on arguments.[15] We propose an improved version of the protocol where agents can do so, as in (Rago and Toni, 2017) for instance. This approach introduces an element of social validation of the arguments: their value can be dramatically influenced by the amount of social support they receive. This allows the public argumentation framework to better reflect the opinion of all the agents.
Votes can either be positive or negative: an agent votes for an argument if she endorses it or against otherwise. We refer to positive arguments as *upvotes* and negative arguments as *downvotes*. Note that here, endorsing an argument means that the argument belongs to the agent's opinion graph.
Votes are aggregated using a weight function such that the more endorsed or well-accepted an argument is, the greater its weight and arguments with an equal number of upvotes and downvotes weight 0.5. Because of its special status, the issue is not voted for or against and weights 1 throughout the game. In this version, the public graph thus becomes a weighted issue-oriented argumentation graph.

[15] See for example ChangeMyView (https://www.reddit.com/r/changemyview/)

Each step of this protocol is similar to a step of the simpler protocol with the addition of a voting stage *after* the learning stage. When computing the hypothetical value that the public graph would take if they played an argument, agents assign a hypothetical weight of 1 to this argument. Agents vote on the new arguments that were played during the step. Note that the order of the stages is crucial, as agents can vote in favor of arguments they have just learned. After the voting step, votes are aggregated into weights for the arguments in the public graph. This mechanism makes the dynamics of the game more complex.

We have performed an experimental study of the hypotheses presented in Section 3.2.1 on page 80, whose results are presented in the following Table.[16]

	Variable 1	Variable 2	R	p value
H1	P_L	$\|V_F - V_M\|$	-0,0645	0,04
H2	c_l	N_C	0,604745	6,6E-101
H3	P_L	AD	-0,53363	1,39E-171
H4	Nb of Clones	AD_{clones}	-0,23606	3,94E-14
H5	$\|Arg(DG_k)\|$	d_k	-0,40972	9,3E-38
H6	P_L	STD	-0.62242	1.8E-108

Hypothesis 1 is not verified at all by the new protocol. Intuitively, as the public graph is now a weighted argumentation graph, it is not clear whether two weighted graphs can converge as well as a weighted graph and a flat one. All of the other hypotheses remain confirmed, albeit with correlation coefficients that are slightly lower than in the first protocol.

The results of these studies provide some evidence that the studied gradual semantics can be meaningfully used in the context of multi-agent debates over a given issue. On the downside, we showed that the empirical support for some hypotheses decreased (one hypothesis being no longer verified) when we augmented the protocol with votes, which reminds us of the importance of such seemingly minor design choices.

3.3 Argumentation-based Negotiation

We now turn our attention to another application of argumentation: whereas until now we were interested in the dialogues in a context of *persuasion*, we now wonder how to tackle the problem of *negotiation*.

Argumentation-based negotiation frameworks are based on the assumption that agents negotiate through the exchange of arguments. Several works that propose specific (see e.g. Dung et al., 2008; Kakas and Moraitis, 2006a; Parsons et al., 1998) or abstract (see e.g. Amgoud et al., 2007) argumentation-based negotiation frameworks share two important underlying hypotheses:

- The selection of arguments that an agent uses to justify her offer to her opponent or to attack or defend another argument, is based solely on her knowledge about the world and her self-interest
- The knowledge that an agent has about her opponent comes exclusively from their interaction during the negotiation. Therefore, this knowledge is restricted by the information revealed through the arguments used by the opponent.

The above assumptions seem rather counterintuitive. For instance, a competent salesperson is expected to use arguments that are appropriate for the customer, even without any prior interaction between them.

In (Bonzon et al., 2012), we presented a new perspective to argumentation-based negotiation that captures these intuitions in an argumentation-based reasoning mechanism for negotiation, where

[16] Correlation level: Dark green = high, light green = moderate, yellow = low, red = no.

agents use both the knowledge they have about the world as well as the (usually incomplete) knowledge they have about the other agents to make crucial decisions at any time. More precisely this new perspective considers that agents use their own arguments for choosing the offers to propose but, whenever possible, use arguments that are *meaningful* for their opponents to support those offers. This policy is also applied to the arguments that agents use for attacking the opponent's arguments.

3.3.1 The Negotiation Mechanism

The negotiation framework of this work is the one of Hadidi et al. (2010). We assume two agents, a_1 and a_2, who are involved in a bilateral negotiation over a set of offers (options) $\mathcal{O} = \{o_1, ..., o_n\}$. We further assume that there is an option $o_D \in \mathcal{O}$ that represents disagreement. The options are mutually exclusive, which means that each agent can choose only one of them at once.

Arguments

As in (Hadidi et al., 2010), we distinguish two types of arguments that are both taken into account in the reasoning mechanism used by the agents (see Amgoud et al., 2008).

Definition 3.14 (Epistemic and pratical arguments)
The set of arguments $\mathcal{A} = \mathcal{A}_e \cup \mathcal{A}_p$, such that $\mathcal{A}_e \cap \mathcal{A}_p = \emptyset$, with

- $\mathcal{A}_p$ the set of **practical arguments** that support offers by trying to justify those offers
- $\mathcal{A}_e$ the set of **epistemic arguments** that represent what the agent believes about the world

Epistemic arguments are denoted by variables γ_i, while practical arguments are by variables δ_i. When no distinction is necessary between arguments, we use variables $a, b, c...$

We assume that an agent is aware of all the arguments of the set $\mathcal{A}$. It encodes the fact that when an agent receives an argument from another agent, it can interpret it correctly and it can also compare it with its own arguments.
Let $\mathcal{F}$ be a function that maps each offer to the arguments that support it, i.e., $\forall o \in \mathcal{O}, \mathcal{F}(o) \subseteq \mathcal{A}_p$. Each argument can support only one option, thus $\forall o_y, o_z \in \mathcal{O}, o_y \neq o_z, \mathcal{F}(o_y) \cap \mathcal{F}(o_z) = \emptyset$. When $\delta \in \mathcal{F}(o)$, we say that o is the conclusion of δ, noted $Conc(\delta) = o$.

Conflicts between arguments are captured by the binary relation $\mathcal{C}$ (see Amgoud et al. (2008)) such that $\mathcal{C} = \mathcal{C}_e \cup \mathcal{C}_p \cup \mathcal{C}_m$, where

- $\mathcal{C}_e$ represents the conflicts between arguments in $\mathcal{A}_e$
- $\mathcal{C}_p$ represents the conflict between practical arguments, such that $\mathcal{C}_p = \{(\delta, \delta') \mid \delta, \delta' \in \mathcal{A}_p, \delta \neq \delta'$ and $Conc(\delta) \neq Conc(\delta')\}$. This relation is symmetric
- $\mathcal{C}_m$ represents the conflicts between epistemic and practical arguments such that $(\gamma, \delta) \in \mathcal{C}_m$, $\gamma \in \mathcal{A}_e$ and $\delta \in \mathcal{A}_p$.

We assume that practical arguments supporting different offers conflict. Thus for any two offers o_y, o_z, $\forall \delta \in \mathcal{F}(o_y)$ and $\forall \delta' \in \mathcal{F}(o_z)$, it holds that $(\delta, \delta') \in \mathcal{C}_p$ and $(\delta', \delta) \in \mathcal{C}_p$.

As in (Amgoud et al., 2008; Hadidi et al., 2010), we assume three binary preference relations between arguments:

- $\succeq_e$ is a partial preorder on the set $\mathcal{A}_e$.
- $\succeq_p$ is a partial preorder on the set $\mathcal{A}_p$.
- $\succeq_m$ is defined on the sets $\mathcal{A}_e$ and $\mathcal{A}_p$ such that $\forall \gamma \in \mathcal{A}_e, \forall \delta \in \mathcal{A}_p, (\gamma, \delta) \in \succ_m$ and $(\delta, \gamma) \notin \succ_m$. That means that any epistemic argument is preferred over any practical argument.

Each preference relation $\succeq_x$ is combined with the relation of conflict $\mathcal{C}_x$ to give a **defeat relation** between arguments, noted $\mathcal{D}_x$

Definition 3.15 (Defeat relation)
Let $x \in \{e, p, m\}$, and $a, b \in \mathcal{A}$. $(a,b) \in \mathcal{D}_x$ if and only if $(a,b) \in \mathcal{C}_x$, and $(b,a) \notin \succ_x$.
We have $\mathcal{D} = \mathcal{D}_e \cup \mathcal{D}_p \cup \mathcal{D}_m$.

Agents theories

As in Hadidi et al. (2010) we assume that an agent a_i has a *theory* represented abstractly.

Definition 3.16 (Agent's theory)
The **theory** $T_i = \langle \mathcal{A}_i, \mathcal{F}_i, \mathcal{D}_i \rangle$ of agent's a_i consists of

- a set $\mathcal{A}_i = \mathcal{A}_{p,i} \cup \mathcal{A}_{e,i}$ of arguments
- a function $\mathcal{F}_i : \mathcal{O} \to 2^{\mathcal{A}_{p,i}}$ that returns the arguments which support a given offer (and thus $\mathcal{A}_{p,i} = \bigcup_{k=1,\ldots,n} \mathcal{F}_i(o_k)$)
- a defeat relation $\mathcal{D}_i$ on $\mathcal{A}_i$

Moreover, we assume that each agent has also knowledge about the other agent she could negotiate with. This allows us to encode situations where an agent has some insight into her opponent's arguments and preferences, allowing her to use arguments of her opponent to support her favorite offers, which could help her obtain a better agreement.
The knowledge agent a_i has on a_j is denoted by $T_{i,j}$. This theory has the same structure as the agent's a_i own theory but we suppose it to be incomplete, as the knowledge a_i has on a_j is partial. The important part of $T_{i,j}$ is $\mathcal{A}_{i,j}$ which contains the arguments agent a_i knows. This set can be empty if a_i does not know anything about a_j or contains a subset of a_j's arguments. We must note that the knowledge an agent has about her opponent is incomplete but accurate (i.e. as far as arguments, preferences on these arguments and conflicts).

The negotiation protocol

Rubinstein (1982) introduced the *Alternating Offers protocol* for bargaining between agents. This protocol has been adapted in the argumentation-based negotiation context in (Hadidi et al., 2010). This protocol is generic, with no time limits and no central coordinator to manage the negotiations, and either of the parties can leave the process at any time.
Arguments and offers are conveyed through *dialogue moves* (or simply *moves*). These moves use the following performatives:

- `propose` allows an agent to propose an offer to another agent, along with an argument that supports it.
- `argue` allows an agent to argue by defending her own offer; or to counter-attack an offer sent by the other agent.
- `reject` allows an agent to inform the other agent that she has no arguments to present, but that she still does not accept her offer.
- `nothing` allows an agent to notify the other agent that she has no argument to present and she either still considers her offer as a most preferred one for her (when she is the proposer), or believes that she has better options that the current offer (when she is the recipient).
- `accept` is used by an agent to notify that she accepts the offer made by her opponent.
- `agree` allows an agent to state that she believes that her current offer is not optimal for herself and therefore accepts the arguments sent by the other agent, who has to start a new round.

A round takes place in an alternating way between two agents. The agent proposing an offer may send moves with performative from {`propose`, `argue`, `agree`, `nothing`, `withdraw`}, whereas the

agent that receives an offer may send moves with performative from {`argue`, `reject`, `accept`, `nothing`, `withdraw`}. Two outcomes are possible: (a) no agreement (disagreement), or (b) an agreement in some round.

We then implemented a reasoning mechanism that allows agents to use their own negotiation theory to find the best offer to propose to their opponent. This offer is supported by the current "strongest" acceptable (wrt the defeat relation) argument in the agents' theory. Then they use the partial knowledge they have of their opponent to find whether this offer is supported by an acceptable argument in the opponent's argumentation theory. If this is the case, this argument is sent for supporting the proposed offer. Otherwise, they are looking at whether there exists an argument that supports this offer in the opponent's theory, which is not currently acceptable, but which could be defended by their own theories to become an acceptable one. The same policy is also applied to choose the arguments that are used for attacking the arguments of the opponent. However, if such arguments do not exist, agents use the arguments of their own theories for supporting or defending an offer as is done in the frameworks where agents have no knowledge of the opponent.

3.3.2 Experimental evaluation

The experimental evaluation is based on two systems. The first implements the method of Hadidi et al. (2010), which does not utilize any form of knowledge about the opponent agent, whereas the second system is an implementation of our approach.

Agent theories have been generated randomly, as sizeable real-life argumentation theories are not readily available. Random theory generation also facilitates the process of creating structurally diverse theories. Indeed, the experimental suite used in this work includes a variety of agent theories with up to 230 arguments, that differ regarding the relation between the preferences on the epistemic arguments of the negotiating agents, as well as the knowledge an agent possesses about her opponent.

The experimental suite contains test cases that are generated by assigning values to two parameters. The first parameter ($\succeq_e$) concerns the percentage of common preferences between epistemic arguments shared by the agents, with values 100% and 50%. The second parameter ($\mathcal{A}_{i,j}$) concerns the portion of the knowledge (i.e. arguments) each agent has on his opponent, with values 0%, 25%, 50% and 100%.

In the following, R_K denotes the round where an agreement is found by using our system (agents have some knowledge K about each other) and $R_{\neg K}$ the round where an agreement is found with the system of Hadidi et al. (2010) (without knowledge about each other). D_K (resp. $D_{\neg K}$) is the distance between the outcome of the negotiation found with our system (resp. with the system of Hadidi et al. (2010)) and the optimal (or ideal) solution for each agent (see Amgoud and Vesic (2011)). Then, nR_K is the number of negotiations where our system found an agreement in fewer rounds than the system of Hadidi et al. (2010); $nR_{\neg K}$ is the number of negotiations where the system of Hadidi et al. (2010) found an agreement in fewer rounds than our system; nD_K denotes the number of negotiations where the distance of the outcome of the negotiation from the optimal solution is smaller for at least one agent and not worse for the other agent in our system than in the one of Hadidi et al. (2010); $nD_{\neg K}$ is the number of negotiations where the distance of the outcome of the negotiation from the optimal solution is smaller for at least one agent and not worse for the other agent in Hadidi et al. (2010) than in our system. The following table presents the comparative results for the experiments where *both systems have found an agreement* (the number of such negotiations over the 180 experimented per test is given in column *nAgr*). Each test (row) consists of 180 negotiations. The number of arguments involved is between 60 and 230 for each agent's theory.

	nR_K	$nR_{\neg K}$	$nAgr$	nD_K	$nD_{\neg K}$
$\succeq_e$: 100%, $\mathcal{A}_{i,j}$: 100%	45	0	152	5	1
$\succeq_e$: 100%, $\mathcal{A}_{i,j}$: 50%	20	2	152	0	0
$\succeq_e$: 100%, $\mathcal{A}_{i,j}$: 25%	0	0	152	0	0
$\succeq_e$: 100%, $\mathcal{A}_{i,j}$: 0%	0	0	152	0	0
$\succeq_e$: 50%, $\mathcal{A}_{i,j}$: 100%	47	1	141	3	2
$\succeq_e$: 50%, $\mathcal{A}_{i,j}$: 50%	4	2	141	1	0
$\succeq_e$: 50%, $\mathcal{A}_{i,j}$: 25%	0	0	141	0	0
$\succeq_e$: 50%, $\mathcal{A}_{i,j}$: 0%	0	0	141	0	0

The analysis of the experimental results summarized in this table gives us useful information about (1) the usability in practice of argumentation-based negotiation and the way it computationally behaves while scaling in a bilateral negotiation context (2) the performance of our approach.
Concerning the first point, this work is (as far as we know) the first one to empirically show that a Dung-based abstract preference-based argumentation framework behaves computationally well while scaling in a bilateral negotiation context. We ran 1440 negotiation experiments which, when resulted in an agreement, did so in reasonable execution times. More precisely the average time for an agreement was between 10s and 15s for a size of 60 arguments for each agent theory and 45s for a size of 230 arguments for each agent theory. Moreover, for more than 1600 negotiations, the number of arguments used by each agent reached 230 which, we believe, is sufficient to model and implement real-life applications.
Concerning the second point, the results show that our system improves the performance of the system of Hadidi et al. (2010) regarding two important criteria, namely the *length of the negotiation* when there is an agreement and the *quality* of the agreement. More precisely concerning the criterion of length, the use of knowledge about the other agent has, (no matter what the % of knowledge about the other agent is), a significant positive impact on the negotiation shortening. This can be important, especially for time-constraint negotiations.
Finally, it is worth noting that both systems find exactly the same solution and in the same round when in our system there is no knowledge at all on the opponent.

3.4 Are agent systems consistent?

Different agents may have different points of view, and that can be modeled through different abstract argumentation frameworks. The problematic of aggregating several argumentation frameworks, either in the context of argumentative protocols as seen all along this chapter, or to obtain a single collective argumentation framework that would appropriately represent the views of the group as a whole (see eg. Bodanza and Auday, 2009; Coste-Marquis et al., 2007; Dunne et al., 2012; Tohmé et al., 2008) is a form of graph aggregation (Endriss and Grandi, 2017). We are given a profile of attack relations, one for each agent, and are asked to compute a suitable compromise attack relation. This is an interesting and fruitful line of research, bringing together concerns in abstract argumentation with the methodology of social choice theory, but it raises one important question: For a given profile of argumentation frameworks, is it conceivable that such a profile would manifest itself? That is, how do we explain the differences in perspective of the individual agents for a given profile? Why do they sometimes report different arguments? And why do they sometimes report different attacks even between those arguments they agree on?
The point that the attack relation should not be viewed as absolute and objective, but may very well depend on the individual circumstances of the agent considering the arguments in question, has been made before by multiple authors (e.g., Amgoud et al., 2008; Baumann, 2012; Bench-Capon et al., 2007; Booth et al., 2013; Gabbriellini and Torroni, 2013; Grossi and van der Hoek, 2013). A widespread explanation for such diversity of views is that agents have different preferences

regarding the arguments at hand. For instance, arguments may come from different sources, which agents may trust to varying degrees. Or the arguments may be attached to different moral or social values, which the agents may prioritize differently. This perspective still assumes an underlying ground truth, which however may be interpreted differently, depending on the agents.

Here we adopt a preference-based approach, in the *value-based* variant due to Bench-Capon (2003). In his model, whether argument A ultimately defeats argument B does not only depend on whether A attacks B in an objective sense, but also on how we rank the importance of the moral or social values attached to A and B: If we rank the value associated with B strictly above that associated with A, we may choose to ignore any attacks of A on B. Thus, differences in their preferences can explain why different agents may report different attacks.

Regarding the fact that agents may also report different sets of arguments, to begin with, the most natural explanation is simply that the agents are not all *aware* of the same arguments. However, depending on the context, it may sometimes also be reasonable to assume that an agent *chooses*, on purpose, not to report certain arguments. For instance, it may be the case that certain values are 'taboo' for some agents and that they prefer not to refer to them and thus choose to suppress any arguments relating to those values.[17] Or agents may choose to ignore arguments they consider irrelevant to minimize communication.

At the technical level, the question we asked in (Airiau et al., 2016, 2017) is the following: Given a profile of argumentation frameworks $\langle \mathsf{AF}_1, \ldots, \mathsf{AF}_n \rangle$, one for each agent, defined over possibly different sets of arguments, can this profile be explained in terms of a single master argumentation framework, an association of arguments with values, and a profile of preference orders over values $(\succcurlyeq_1, \ldots, \succcurlyeq_n)$, one for each agent? Or, as we shall put it: Can the profile of argumentation frameworks observed be *rationalized*? To be able to answer this question in the affirmative, for every agent i, we require AF_i to be exactly the argumentation framework we obtain when the master argumentation framework with its associated values is first restricted to the arguments agent i is aware of and then any attacks that are in conflict with the preference order $\succcurlyeq_i$ are being canceled.

First and foremost, following Bench-Capon (2003), we define an *audience-specific value-based argumentation framework* (AVAF) as an AF equipped with a function associating each argument with the social or moral value it advances, combined with a preference order declared over those values. While the mapping from arguments to values is fixed and the same for everyone, the preferences over values are those of a particular agent (the "audience").

Definition 3.17 (AVAF)
An **audience-specific value-based argumentation framework** is defined as a 5-tuple $\langle \mathcal{A}, \mathcal{R}, \mathit{Val}, \mathit{val}, \succcurlyeq \rangle$, where $\langle \mathcal{A}, \mathcal{R} \rangle$ is an argumentation framework, Val is a finite set of values, $\mathit{val} : \mathcal{A} \to \mathit{Val}$ is a mapping from arguments to values, and $\succcurlyeq$ is the audience's preference order on Val.

We call $\langle \mathit{Val}, \mathit{val} \rangle$ the AVAFs *value-labeling*. Let $=_{\mathit{val}}$ be the equivalence relation on arguments induced by val: $A =_{\mathit{val}} B$ if and only if $\mathit{val}(A) = \mathit{val}(B)$.

Now suppose an agent is presented with an AF and a value-labeling. In Bench-Capon's model, this agent will uphold a proposed attack $(A, B) \in \mathcal{R}$ and therefore accept that A *defeats* B, unless she strictly prefers the value associated with the attackee B to the value associated with the attacker A (Bench-Capon, 2003).

Definition 3.18 (Defeated Arguments)
Given an AVAF $\langle \mathcal{A}, \mathcal{R}, \mathit{Val}, \mathcal{V}^{\succcurlyeq}_{,} \rangle$, we say that argument $A \in \mathcal{A}$ defeats argument $B \in \mathcal{A}$, denoted $(A, B) \in \mathcal{D}$, if and only if we have $(A, B) n \mathcal{R}$ but it is not the case that $\mathit{val}(B) \succ \mathit{val}(A)$.

We call $\mathcal{D}$ the defeat-relation *induced* by the AVAF. We stress that saying 'it is not the case that

[17] Similar ideas have been explored for the definition of semantics that attempts to only make use of certain arguments if absolutely necessary (Cayrol et al., 2002).

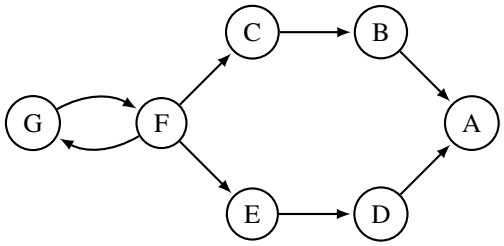
$val(B) \succ val(A)$' is the same as saying '$val(A) \succcurlyeq val(B)$ is the case' only when the preference order $\succcurlyeq$ is complete.

Note that for any given AVAF $\langle \mathcal{A}, \mathcal{R}, Val, val, \succcurlyeq \rangle$ the induced defeat-relation $\mathcal{D}$ is, just like an attack relation $\mathcal{R}$, an irreflexive binary relation on $\mathcal{A}$. Thus, we can (and will) think of $\langle \mathcal{A}, \mathcal{D} \rangle$ as just another AF.

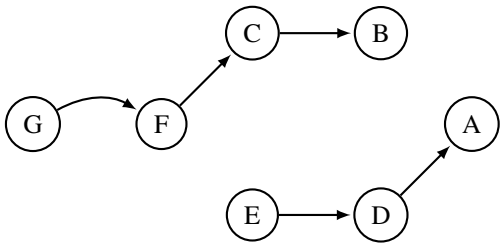
Example 3.4 Pollution is becoming a major health problem in big cities. City councils are facing the question of possibly banning polluting vehicles, specifically diesel cars, from the city centers. A city council might entertain the following arguments:

- (A) Diesel cars should be banned from the inner city center to decrease pollution.
- (B) Artisans, who deserve special protection from the city council, cannot change their vehicles, as that would be too expensive for them.
- (C) The city can offer financial assistance to artisans.
- (D) There are only very few alternatives to using diesel cars. Specifically, the autonomy of electric cars is poor, as there are not enough charging stations around.
- (E) The city can set up more charging stations.
- (F) In times of financial crisis, the city should not commit to spending additional money.
- (G) Health and climate change issues are important, so the city has to spend what is needed to tackle pollution.

The following graph shows the AF generated by these arguments, together with a natural attack relation $\mathcal{R}$ between them:



We can associate the arguments presented in this example with four types of values. Arguments A and G concern environmental responsibility (value `env`), B and C are about social fairness (value `soc`), F promotes economic viability (value `econ`), and D and E pertain to infrastructure efficiency (value `infra`). We thus have that $Val = \{\texttt{env}, \texttt{soc}, \texttt{econ}, \texttt{infra}\}$, as well as that $val(A) = val(G) = \texttt{env}$, $val(B) = val(C) = \texttt{soc}$, $val(F) = \texttt{econ}$, and $val(D) = val(E) = \texttt{infra}$. Let us now assume that a particular councilor wants to promote the values of environmental responsibility and infrastructure efficiency over the other two values. So her preferences might be given by the following weak order: $\texttt{env} \sim \texttt{infra} \succ \texttt{soc} \sim \texttt{econ}$.
This induces a defeat-relation $\mathcal{D}$ for our councilor that corresponds to the following graph:



For instance, the attack from B to A got removed, because $val(A) = \texttt{env} \succ \texttt{soc} = val(B)$. Overall, three attacks got removed.

■

In the sequel, we are going to use standard set-theoretical operations (e.g., $\cap$, $\subseteq$) on binary relations (understood as sets of pairs). Furthermore, $R^{-1} = \{(x,y) \mid (y,x) \in R\}$ is the inverse of a binary relation, R^+ is its transitive closure, and R^* is its reflexive-transitive closure. $R \circ R'$ is the composition of R and R'. We also define $R^+_{val} := (R \cup =_{val})^* \circ R \circ (R \cup =_{val})^*$, which is like the usual transitive closure, except that we can move to arguments with the same value, even if not connected by R. Finally, for any binary relation R defined on some set S, we use $R\restriction_S = R \cap (S \times S)$ to denote the restriction of R to S.

3.4.1 The Rationalizability Problem

Let $N = \{1,\ldots,n\}$ be a finite set of *agents* (or *audiences*). Suppose each of these agents supplies us with an AF, not necessarily over the same set of arguments. We call this a *profile* of AFs.
As we think of each AF in such a profile as the result of having imposed the corresponding agent's preferences on some underlying master AF, we write individual AFs as $\langle \mathcal{A}_i, \mathcal{D}_i \rangle$ (rather than as $\langle \mathcal{A}_i, \mathcal{R}_i \rangle$). Here, $\mathcal{A}_i$ is the set of arguments agent i is *aware* of and $\mathcal{D}_i$ is the defeat-relation on $\mathcal{A}_i$ adopted by i. A profile of such AFs is denoted as $\mathbf{AF} = (\langle \mathcal{A}_1, \mathcal{D}_1 \rangle, \ldots, \langle \mathcal{A}_n, \mathcal{D}_n \rangle)$. Let $\mathcal{A} := \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_n$ denote the set of all arguments at least one agent is aware of.
Now we may ask whether the profile we observe can be *rationalized* (i.e., whether it can be explained) in terms of a common master AF and a common value-labeling, together with a profile of preference orders, one for each agent. This question gives rise to the *rationalizability problem*. We define an entire family of rationalizability problems, parameterized by a set of *constraints* imposed on the solutions admitted.

Definition 3.19 (Rationalizability)
A profile of AFs, $\mathbf{AF} = (\langle \mathcal{A}_1, \mathcal{D}_1 \rangle, \ldots, \langle \mathcal{A}_n, \mathcal{D}_n \rangle)$, is called **rationalizable** for a given set of constraints if there exists an attack relation $\mathcal{R}$ on $\mathcal{A} = \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_n$, a set of values *Val* with a mapping $val : \mathcal{A} \to Val$, and a profile $(\succcurlyeq_1, \ldots, \succcurlyeq_n)$ of preference orders on *Val*, all meeting said constraints, such that, for all agents $i \in N$ and all arguments $A, B \in \mathcal{A}_i$, it is the case that $(A,B) \in \mathcal{D}_i$ if and only if $(A,B) \in \mathcal{R}$ but not $val(B) \succ_i val(A)$.

We refer to $\langle \mathcal{A}, \mathcal{R} \rangle$ as the *master AF*, and consequently to $\mathcal{R}$ as the *master attack relation*.
Some comments on how to interpret Definition 3.19 are in order. Given the presumed existence of $\langle \mathcal{A}, \mathcal{R} \rangle$, $val : \mathcal{A} \to Val$, and $(\succcurlyeq_1, \ldots, \succcurlyeq_n)$, we think of the observed profile $\mathbf{AF} = (\langle \mathcal{A}_1, \mathcal{D}_1 \rangle, \ldots, \langle \mathcal{A}_n, \mathcal{D}_n \rangle)$ as having come about as the result of the following process:

1. Each agent $i \in N$ becomes aware of some subset $\mathcal{A}_i \subseteq \mathcal{A}$ of the full set of arguments, and thus of the AF $\langle \mathcal{A}_i, (\mathcal{R})\restriction_{\mathcal{A}_i} \rangle$, i.e., of the restriction of the master attack relation to the set of arguments she is aware of
2. Agent i removes any attacks in this AF that are at odds with her preferences, i.e., we get $(A,B) \in \mathcal{D}_i$ for $A, B \in \mathcal{A}_i$ if and only if $(A,B) \in (\mathcal{R})\restriction_{\mathcal{A}_i}$ but not $val(B) \succ val(A)$

Thus, we have made a specific modeling choice when defining rationalizability: We assume that agents *first* choose (possibly unconsciously) the subset of arguments to report, and only *then* reduce the attack relation defined on that subset according to their individual preferences. Another option would have been to assume that the agents first reduce the master attack relation according to their own preferences, and then choose a subset of arguments to report (e.g., those that they deem most relevant or significant).
Definition 3.19 can be instantiated for different types of *constraints*. In the following, we are going to consider the following constraints (but others may be of interest as well):

- the master attack relation $\mathcal{R}$ may be fixed,
- the value-labeling $\langle Val, val \rangle$ may be fixed,
- the number of values $|Val|$ may be bounded from above by some constant k,

- the preference orders $\succcurlyeq_i$ may be required to be complete.

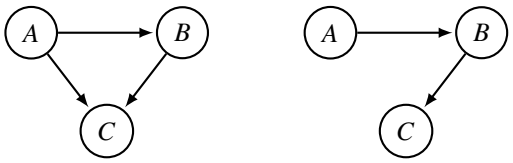
In addition, we will consider two *restrictions* of the problem, namely the case where all individual argument sets coincide (i.e., where $\mathcal{A}_i = \mathcal{A}_j$ for all $i, j \in N$), and the single-agent case (with $n = 1$).

With these definitions in place, we may now ask: For a given set of constraints, can we characterize the class of all profiles of AFs that can be rationalized? And can we check efficiently whether a given profile is rationalizable?

3.4.2 The Single-Agent Case

We first consider the single-agent case of the rationalizability problem. This is not only useful for gaining an understanding of the multi-agent case but is also interesting in its own right. For example, it may be the case that there is some 'ground truth' available and we know what the correct attack relation is (e.g., due to the logical structure of the arguments), but that a specific agent is still reporting a different AF. Can this subjective AF be explained in terms of the value-based model? That is, is this framework compatible with what we know to be the ground truth?

Example 3.5 Consider a scenario with three arguments, $\mathcal{A} = \{A, B, C\}$, with a fixed master attack relation $\mathcal{R} = \{(A,B),(B,C),(A,C)\}$. Suppose we observe a single agent who only declares $\mathcal{D} = \{(A,B),(B,C)\}$. Below, the master attack relation is shown on the left and the observed individual attack relation is shown on the right:



Can we rationalize this omission of the attack of A on C? Rationalization requires A and C to be labeled with distinct values, say v_A and v_C, and our agent must prefer v_C to v_A for $(A,C) \in \mathcal{R}$ to get canceled. Are two values enough? The answer is no: If we reuse, say, value v_A to also label argument B, then $(B,C) \in \mathcal{R}$ would get canceled as well. Similarly, if we reuse v_C for B, then $(A,B) \in \mathcal{R}$ would get cancelled. Thus, we need a third value v_B. Now there is a rationalization, with the agent's preference order ranking v_C above v_A, and v_B being incomparable to the other two values. Observe that, even with three values, rationalization is impossible if we require the preference order to be complete, i.e., if we require it to not leave any two values incomparable. ■

In the single-agent case, we are given an AF $\langle \mathcal{A}, \mathcal{D} \rangle$. A solution consists of an AVAF $\langle \mathcal{A}, \mathcal{R}, \mathit{Val}, \mathit{val}, \succcurlyeq \rangle$, over the same set of arguments $\mathcal{A}$, that induces $\mathcal{D}$. We consider this problem for several types of constraints on solutions. We aim to provide polynomial-time algorithms for computing solutions and, where possible, to provide exact characterizations of those solutions. We begin with the simplest of all scenarios, where there are no constraints imposed on permissible rationalizations, and observe that the problem of rationalizability is trivial in this case:

Proposition 3.8 (No Constraints)
In the absence of constraints, every single AF is rationalizable.

The same result applies to rationalization under any set of constraints referring only to *Val* and *val*. It also continues to apply if we require the preference order to be complete. The main insight here is that any natural instance of the single-agent problem that is nontrivial will involve a constraint on the master attack relation. Therefore, we consider rationalizability problems with a given fixed master attack relation.

Proposition 3.9 (Fixed Attack Relation)

A single AF $\langle \mathcal{A}, \mathcal{D} \rangle$ is rationalizable by an AVAF with a given fixed master attack relation $\mathcal{R}$ if and only if all of the following three conditions are satisfied:

(i) $\mathcal{D} \subseteq \mathcal{R}$;

(ii) $(\mathcal{R} \setminus \mathcal{D})$ is acyclic;

(iii) $\mathcal{D} \cap (\mathcal{R} \setminus \mathcal{D})^+ = \emptyset$.

All three conditions can be checked in polynomial time, so we obtain a tractability result:

Corollary 3.1 (Fixed Attack Relation)
Whether a single AF is rationalizable by an AVAF with a given fixed master attack relation can be decided in polynomial time.

We now investigate what happens when we add such constraints, and first, consider the most extreme case where the full value-labeling is fixed from the outset. This is a natural scenario to consider in those cases in which we are willing to assume that the question of which value a given argument relates to is a matter that can be settled objectively.

Proposition 3.10 (Fixed Value-Labeling)
A single AF $\langle \mathcal{A}, \mathcal{D} \rangle$ is rationalizable by an AVAF with a given fixed master attack relation $\mathcal{R}$ and a given fixed value-labeling $\langle Val, val \rangle$ if and only if all of the following three conditions are satisfied:

(i) $\mathcal{D} \subseteq \mathcal{R}$;

(ii) the relation $\bigcup_{(A,B) \in (\mathcal{R} \setminus \mathcal{D})} \{(val(A), val(B))\}$ is acyclic;

(iii) $\mathcal{D} \cap (\mathcal{R} \setminus \mathcal{D})^+_{val} = \emptyset$.

This characterization immediately provides us with a polynomial-time algorithm. Thus, we obtain the following result:

Corollary 3.2 (Fixed Value-Labeling)
Whether a single AF is rationalizable by an AVAF with a given fixed master attack relation and a given fixed value-labeling can be decided in polynomial time.

The final single-agent scenario we want to consider here is one where we are not given the full value-labeling but merely an upper bound on the number of values that may be used for rationalization.[18] This scenario comes about when there is no unique objective mapping from arguments to values and we are looking for a "simple" explanation for an observed defeat relation only involving a limited number of different values. (For instance, when values correspond to different sources providing information, their number may be known.) From an algorithmic point of view, this is the most demanding problem considered so far. Still, at least for the case of complete preferences, also for this problem we can establish the existence of a polynomial-time algorithm, as the following result shows.

Proposition 3.11 (Bound on Values)
Whether a single AF is rationalizable by an AVAF with a given fixed master attack relation, a given upper bound on the number of values, and a complete preference order can be decided in polynomial time.

Assuming completeness of the preference order (i.e., excluding the possibility of an agent not being able to compare the importance of two given values) is sometimes reasonable, but certainly not always. Whether single-agent rationalizability for a bounded number of values remains polynomial

[18] Thus, this scenario requires solving the decision problem corresponding to the optimization problem of computing the minimal number of values needed for rationalization.

for possibly incomplete preferences is a nontrivial open question of some interest.

Recall that Example 3.5 on page 91 has demonstrated that there indeed are single-agent scenarios where rationalization is possible with incomplete preferences but impossible with complete preferences. We conclude this section with one further observation on the impact the choice of preference order can have on rationalizability. We show that even when rationalization is possible with a complete preference order, imposing that requirement may radically increase the number of values we need to use.

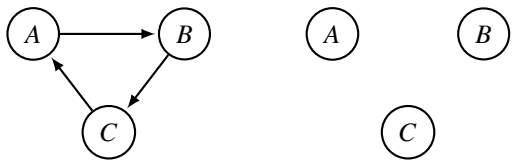
Proposition 3.12 (Value Ratio)
The ratio between the number of values required to rationalize a given AF by an AVAF with a given fixed master attack relation and a complete preference order and the number of values required in case the requirement to use a complete preference order is dropped, in the worst case, cannot be bounded from above by a constant.

Thus, in principle, the number of values required for rationalization can grow arbitrarily when we exclude the possibility of an agent declaring two values preferentially incomparable.

3.4.3 The Multi-agent Case

We now turn to the multi-agent case. In presenting our results for each type of constraint considered, we are specifically going to focus on the extent to which the (positive) results obtained for the single-agent case carry over to this more general scenario. To get started, recall that we have seen that, in the absence of constraints, *every* single AF can be rationalized (Proposition 3.8 on page 91). The following example shows that this result does not generalize to profiles with (at least) two AFs.

Example 3.6 Consider a profile of two AFs over a common set of three arguments. Suppose $\mathcal{D}_1 = \{(A,B),(B,C),(C,A)\}$, while $\mathcal{D}_2 = \emptyset$, as shown below:



Any value-labeled AF and preference profile that could rationalize this profile would have to have an attack relation $\mathcal{R}$ that includes, at least, the attacks (A,B), (B,C), and (C,A), as otherwise, these edges could not have occurred in the first AF. But this means that the second preference order, to be able to cancel these attacks, must at least include the comparisons $val(B) \succ_2 val(A)$, $val(C) \succ_2 val(B)$, and $val(A) \succ_2 val(C)$. But then $\succcurlyeq_2$ is not acyclic, thereby contradicting our assumptions on well-defined preference orders. Thus, this profile cannot be rationalized, even in the absence of any kind of constraint. ∎

We are first going to investigate the question of when we can decompose a given multi-agent rationalizability problem into a set of n single-agent rationalizability problems that can be solved independently of each other (but that still require us to provide a global solution involving a common master AF and a common value-labeling). Example 3.6 shows that this kind of decomposition is *not* possible when we do not impose any constraints during rationalization. On the other hand, for the scenarios of Propositions 3.9 on page 91 and 3.10 on the facing page, it is easy to see that decomposition is possible:

- If the only constraint is that the master attack relation is fixed, then every agent's rationalizability problem can be solved independently.
- If the only constraints are that the master attack relation and the value-labeling are fixed, then every agent's rationalizability problem can also be solved independently.

But what if the master attack relation is not given? Consider the generic profile $\mathbf{AF} = (\langle \mathcal{A}_1, \mathcal{D}_1 \rangle, \ldots, \langle \mathcal{A}_n, \mathcal{D}_n \rangle)$. Any rationalization of $\mathbf{AF}$ must involve a master attack relation $\mathcal{R}$ with $\mathcal{R} \supseteq \mathcal{D}_1 \cup \cdots \cup \mathcal{D}_n$, because no agent can create an edge not already included in $\mathcal{R}$. Any additional edges in $\mathcal{R}$ will make rationalization only harder if they make a difference at all. Thus, rationalization is possible at all if and only if rationalization is possible with the fixed master attack relation $\mathcal{R} := \mathcal{D}_1 \cup \cdots \cup \mathcal{D}_n$. Given these insights, together with Corollaries 3.1 on page 92 and 3.2 on page 92, we obtain the following result:

Proposition 3.13 (Decomposable Cases)
Whether a profile of AFs is rationalizable can be decided in polynomial time by solving the problem independently for each agent, in at least the following cases:

(a) No constraints are given.

(b) The master attack relation and/or the value-labeling is fixed.

Thus, of all the constraints we have considered here, only the one specifying an upper bound on the number of values leads to a 'genuine' multi-agent rationalization problem. Let us now consider this problem in some detail.

For the remainder of this section, we are always going to assume that a fixed master attack relation $\mathcal{R}$ is part of the constraints considered. By our reasoning above, any tractability result obtained under this assumption immediately extends to the case where no master attack relation is specified.

Our first result on multi-agent rationalization with a bound on the number of values to be used is negative: in the most general case, this problem is intractable.

Proposition 3.14 (Bound on Values: General Case)
Deciding whether a profile of AFs is rationalizable by an AVAF with a given fixed master attack relation and a given upper bound (of at least 3) on the number of values is an NP-complete problem.

This is bad news. But are there special cases where rationalizability is tractable after all? Indeed, if all agents are aware of the exact same set of arguments *and* if we are allowed to use at most two values, then deciding rationalizability is tractable:

Proposition 3.15 (Two Values and Common Argument Sets)
Whether a profile of AFs over a common set of arguments can be rationalized by an AVAF with a given fixed master attack relation and using at most two values can be decided in polynomial time.

Proposition 3.15 suggests that rationalizability becomes easier when the argument sets reported by the agents are all exactly the same. The following simple observation shows that the opposite is also true: if the argument sets are all radically different from each other, then rationalization also becomes easy.

Proposition 3.16 (Mutually Exclusive Argument Sets)
Any profile of AFs of the form $\mathbf{AF} = (\langle \mathcal{A}_1, \mathcal{D}_1 \rangle, \ldots, \langle \mathcal{A}_n, \mathcal{D}_n \rangle)$, with $\mathcal{A}_i \cap \mathcal{A}_j = \emptyset$ for all pairs of agents $i, j \in N$, can be rationalized using just a single value.

If all agents report mutually disjoint sets of arguments and we are given a fixed master attack relation, we might require more than just one value to achieve rationalization. Note that this case is covered by Proposition 3.13.

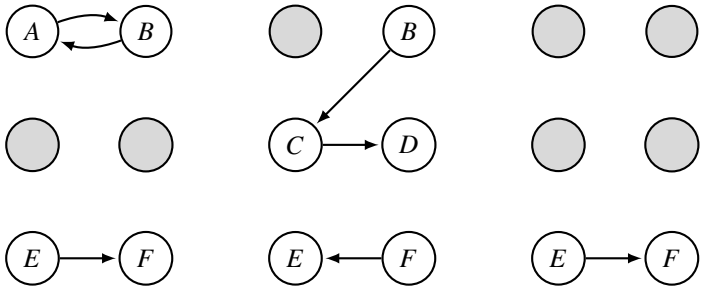
Recall that in case there is no bound on the number of values (or, equivalently, if $k = |\mathcal{A}|$), we already know that rationalization is tractable (as this follows from Proposition 3.13). Finally, we show that the rationalizability problem remains tractable when the bound k is 'large', in the sense of only reducing the number of allowed values by a constant d (relative to the maximum $k = |\mathcal{A}|$).

Proposition 3.17 (Large Bound on Values)
Let $d \in \mathbb{N}$ be an arbitrary constant. Whether a profile $\mathbf{AF} = (\langle \mathcal{A}_1, \mathcal{D}_1 \rangle, \ldots, \langle \mathcal{A}_n, \mathcal{D}_n \rangle)$ is rationalizable by an AVAF with a given fixed master attack relation and at most $k := |\mathcal{A}_1 \cup \cdots \cup \mathcal{A}_n| - d$ values can be decided in polynomial time.

3.4.4 Application Scenarios

There are several different application scenarios where dealing with questions of rationalizability will be valuable. In this section, we list and illustrate some of them.
First, given the growing interest in the abstract argumentation research community in questions of aggregation of AFs, it is important to have a clear understanding of what types of scenarios the question of aggregation is in fact relevant. Our notion of rationalizability provides a suitable definition for this purpose. It allows for a systematic scan of the different examples used in the literature – not to dismiss those failing the test, but to point out that one must be careful with the interpretation used.

Example 3.7 Let us see whether the example given by (Coste-Marquis et al., 2007, Example 7) passes this test. We are given $\mathsf{AF}_1 = \langle \{A,B,E,F\}, \{(A,B),(B,A),(E,F)\} \rangle$, $\mathsf{AF}_2 = \langle \{B,C,D,E,F\}, \{(B,C),(C,D),(F,E)\} \rangle$, and $\mathsf{AF}_3 = \langle \{E,F\}, \{(E,F)\} \rangle$:

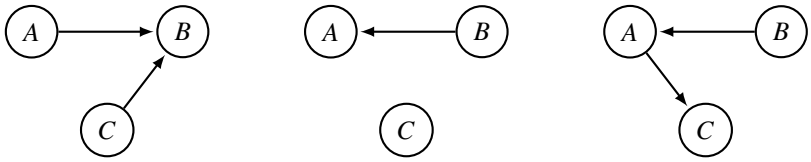


This profile indeed does pass the test. It can be rationalized by using as master attack relation the union of the individual relations. It is sufficient to set $v_E \neq v_F$, while A, B, C, and D can all take the same value, either that of E or that of F. Thus, two values suffice. ∎

The second application of our work concerns aggregation itself. In a scenario where multiple AFs need to be aggregated, we may use the notion of rationalizability to choose between alternative aggregation techniques, depending on the result of the rationalizability test. For example, if a profile turns out to be rationalizable for a given preference model (e.g., for complete preference orders), we may reasonably assume that this model is a good abstraction of reality and aggregate the AFs by aggregating the inferred preferences (which is a much better-studied problem than that of aggregating AFs). For instance, we may use the well-known Kemeny rule (Kemeny, 1959; Zwicker, 2016) to aggregate the preferences,[19] and then apply the collective preference order obtained to the master attack relation inferred.

Example 3.8 Consider the following profile of AFs, in which each of the three agents reports the same set of arguments $\{A,B,C\}$:

[19] For a given profile of preference orders, the Kemeny rule returns the preference order that minimizes the number of times an agent disagrees on the relative ranking of two alternatives. In other words, the Kemeny rule minimizes the sum of the Kendall tau distances between the outcome and the individual preference orders.



The union of the individual AFs together with the following preferences achieves rationalization: $v_C \succ_1 v_A \succ_1 v_B$, $v_B \succ_2 v_C \succ_2 v_A$, and $v_B \succ_3 v_A \succ_3 v_C$. Observe that this is the only rationalization with complete and strict preferences. The result of applying the Kemeny rule to this profile of preferences is $v_B \succ v_C \succ v_A$ (with a Kendall tau distance of 3 from the collection of preferences). Hence, the aggregated AF with this technique would be the same as the AF of agent 2. ∎

But when rationalization fails, this approach does not make sense, and we should look for a different method of aggregation. In such a case, there is a more substantial disagreement between the agents: maybe the model of preferences has to be changed, maybe the agents differ on the assignment of values to arguments, or maybe the agents interpret the arguments differently. Importantly, failure of rationalization can also provide hints as to where disagreement occurs.

A third application we foresee is in the context of online debating platforms, where value-based argumentation systems already are used as a modeling tool (Pulfrey-Taylor et al., 2011). In this context, AFs are (typically) not obtained via a one-shot process, but rather retrieved interactively, by monitoring the utterances of the participants. Our approach could be used to detect inconsistencies as they occur, and thus to trigger clarification questions on the fly.

Example 3.9 Suppose the following sequence of utterances occurs in a given debate:

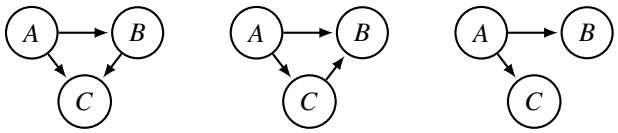
- Agent 1: A defeats B.
- Agent 2: B defeats A.
- Agent 3: There is no defeat between A and B.

At this stage, it is clear that this collection of AFs cannot be rationalized, because agent 3 would have to both prefer the value of A over that of B, and the value of B over that of A. A clarification is required to identify the mismatch. For example, the system could ask agent 3 whether she believes there is no attack between A and B. ∎

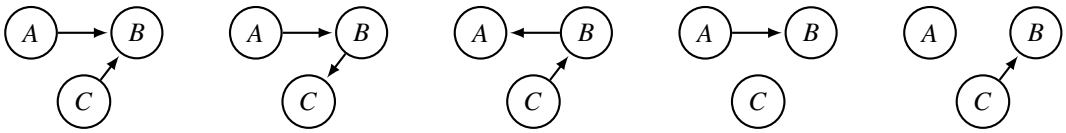
Interestingly, a similar dynamic perspective to solve inconsistencies in a framework mixing opinion polling and argumentation has been proposed by Rago and Toni (2017), albeit in their case the problem is to rationalize the *votes* of users. Interleaving the elicitation of preferences over values within a dialectical process is also proposed by Bench-Capon et al. (2007). But these authors take a different perspective. While we assume that an agent's preferences over values are fixed from the outset and just need to be 'discovered' during rationalization, they do not take this assumption for granted. Instead, the ranking of values is built as part of the dialectical process, whereby an agent (in their case, just one agent) aims at rationalizing her position (i.e., the arguments she wants to see accepted or rejected).

This leads us naturally to our final point of discussion, which concerns the nature of what is observed. So far we have assumed that the agents express AFs, which we can observe directly. But in many situations, it may be more natural to assume that each agent only reports the set of arguments she accepts (a so-called *extension*), or a (partial) *labeling* of the arguments with the labels 'accept' and 'reject'. Dunne et al. (2014) have addressed the challenging problem of inferring an AF from such an extension (or a set of such extensions) that could serve as an explanation for the behavior observed. Of course, there often will be *many* possible AFs that could explain a given set of accepted arguments. Our approach could be used to narrow down the range of possible explanations when performing this task for several agents in parallel, by imposing the constraint that the profile of AFs we infer, one for each extension observed, should be rationalizable.

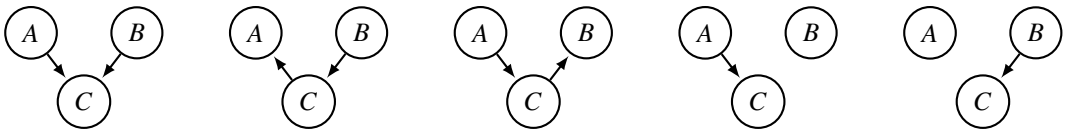
Example 3.10 Suppose there are three agents and three arguments. Agent 1 reports extension $\{A\}$, agent 2 reports extension $\{A,C\}$, and agent 3 reports extension $\{A,B\}$. Suppose these reports have come about through each agent applying one of the well-known semantics proposed by Dung (1995) to some AF declared over the full set $\{A,B,C\}$. For the sake of simplicity, let us exclude the possibility of mutual attacks between pairs of arguments. Then agent 1, who only considers A acceptable, must have one of the following three AFs:



Now, for agent 2, there must be no attack between A and C, while B must get attacked by at least one of them. This leaves five possible cases:



Finally, for agent 3, we can have no attack between A and B. She must have generated her position based on one of the following five AFs:



But now, in terms of rationalization, we see that some combinations are impossible, such as the profile consisting of the third AF for agent 1, the second for agent 2, and the third for agent 3. To see this, note that the master attack relation would have to contain both $(B,C) \in \mathcal{R}$ (for agent 2) and $(C,B) \in \mathcal{R}$ (for agent 3). But then agent 1 would have to have one of these attack relations in her system, as she cannot both strictly prefer the value of B to that of C and *vice versa*. While this does not allow us to uniquely define a profile of AFs, this method can nevertheless guide the search amongst the AFs that are compatible with the extensions observed. ∎

Similar ideas may also have useful applications in the context of analyzing people's decisions *a posteriori*. An example of an application of this kind is the analysis of a participatory decision setting involving an environmental project in Québec, which was carried out by Tremblay and Abi-Zeid (2016). In their work, they first extracted an AF with 20 arguments, labeled by 7 values, from the debates they analyzed. They then imposed a number of technical constraints, eventually obtaining 18 subgraphs of the master AF as possible candidates for the kind of AF that may in practice have guided the deliberations of the committee responsible for taking a decision about which arguments to accept. They then analyzed each of these 18 AFs in combination with one of the possible preference orders over the 7 values, to test whether and how often the decision recommended by a given AF coincides with the decision actually observed in practice. (That decision consisted in accepting 5 of the 20 arguments considered.) To make this analysis manageable, the choice of the 18 AFs considered required several judgment calls. Here, the concept of rationalizability may offer an alternative route. For a set of arguments we observe to have been accepted in practice, we may first induce a number of possible AFs that could explain this extension, using the approach of Dunne et al. (2014), and then apply our rationalization approach to check whether any of these AFs is rationalizable, given the constraints regarding values we have been able to extract from the debate.

4. Interaction in Games

In multi-agent systems, several agents interact, cooperatively or not, to achieve a given objective, that can be personal to each agent, or common to the group of agents. These interactions can take different forms: they can be purely strategic when agents seek to satisfy their own interests, and reason about the other agents' strategy. The fate of each of the agents, that is the satisfaction of her preferences, depends not only on her own decisions but also on the decisions taken by the other members of the system. Therefore, the optimal decision for an individual depends not only on her own actions but also on what other agents are doing. As agents are not in total control of their own fate, we say that they are in *strategical interaction*. In this context, game theory presents an interesting set of tools to model and study these interactions. Agents can also need to interact with each other to take decisions together or agree on a course of action. Here again, game theory, and more especially cooperative game theory, is the natural way to study these problems.
However, game theory is not suitable anymore when agents need to communicate to resolve differences of opinion and conflicts of interest, to work together to resolve dilemmas, to find evidence, or simply to exchange information. In this case, the mechanisms proposed in argumentation theory, that allow agents to present arguments to support their positions, are more suitable.

In this chapter, we are interested in studying the link between argumentation and game theory. More specifically, we asked ourselves how the tools from game theory, which are widely studied, can be used in the setting of abstract argumentation theory. I first came to this idea at the end of my PhD. thesis, which focused on the study of Boolean games, when I realized that there were many similarities between argumentation systems and CP-Boolean games[1]. I wondered first if it was possible to find a translation between both systems and then to identify some links between the pure strategy Nash equilibrium of a CP-Boolean game, and the preferred extensions of the corresponding argumentation system.
Later, we studied how to guide agents as to what argument should be put forward in a debate, and proposed to exploit cooperative game theory to account for the decision-problem faced by an agent in this context.

First and foremost, the next section introduces some background on game theory, which aims at developing mathematical tools to model strategic interactions between several agents. More specifically, Boolean games make it possible to represent strategic games succinctly by taking advantage of the expressivity and the conciseness of propositional logic.
Once again, our goal here is not to give an exhaustive state of the art on game theory but to introduce some concepts that will be useful in the rest of this chapter.[2]

[1] See Section 4.2.3 on page 106.

[2] But there are some great books on this subject if the reader is interested in Game Theory, for example (Osborne, 2004; Osborne and Rubinstein, 1994).

4.1 Basic concepts of Game theory

Game theory is a mathematical approach to studying the behavior - planned, real, or justified a posteriori - of individuals faced with situations of antagonism. If the precursors were Cournot (1838) and Edgeworth (1897), it was the publication in 1944 of the book by von Newmann and Morgenstern (1944), *The Theory of Games and Economic Behaviour*, which truly established game theory as a new discipline. In this work, von Newmann and Morgenstern proposed a solution in the particular case of a *zero-sum game* (what is won by one is lost by the other, and vice versa). In 1950, Nash (1950) showed how the ideas developed by Cournot could serve as the basis for constructing an equilibrium theory for non-zero-sum games, which generalizes the solution proposed by Von Neumann and Morgenstern.
Game theory studies situations in which the fate of each participant depends not only on the decisions she makes but also on the decisions made by other participants. The optimal choice for an agent (called **player**) therefore generally depends on the choices of the other agents. As each player is not totally in control of her own fate, we say that the agents are in a situation of *strategic interaction*.
It is assumed in such games that the players know each other: they know how many players there are, and who they are. As the final outcome for each player depends on the actions of the others, each player must get an idea as precise as possible of the strategies chosen (or likely to be chosen) by the other players. For this, we assume that **the agents are rational**, that is that each player strives to make the best decisions for herself, and knows that the other players do the same.

4.1.1 Game taxonomy

Succinctly, a game is a set of players, a set of strategy profiles (*i.e.* a vector of strategies, one strategy for each player representing a possible choice for her) and a utility function that gives each player her profit according to each strategy profile. Each strategy profile is called **an issue of the game**. A game can be static or dynamic; and cooperative or non-cooperative.

Static games: A game is static if players choose their strategies simultaneously, and then receive their respective utility.

Dynamic games: A game is dynamic if it proceeds in several steps. In a **dynamic game with perfect information**, each player knows all past players' choices and knows all possible strategies of every player. If several players choose their actions simultaneously at a given step, or if players do not know every strategy of the other players, the game is **dynamic with imperfect information**.[3]

Cooperative games: A game is cooperative if players can make agreements and form coalitions[4] to achieve together their goals. A cooperative game has **transferable utilities** if it is possible to add players' utilities and to distribute this sum to members of a coalition (there exists a "common currency" with which one can make transfers) and has **non-transferable utilities** otherwise.

Non-cooperative games: A game is non-cooperative if players cannot make agreements. Non-cooperative games can be divided into two cases: zero-sum games and non-zero-sum games. **Zero-sum games** are games in which the "algebraic" sum of players' profits is a constant: what a player gains is necessarily lost by another player.

The prisoner's dilemma is a famous example of a static, non-cooperative and non-zero-sum game.

[3] This notion of perfect and imperfect information does not make sense in static games.
[4] A coalition is a subset of players.

Example 4.1 Two suspects are arrested by the police. Each prisoner is in solitary confinement and cannot communicate with the other. The prosecutors lack sufficient evidence to convict the pair on the principal charge, but they have enough to convict both on a lesser charge. The prosecutors offer each prisoner the same deal. Each prisoner is given the opportunity either to betray the other by testifying that the other committed the crime or to cooperate with the other by remaining silent. The possible outcomes are:

- If one testifies against her partner and the other remains silent, the betrayer goes free while the silent accomplice receives the full 5-year sentence.
- If both remain silent, both prisoners are sentenced to only one year in jail for a minor charge.
- If each betrays the other, each receives a 3-year sentence.

This problem can be formalized by a two-player game, each player having two possible strategies: to cooperate with (denoted by C) or to defect from (i.e., betray) the other player (denoted by D).

- The set of players is $N = \{1,2\}$
- Prisoner 1 has two possible strategies: $s_{1_1} = C$ and $s_{1_2} = D$. Prisoner 2 has the same possibilities: $s_{2_1} = C$ and $s_{2_2} = D$.
- There are thus four strategy profiles: CC, CD, DC, DD
- The utility functions are the following:
  - $u_1(C,C) = u_2(C,C) = -1$,
  - $u_1(D,D) = u_2(D,D) = -3$,
  - $u_1(C,D) = u_2(D,C) = -5$
  - $u_1(D,C) = u_2(C,D) = 0$.

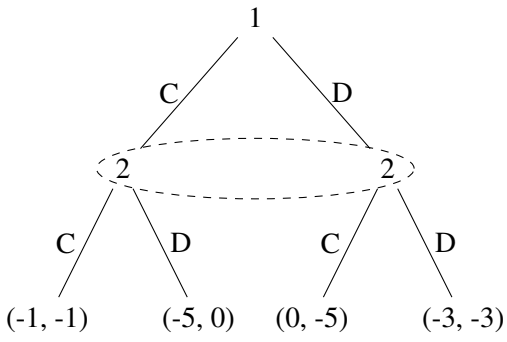
■

4.1.2 Game representation

A strategic game can be represented in two different but equivalent ways: in normal form or extensive form.

The **extensive form** of a game is a **decision tree** describing the possible strategies of each player at each stage of the game. A node of this tree specifies the current player, as well as the information available to her. A branch corresponds to the different alternatives available at each time, and a leaf gives the gains of each player for the corresponding strategy profile.

Example 4.1 **(continuing from p. 101)** An extensive form of the prisoner's dilemma is the following:



In the leaves, the first element of each pair represents Player's 1 utility, while the second one

represents Player's 2 utility.
The dotted circle surrounding the two occurrences of Player 2 means that this player does not know in which situation she is: she does not know if her accomplice has chosen to betray her or to keep quiet.
This game has another extensive form in which the root node corresponds to the second player. As the two players play simultaneously, these two representations are equivalent.

■

The **normal form** of a game gives the set of players, the set of strategies of each player and the profits for each strategy profile. This corresponds to a **matrix form** which associates with each strategy profile s a n-tuple giving the utility obtained by each player: $(u_1(s), u_2(s), \ldots, u_n(s))$.

Example 4.1 **(continuing from p. 101)** The normal form of the prisoner's dilemma is the following:

1 \ 2	C	D
C	(-1, -1)	(-5, 0)
D	(0, -5)	(-3, -3)

■

4.1.3 Solution concepts

There exist many solution concepts in game theory, but I will only present the pure-strategy Nash equilibrium and the Shapley value in this document. Interested readers can refer to the following books for more details (Osborne, 2004; Osborne and Rubinstein, 1994).

Pure-strategy Nash equilibria

Nash equilibrium, introduced by Nash (1950), is a fundamental solution concept in Game Theory. It describes an issue of the game in which no player wishes to modify her strategy given the strategy of each other player. So a Nash equilibrium is a strategy profile where no player may find it beneficial to deviate if she assumes that the other players will not deviate either. There exist several versions of this concept: pure-strategy Nash equilibrium (PNE) or mixed-strategy Nash equilibrium (where probabilities are associated with the strategies).

A **pure strategy Nash equilibrium (PNE)** is a strategy profile such that each player's strategy is an *optimal response* to the strategies of the other players. In other words, a pure strategy Nash equilibrium is a strategy profile where no player may find it beneficial to deviate if it assumes that the other players will not deviate either.

Example 4.1 **(continuing from p. 101)** Strategy profile DD is a pure-strategy Nash equilibrium of the prisoner's dilemma. Indeed, one can check that: $u_1(DD) \geq u_1(CD)$ and $u_2(DD) \geq u_2(DC)$.
This is the only PNE of this game:

- as $u_1(DC) \geq u_1(CC)$, CC cannot be a PNE (if player 2 chooses to remain silent, player 1 would be better off betraying)
- as $u_2(DD) \geq u_2(DC)$, DC cannot be a PNE
- as $u_1(DD) \geq u_1(CD)$, CD cannot be a PNE

■

Existence and unicity of a PNE are not guaranteed.

Shapley value

A **coalitional (or cooperative) game** is a pair (N, v), where N denotes a finite set of players and v is the **characteristic function**, assigning to each $S \subseteq N$, a real number $v(S) \in \mathbb{R}$, with $v(\emptyset) = 0$ by convention. A group of players $T \subseteq N$ is called a **coalition** and $v(T)$ is called the *worth* of this coalition.
Given a game, a **power index** may be computed to convert information about the worth that coalitions can achieve into a personal attribution (of payoff) to each of the players:

$$\pi_i^{\mathbf{p}}(v) = \sum_{S \subset N : i \notin S} p_s \big(v(S \cup \{i\}) - v(S) \big)$$

for each $i \in N$, where p_s represents the probability that a coalition $S \in 2^N$ (of cardinality s) with $i \notin S$ forms.[5]
The **Shapley value** (Shapley, 1953) is a power index $\pi^{\hat{\mathbf{p}}}(v)$ with

$$\hat{p}_s = \frac{1}{n\binom{n-1}{s}} = \frac{s!(n-s-1)!}{n!}$$

for each $s = 0, 1, \ldots, n-1$, where n is the number of players.

4.2 Boolean games

During my PhD, I studied Boolean games, which are a logical setting for succinctly representing static games, taking advantage of the expressive power and conciseness of propositional logic. After my PhD, I began interested in the study of the links between argumentation and Boolean games. The main idea was to use specific tools of game theory, and some particular properties of Boolean games, for computing the extensions of an argumentation framework (Dung, 1995).
I will start by introducing some basics of Boolean games.

4.2.1 Basic concepts on Boolean Games

For static games, extended form and normal form coincide, and utility functions are usually represented explicitly, by listing the values for each combination of strategies. However, the number of utility values which must be specified, that is, the number of possible combinations of strategies, is exponential in the number of players, which renders such an explicit way of representing the preferences of the players unreasonable when the number of players is not very small. This becomes even more problematic when the set of strategies available to an agent consists in assigning a value from a finite domain to each of a given set of variables (which is the case in many real-world domains). In this case, representing utility functions explicitly leads to a description whose size is exponential both in the number of agents ($n \times 2^n$ values for n agents each with two available strategies) and in the number of variables controlled by the agents ($2 \times 2^p \times 2^p$ values for two agents each controlling p Boolean variables). Thus, in all these cases, specifying players' preferences explicitly is unreasonable, both because it would need exponential space, and because studying these games (for instance by computing solution concepts such as pure-strategy Nash equilibria) would require accessing all of these utility values at least once and would take time exponential in the numbers of agents and variables in all cases.
A way out consists in using a language for representing individual preferences (either preference relations or utility functions) on combinatorial (multivariable) domains in a *structured* and *compact* way. These languages exploit to a large extent the structural properties of preferences (such as conditional independencies between variables).

[5] So, coalitions of the same size have the same probability to form

Boolean games introduced in (Dunne and van der Hoek, 2004; Harrenstein, 2004; Harrenstein et al., 2001) are two-player and zero-sum games in which Player 1's utility function (and Player 2's utility function which is its opposite) is represented by a formula in propositional logic, called the Boolean form of the game. These three restrictions (2-player, zero-sum, binary preferences) strongly limit the expressiveness of the framework and were relaxed in (Bonzon, 2007; Bonzon et al., 2006, 2009b).

Let $V = \{a, b, \ldots\}$ be a finite set of propositional variables, a Boolean game with n players on V is a n-player game in which each strategy of each player consists in assigning a truth value to all variables belonging to a subset of V. Each player's preferences are given by a propositional formula φ_i over variables of V.

Definition 4.1 (Boolean game)
A **n-player Boolean game** is a 4-tuple (N, V, π, Φ), with

- $N = \{1, 2, \ldots, n\}$ the set of players (also called agents)
- V a set of propositional variables
- $\pi : N \mapsto V$ a control assignment function which defines a partition of V
- $\Phi = \{\varphi_1, \ldots, \varphi_n\}$ a set of goals, each φ_i being a satisfiable propositional formula of L_V[6]

The *control assignment function* π associates with each player all variables she controls. π_i denotes the set of variables controlled by player i. As each variable is controlled by one and only one player, $\{\pi_1, \ldots, \pi_n\}$ is a partition of V.

The use of Boolean games allows a very compact representation of games. This point is illustrated by the following example which is a simplified variant of the prisoner's dilemma problem.

Example 4.2 Consider n prisoners (denoted by $1, \ldots, n$), and only two kinds of sentence (jail or freedom). The same proposal is made to each of them: "Either you cover your accomplices (denoted by C_i, $i = 1, \ldots, n$) or you denounce them ($\neg C_i$, $i = 1, \ldots, n$). Denouncing makes you free while your partners will be sent to prison (except those who denounced you as well; these ones will also be free). But if none of you chooses to denounce, everyone will be free[7]".
Here is the representation of this game in normal form for $n = 3$:

strategy of 3: C_3		
1 \ 2	C_2	$\neg C_2$
C_1	$(1,1,1)$	$(0,1,0)$
$\neg C_1$	$(1,0,0)$	$(1,1,0)$

strategy of 3: $\neg C_3$		
1 \ 2	C_2	$\neg C_2$
C_1	$(0,0,1)$	$(0,1,1)$
$\neg C_1$	$(1,0,1)$	$(1,1,1)$

So, for n prisoners, we have a n-dimension matrix, therefore 2^n n-tuples must be specified.
This game can be expressed much more compactly by the following Boolean game $G = (N, V, \pi, \Phi)$, where

- $N = \{1, 2, \ldots, n\}$
- $V = \{C_1, \ldots, C_n\}$
- $\forall i \in \{1, \ldots, n\}, \pi_i = \{C_i\}$
- $\forall i \in \{1, \ldots, n\}, \varphi_i = (C_1 \wedge C_2 \wedge \ldots \wedge C_n) \vee \neg C_i$.

■

[6]The notation L_S denotes the subset of L defined on the set of propositional variables S, L being a propositional logical language.

[7]Notice that the limitation to binary preferences makes it impossible to express that a player prefers the situation

The choice of dichotomous utilities (where agents can only express plain satisfaction or plain dissatisfaction, with no intermediate levels) is an important loss of generality. This restriction can be relaxed by replacing the preference component of a Boolean game with an input expressed in a (propositional) language for *compact preference representation*. Languages for preference representation may be either *ordinal* (i.e., expressed by weak orders) or *cardinal* (i.e., expressed by utility functions).

4.2.2 CP-nets

We consider here a very popular language for compact preference representation on combinatorial domains, namely CP-nets.
This graphical model exploits conditional preferential independence to structure the decision maker's preferences under a *ceteris paribus* assumption. They were introduced in Boutilier et al. (1999) and extensively studied in many subsequent papers, most notably (Boutilier et al., 2004a,b).
Although CP-nets generally consider variables with arbitrary finite domains, for the sake of simplicity here we consider only "propositionalized" CP-nets, that is, CP-nets with binary variables (note that this is not a real loss of generality, as all our definitions and results can be easily lifted to the more general case of non-binary variables).
CP-nets are based on the comparison criterion *Ceteris Paribus*: if an agent expresses in natural language a preference such that "a round table will be better in the living room than a square table", she does not want to say that any round table would be better than any square table; she wants to express the fact that she prefers a round table to a square table if they do not significantly differ on their other characteristics. This is the *Ceteris Paribus* principle which leads to the following notion of independence:

Definition 4.2 (Conditionally preferentially independence)
Let V be a set of propositional variables and $\{X,Y,Z\}$ be a partition of V. X is **conditionally preferentially independent** of Y given Z if and only if $\forall z \in 2^Z$, $\forall x_1,x_2 \in 2^X$ and $\forall y_1,y_2 \in 2^Y$ we have:

$$x_1y_1z \succeq x_2y_1z \text{ if and only if } x_1y_2z \succeq x_2y_2z$$

For each variable $X \in V$, the agent specifies a set of *parent variables* $Pa(X)$ that can affect her preferences over the values of X. Formally, X and $V \setminus (\{X\} \cup Pa(X))$ are conditionally preferentially independent given $Pa(X)$. This information is used to create the CP-net:

Definition 4.3 (CP-net)
Let V be a set of propositional variables. $\mathcal{N} = \langle \mathcal{G}, \mathcal{T} \rangle$ is a **CP-net on** V, where $\mathcal{G}$ is a directed graph over V, and $\mathcal{T}$ is a set of **conditional preference tables** $CPT(X_j)$ for each $X_j \in V$. Each $CPT(X_j)$ associates a linear order $\succ_p^j$ with each instantiation $p \in 2^{Pa(X_j)}$.

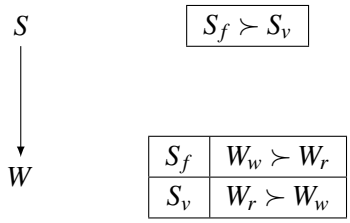
The preference information captured by a CP-net $\mathcal{N}$ can be viewed as a set of logical assertions about an agent's preference ordering over complete assignments to variables in the network. These statements are generally not complete, that is, they do not determine a unique preference ordering.

Definition 4.4 (Induced preference relation)
The **preference relation** over outcomes induced by a CP-net $\mathcal{N}$ is denoted by $\succ_{\mathcal{N}}$, and defined by $\forall o,o' \in 2^V$, $o \succ_{\mathcal{N}} o'$ if and only if $\mathcal{N} \models o \succ o'$.

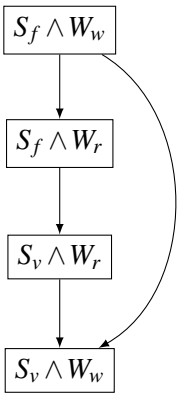
Informally, a CP-net $\mathcal{N}$ is satisfied by $\succ$ if $\succ$ satisfies each of the conditional preferences expressed in the CPTs of $\mathcal{N}$ under the *ceteris paribus* interpretation.

where he denounces and the others cooperate to the situation where everyone cooperates, and the latter to a situation where everyone denounces. To do so we need a more sophisticated language (see Bonzon et al., 2009b).

Example 4.3 Consider the following CP-net about my preferences for dinner. Variables S and W correspond respectively to the soup and the wine. I strictly prefer to eat fish soup (S_f) rather than vegetable soup (S_v), and about wine, my preferences depend on the soup I eat: I prefer red wine (W_r) with vegetable soup ($S_v : W_r \succ W_w$) and white wine (W_w) with fish soup ($S_f : W_w \succ W_r$). So $D(S) = \{S_f, S_v\}$ and $D(W) = \{W_r, W_w\}$.



The preference relation induced by this CP-net is the following. The bottom element ($S_v \wedge W_w$) is the worst case and the top element ($S_f \wedge W_w$) is the best case.



There is an arrow between the nodes ($S_f \wedge W_w$) and ($S_v \wedge W_w$) because we can compare these states, every other thing being equal.
In this case, we can completely order the possible states (from the most preferred one to the least preferred one) :

$$(S_f \wedge W_w) \succ (S_f \wedge W_r) \succ (S_v \wedge W_r) \succ (S_v \wedge W_w)$$

This relation $\succ$ is the only ranking that satisfies this CP-net. ∎

4.2.3 CP-Boolean games

In Bonzon et al. (2009b), Boolean games are generalized with non dichotomous preferences: they are coupled with propositionalized CP-nets.

Definition 4.5 (CP-Boolean game)
A **CP-Boolean game** is a 4-uple $G = (N, V, \pi, \Phi)$, where $N = \{1, \ldots, n\}$ is a set of players, $V = \{x_1, \ldots, x_p\}$ is a set of propositional variables, $\pi : N \mapsto V$ is a control assignment function which defines a partition of V, and $\Phi = \langle \mathcal{N}_1, \ldots, \mathcal{N}_n \rangle$. Each $\mathcal{N}_i$ is a CP-net on V whose graph is denoted by $\mathcal{G}_i$, and $\forall i \in N$, $\succeq_i = \succeq_{\mathcal{N}_i}$.

Each CP-net $\mathcal{N}_i$ is a compact representation of the preference relation of player i on S.

Definition 4.6 (Strategy, strategy profile)

Let $G = (N, V, \pi, \Phi)$ be a CP-Boolean game. A **strategy** s_i for a player i is a π_i-interpretation. A **strategy profile** s is a n-tuple $s = (s_1, \ldots, s_n)$ where for all i, $s_i \in 2^{\pi_i}$.

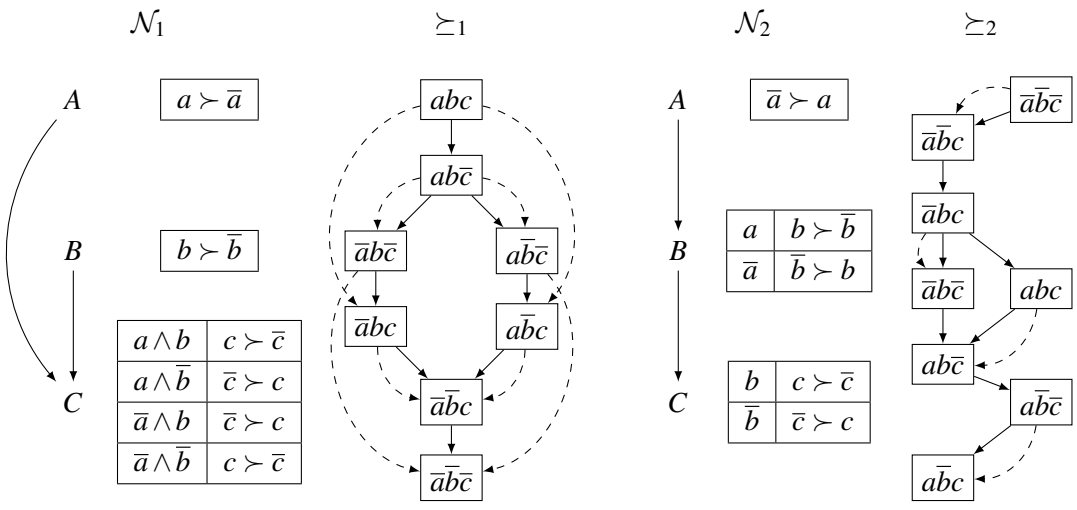
In other words, a strategy for i is a truth assignment for all the variables i controls. As $\{\pi_1, \ldots, \pi_n\}$ forms a partition of V, a strategy profile defines an (unambiguous) interpretation for V. Slightly abusing notation and words, we write $s \in 2^V$, to refer to the value assigned by s to some variable. We make use of the following notations which are standard in game theory. Let $G = (N, V, \pi, \Phi)$ be a Boolean game with $N = \{1, \ldots, n\}$, and $s = (s_1, \ldots, s_n)$, $s' = (s'_1, \ldots, s'_n)$ be two strategy profiles.

- s_{-i} denotes the projection of s onto $N \setminus \{i\}$: $s_{-i} = (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$
- π_{-i} denotes the set of the variables controlled by all players except i: $\pi_{-i} = V \setminus \pi_i$
- (s'_i, s_{-i}) denotes the strategy profile obtained from s by replacing s_i with s'_i without changing the other strategies: $(s'_i, s_{-i}) = (s_1, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n)$.

Pure strategy Nash equilibria (PNE) are classically defined for games where preferences are complete, which is not necessarily the case here. So we introduce the notion of *strong* PNE.

Definition 4.7 (Strong PNE)
Let $G = (N, V, \pi, \Phi)$ and $Pref_G = \langle \succeq_1, \ldots, \succeq_n \rangle$ the collection of preference relations on 2^V induced from Φ. Let $s = (s_1, \ldots, s_n) \in 2^V$. s is a **strong PNE (SPNE)** for G iff $\forall i \in \{1, \ldots, n\}$, $\forall s'_i \in 2^{\pi_i}, (s'_i, s_{-i}) \preceq_i (s_i, s_{-i})$.

Example 4.4 Consider the CP-Boolean game $G = (N, V, \pi, \Phi)$ where $N = \{1, 2\}$, $V = \{a, b, c\}$, $\pi_1 = \{a, b\}$, $\pi_2 = \{c\}$, $\mathcal{N}_1$ and $\mathcal{N}_2$, as well as $\succeq_1 = \succeq_{\mathcal{N}_1}$ and $\succeq_2 = \succeq_{\mathcal{N}_2}$, are represented on the following figure.
Arrows are oriented from more preferred to less preferred strategy profiles; we do not draw edges that are obtained from others by transitivity; and the dashed arrows indicate the links taken into account to compute Nash equilibria.



Using these partial pre-orders, the SPNE of G is $\{abc\}$. ■

The following proposition has been shown in Bonzon et al. (2009b).

Proposition 4.1 Let $G = (N, V, \pi, \Phi)$ be a CP-Boolean game such that graphs $\mathcal{G}_i$ are all identical ($\forall i, j \in N$, $\mathcal{G}_i = \mathcal{G}_j$) and acyclic. Then G has one and only one strong PNE.

The proof of this result makes use of the *forward sweep* procedure (Boutilier et al., 2004a) for

outcome optimization (this procedure consists of instantiating variables following an order compatible with the graph, choosing for each variable its preferred value given the value of its parents). Moreover, this SPNE can be built in polynomial time.

4.3 Argumentation and Boolean games

The leading idea here consists in translating an argumentation system AF into a Boolean CP-game G, to use specific tools of game theory, and some particular properties of Boolean CP-games for computing the extensions of AF. This is a joint work with Caroline Devred and Marie-Christine Lagasquie-Schiex.[8]
This idea is born from the following facts:

- argumentation and games have many strong links, in particular, the fact that both can represent interactions between rational agents
- a static game should allow the representation of an abstract argumentation framework: in both cases, agents give their arguments (resp. strategies) without analyzing those of the other agents, this analysis will be made afterward with the computation of extensions (resp. PNE)
- the graphical aspect of the CP-nets is similar to the graphical aspect of the argumentation (interaction graph)

This work aims to establish a new link between argumentation and games, but not to obtain more efficient algorithms for computing the extensions (there already exist many efficient algorithms defined in literature (see Cayrol et al., 2003; Doutre and Mengin, 2001)).

4.3.1 Add-ons on argumentation theory

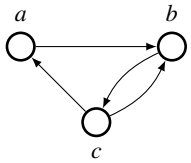
We already defined the basics of argumentation in Section 2.1 on page 9, but we will need some other notions in the following.
An "elementary" cycle, as defined by Amgoud et al. (2008), is a cycle that does not contain another cycle. However, as the classical definition of an "elementary" cycle in graph theory[9] is not the same as the one used in (Amgoud et al., 2008), we will use the word "minimal" instead of "elementary" to avoid any ambiguity.

Definition 4.8 (Cycle, Minimal cycle)
Let $X = \{a_0, \ldots, a_n\}$ and $Y = \{b_0, \ldots, b_m\}$ be two non-empty sequences of arguments of $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$.

- X is a **cycle** if and only if $\forall i \leq n-1$, $a_i \mathcal{R} a_{i+1}$ and $a_n \mathcal{R} a_0$. $n+1$ is the length of X.
- Considering that X and Y are cycles, X **strictly contains** Y, denoted by $Y \subset X$, if and only if $m < n$ and $\exists i = 0 \ldots n$ such that $a_i = b_0$, $a_{(i+1) \text{ modulo } (n+1)} = b_1, \ldots, a_{(i+m) \text{ modulo } (n+1)} = b_m$.
- X is a **minimal cycle** if and only if X is a cycle and $\nexists$ a cycle X' such that $X' \subset X$.

Example 4.5 In the following graph, the cycle $\{a, b, c\}$ is not minimal because it contains the minimal cycle $\{b, c\}$.

[8] Detailed proofs and algorithms can be found in (Bonzon et al., 2009a, 2010).
[9] In graph theory, an elementary cycle is a cycle in which a vertex cannot appear twice (see Berge, 1973).



■

We will also need the following properties (Amgoud et al., 2008; Doutre, 2002; Dung, 1995; Dunne and Bench-Capon, 2001, 2002):

Proposition 4.2
Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework such that $\mathcal{A} \neq \varnothing$.

1. Each unattacked argument belongs to every preferred extension of AF (see Dung, 1995).
2. An acyclic argumentation framework AF contains only one preferred extension (see Doutre, 2002; Dunne and Bench-Capon, 2001, 2002).
3. If $\varnothing$ is the unique preferred extension then AF contains at least an odd-length cycle (see Doutre, 2002; Dunne and Bench-Capon, 2001, 2002).
4. If AF does not contain a minimal odd-length cycle then its preferred extensions are not empty (see Amgoud et al., 2008).
5. If AF does not contain a minimal odd-length cycle then each preferred extension is also stable[10] (see Amgoud et al., 2008).

Many works in argumentation consider that odd-length cycles may be considered paradoxes, as they are a generalization of an argument attacking himself, in particular, if these cycles are minimal. Even if some odd-length cycles may make sense, we consider in this work that *argumentation frameworks should not contain minimal odd-length cycles*[11]. However, argumentation frameworks with even-length cycles will be taken into account.
As an argumentation framework without any minimal odd-length cycle is coherent, an interesting consequence occurs:

Corollary 4.1 Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework without minimal odd-length cycles. Let $E \in \mathcal{E}_{pr}(\mathsf{AF})$ be a preferred extension of AF. Let a be an argument of AF. If there is no attacker of a in E then $a \in E$.

4.3.2 Translation of an Argumentation system into a CP-Boolean game

The translation of an argumentation system into a CP-Boolean game can be done on argumentation systems that do not contain any minimal odd-length cycle. We thus need to make a "precompilation" of the argumentation system we want to translate, to remove the minimal odd-length cycles. This removal can be done in polynomial time if we choose to remove **all** the odd-length cycle, even if they are not minimal, or in exponential time if we remove **only** the minimal cycles:[12]

- To remove **all** odd-length cycles, the two following algorithms can be used:
  - IsCyclic which returns `true` if there exists at least one cycle in the argumentation graph. This algorithm is linear:

    (Step 1) removing all the vertices which do not have predecessors;

    (Step 2) iterating Step 1 until either all the remained vertices have at least one predecessor (there is a cycle in the initial graph), or the graph is empty (there is no cycle

[10] One says that the argumentation framework is coherent (see Dung (1995)).
[11] And, if it is not the case, these minimal odd-length cycles will be removed from the argumentation framework.
[12] This algorithm is detailed in (Bonzon et al., 2009a, 2010).

in the initial graph).

– REMODDCYCLES for removing the odd-length cycles if there are some of them in the AF. This algorithm is polynomial:

(Step 1) computation of the Boolean adjacency matrix corresponding to all shortest odd-length paths of attack (in terms of length); it is sufficient to take the Boolean adjacency of the graph $\mathcal{M}$ ($\mathcal{M}(i,j) = 1$ if there is an edge from i to j in AF) and to compute $\mathcal{M}^{\text{olc}} = \mathcal{M}^1 + \mathcal{M}^3 + \ldots + \mathcal{M}^{2n-1}$ with $n = |N|$;[13]

(Step 2) removal of all the arguments for which the diagonal element of the adjacency matrix $\mathcal{M}^{\text{olc}}$ is 1;

(Step 3) removal of all the edges having one removed argument as the endpoint or as the start point.

One can say that as Algorithm ISCYCLIC does not directly detect odd-length cycles, it is useless in the precompilation. However, as ISCYCLIC is a linear-time algorithm whereas REMODDCYCLES is only a polynomial-time one, it is interesting to avoid an unnecessary execution of REMODDCYCLES when AF is acyclic.

- The algorithm allowing to remove **only** minimal odd-length cycles is less efficient than the previous ones (it is an exponential-time algorithm), but it allows the conservation of some odd-length cycles which makes sense.

Let AF be an argumentation framework that does not contain minimal odd-length cycles. The principles of the algorithm TRANSLAFTOCPBG allowing to translate such an AF into a CP-Boolean game G are the following:

- each argument of AF is a variable of G;
- each variable is controlled by a different player (so we have as many players as variables);
- the CP-nets of all players are defined in the same way:
  - the graph of the CP-net is exactly the directed graph of AF;
  - the preferences over each variable v which is not attacked are $v \succ \overline{v}$ (if an argument is not attacked, we want to protect it; so the value `true` of the variable v is preferred to its value `false`),
  - the preferences over each variable v which is attacked by the set of variables $\mathcal{R}^{-1}(v)$ depends on these variables: if at least one variable $w \in \mathcal{R}^{-1}(v)$ is satisfied, v cannot be satisfied (so we have $\bigvee_{w \in \mathcal{R}^{-1}(v)} w : \overline{v} \succ v$[14]); otherwise, if all variables $w \in \mathcal{R}^{-1}(v)$ are not satisfied, v can be satisfied (and so $\bigwedge_{w \in \mathcal{R}^{-1}(v)} \overline{w} : v \succ \overline{v}$).

The construction of a CP-Boolean game G from an argumentation framework AF is made in polynomial time (even if AF is cyclic and if we have to remove all its odd-length cycles).
The use of this algorithm implies the following property:

Proposition 4.3 Let $\text{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework without minimal odd-length cycles. Let $G = (N, V, \pi, \Phi)$ be the CP-Boolean game obtained by translating AF using TRANSLAFTOCPBG. s is a preferred extension of AF if and only if s is a SPNE for[a] G.

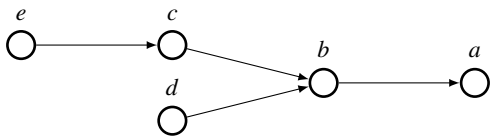
[a] Recall that s denotes a V-interpretation, that is if $s = a\overline{b}c$ for example, this corresponds to the set $\{a, c\}$.

The translation of an AF into a CP-Boolean game is illustrated in the following example.

[13] The bound $2n-1$ is obtained using a general result given by graph theory: if a directed graph contains a path from a to b then there exists an elementary path – in the classical sense given by graph theory: a path in which each vertex appears only once – from a to b.

[14] The formula $w : \overline{v} \succ v$ (resp. $\overline{w} : \overline{v} \succ v$) means that, for the value `true` (resp. `false`) of the variable w, the value `false` of the variable v is preferred to its value `true`.

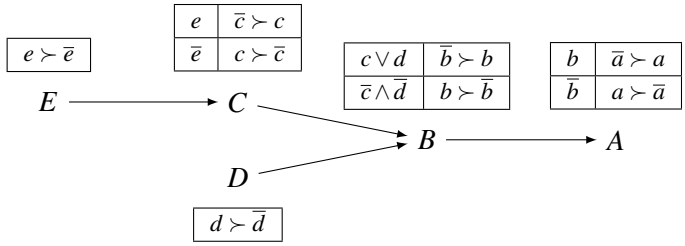
Example 4.6 Consider the following acyclic argumentation systems, $\mathsf{AF} = \langle\{a,b,c,d,e\}, \{(b,a), (c,b), (d,b), (e,c)\}\rangle$.



By applying TRANSLAFTOCPBG, AF is transformed in a CP-Boolean game $G = (N,V,\pi,\Phi)$, such that:

- $V = \{a,b,c,d,e\}$
- $N = \{1,2,3,4,5\}$
- $\pi_1 = \{a\}$, $\pi_2 = \{b\}$, $\pi_3 = \{c\}$, $\pi_4 = \{d\}$ and $\pi_5 = \{e\}$

The following CP-net represents the preferences of all players. To distinguish the CP-net from the AF, nodes in the CP-net are in uppercase, whereas nodes in the AF are in lowercase.



G has one SPNE $\{ed\overline{c}\overline{b}a\}$ and AF has only one preferred extension $\{e,d,a\}$. ∎

The following example shows the translation of a cyclic AF but with only even-length-cycles:

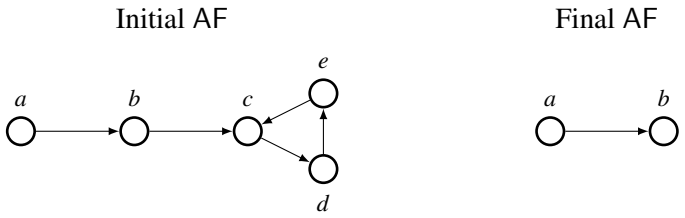
Example 4.7 Consider $\mathsf{AF} = \langle\{a,b\},\{(a,b), (b,a)\}\rangle$.

As AF is cyclic, but contains only even-length cycles, TRANSLAFTOCPBG can be applied. We obtain $G = (N,V,\pi,\Phi)$, such that $V = \{a,b\}$, $N = \{1,2\}$, $\pi_1 = \{a\}$, $\pi_2 = \{b\}$. The following CP-net represents the preferences of all players:

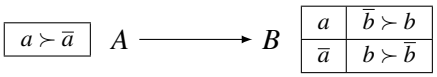
G has two SPNEs $\{a\overline{b}, \overline{a}b\}$ and AF has two preferred extensions $\{a\}$, $\{b\}$. ∎

And then we give an example of an AF with odd-length cycles:

Example 4.8 Consider $\mathsf{AF} = \langle\{a,b,c,d,e\}, \{(a,b), (b,c), (c,d),(d,e), (e,c)\}\rangle$. The initial AF is cyclic and contains a minimal odd-length cycle. This cycle has to be removed before we can apply TRANSLAFTOCPBG, and the final AF contains only a and b.



So, by applying TRANSLAFTOCPBG, We obtain $G = (N, V, \pi, \Phi)$, such that $V = \{a, b\}$, $N = \{1, 2\}$, with $\pi_1 = \{a\}$, $\pi_2 = \{b\}$ and the following CP-net which represents the preferences of all players:



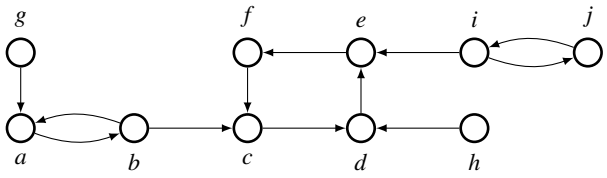
G has one SPNE $\{a\bar{b}\}$ and the final AF (after removal of minimal odd-length cycles) has one preferred extension $\{a\}$. ∎

4.3.3 Computation of preferred extensions

Since preferred extensions correspond exactly to SPNEs, the main properties of the computation of SPNE in CP-Boolean games can be applied. The first interesting case concerns the acyclic argumentation frameworks:

Proposition 4.4 Let AF be an argumentation framework without minimal odd-length cycles. Let G be the CP-Boolean game obtained from AF by applying TRANSLAFTOCPBG. If AF is acyclic, AF has one and only one preferred extension which is computable in polynomial time using G.

This proposition holds for the simple case of acyclic argumentation frameworks. The computation of SPNE(s) for cyclic argumentation frameworks is more complex.
First, we need the algorithm COMPINTCYCLEFORPROP, which returns the cycle (or one of the cycles if there are several) in a given set of variables which allows to reach more variables as possible.
For instance, on the following graph:



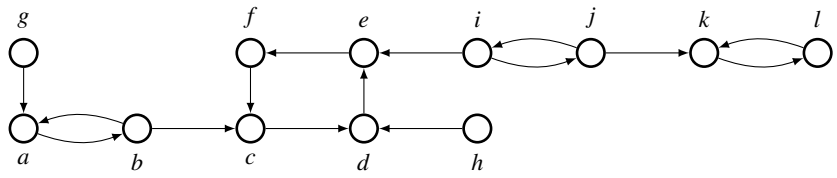
$\{a, b\}$ allows to reach the variables a, b, c, d, e, f, and $\{i, j\}$ allows to reach the variables i, j, c, d, e, f. This type of cycle allows a more interesting propagation of values over the graph (if they are the starting point of a propagation process then this propagation is more efficient).

Let $\mathcal{N}$ be the CP-net representing goals of players of a CP-Boolean game, the principles of Algorithm COMPSPNEREC, allowing to recursively compute the SPNEs of a CP-Boolean game obtained from an argumentation framework are the following:

- instantiation of all unattacked variables (which have no parents in $\mathcal{N}$ and are satisfied in the SPNE)
- propagation of these instantiations as long as possible
- once all feasible instantiations have been done, loop:

- if all variables have been instantiated, the SPNE can be returned
- else, with Algorithm CompIntCycleForProp, the more interesting[15] cycle C remaining is computed (there is one, otherwise, all variables would have been instantiated)
- using the current state of the current SPNE, create two new SPNEs; the first one contains a variable of C instantiated to `true`, and the second one contains this same variable instantiated to `false`
- propagation of these instantiations for each one of these SPNEs as long as possible.

Example 4.9 Using the following graph:



the steps of the computation process are:

- g and h are instantiated to `true` (current state of SPNE $= gh$)
- a and d are instantiated to `false` (current state of SPNE $= gh\bar{a}\bar{d}$)
- b is instantiated to `true` (current state of SPNE $= gh\bar{a}\bar{d}b$)
- c is instantiated to `false` (current state of SPNE $= gh\bar{a}\bar{d}b\bar{c}$)
- at this point the simple propagation stops. We must compute the interesting cycles in the remaining set of variables $\{e,f,i,j,k,l\}$ and the result is $\{i,j\}$.
- the propagation process is restarted with the following current states of two SPNEs: $gh\bar{a}\bar{d}b\bar{c}i$ and $gh\bar{a}\bar{d}b\bar{c}\bar{i}$
- ...
- at the end of the propagation process, three instantiations are obtained $gh\bar{a}\bar{d}b\bar{c}i\bar{e}f\bar{j}k\bar{l}$, $gh\bar{a}\bar{d}b\bar{c}\bar{i}ef\bar{j}k\bar{l}$ and $gh\bar{a}\bar{d}b\bar{c}i\bar{e}f\bar{j}\bar{k}l$. These SPNEs correspond to the three preferred extensions $\{g,h,b,e,j,l\}$, $\{g,h,b,f,i,k\}$ and $\{g,h,b,i,f,l\}$.

■

The following proposition shows that Algorithm CompSPNERec allows to exactly compute the set of SPNEs of the CP-Boolean game.

Proposition 4.5 Let G be a CP-Boolean game given by Algorithm TranslAFtoCPBG. Let SP be the set of strategy profiles of G given by Algorithm CompSPNERec. $s \in SP$ if and only if s is a SPNE for G.

Another interesting question regarding the links between game and argumentation theory is to study how the concepts of cooperative game theory can guide agents as to what argument should be put forward in a debate.

4.4 Coalitional games for abstract argumentation

Strategical interactions between agents are central when we consider multi-agent debates. A central problem faced by agents contributing in such a multiagent debate is that they have to put forward arguments taking into account their own goals, but also how the audience (the other agents taking part in the debate) may receive their arguments, and also possibly whether the rules of the debate

[15] That is, the cycle that allows the more efficient propagation.

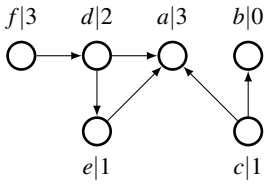
will allow them to put forward these arguments. Consider, for instance, the attitude of politicians participating in public debates: their choice to embrace arguments often depends on factors like, for instance, the popularity of the arguments, the share of voters supporting those arguments, a degree of personal satisfaction, the consensus generated by those arguments in an assembly or a forum, the contiguity with a political position, etc.
This results in a complex decision-making problem, where most of the parameters are likely to be uncertain: what are the arguments known by other agents? What are their own goals? Perhaps the most basic type of uncertainty is that it is virtually impossible to exactly predict what combination of arguments will result from the debate. In this context, it is not clear how the debate will evolve, and thus the decision as to what argument to put forward is a difficult one.
In this joint work with Nicolas Maudet and Stefano Moretti (Bonzon et al., 2014), we investigated how to guide agents as to what move is good to play and proposed to exploit cooperative game theory to account for the decision problem faced by an agent in this context.
We assume that each agent has (cardinal) preferences over **single** arguments $v : \mathcal{A} \to \mathbb{N}$, and that they take part in a debate where the outcome is difficult to predict. It may be that they don't know how the debate will evolve, what are the arguments known by the other agents, or how people will vote. The outcome is evaluated thanks to an **argumentative filter**, that is a function $M(S)$ which returns the *valuable* arguments. Then:

$$v(S) = \sum_{a \in M(S)} v(a)$$

for each $S \subseteq \mathcal{A}$ with $|S| \geq 2$.
The argumentative filter may be (almost) any function $M(S)$ which tells us which arguments are valuable in the end. For instance:

- **anything uttered**: any argument just put forward during the debate is valuable
- **final word**: any unattacked argument in the debate outcome is valuable
- **grounded extension**: any acceptable argument under the grounded extension is valuable

Example 4.10 Let the following argumentation system, where the number next to an argument's name represents the valuation (cardinal preference) of the agent.



If the argumentative filter is:

- "anything uttered", $M(S) = \mathcal{A}$, and $v(S) = 10$.
- "final word", $M(S) = \{c, f\}$, and $v(S) = 4$.
- "grounded extension", $M(S) = \{c, e, f\}$, and $v(S) = 5$.

■

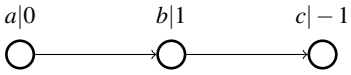
Definition 4.9 (Coalitional Argumentation Framework (CAF))
A **Coalitional Argumentation Framework** (CAF) is a triple $\langle \mathcal{A}, \mathcal{R}, v \rangle$ where

- $\langle \mathcal{A}, \mathcal{R} \rangle$ is an argumentation framework;
- $v : 2^{\mathcal{A}} \to \mathbb{R}$ is a map that associates to each coalition $S \subseteq \mathcal{A}$ its "worth" for the user

- $\forall a \in \mathcal{A}$, $v(a)$ is given by a cardinal preference relation over $\mathcal{A}$
- $\forall S \subseteq (A)$, with $|S| \geq 2$, $v(S)$ is computed thanks to the argumentative filter $M(S)$

Given a CAF $\langle \mathcal{A}, \mathcal{R}, v \rangle$, we study the problem of providing a measure representing the relevance of arguments. The first idea is to compute the marginal contribution of each argument, as well as their Shapley value (see Section 4.1.3 on page 103).

Example 4.11 Let $M(S)$ be "final word"



S :	$\{a\}$	$\{b\}$	$\{c\}$	$\{a,b\}$	$\{a,c\}$	$\{b,c\}$	$\{a,b,c\}$
$M(S)$:	$\{a\}$	$\{b\}$	$\{c\}$	$\{a\}$	$\{a,c\}$	$\{b\}$	$\{a\}$
$v(S)$:	0	1	-1	0	-1	1	0

We can now compute the marginal contributions of each argument:

S	$v(S)$	$v(S) \cup \{a\}$	$v(S) \cup \{b\}$	$v(S) \cup \{c\}$
$\emptyset$	0	0	1	-1
$\{a\}$	0	–	0	-1
$\{b\}$	1	0	–	1
$\{c\}$	-1	-1	1	–
$\{a,b\}$	0	–	–	0
$\{a,c\}$	-1	–	0	–
$\{b,c\}$	1	0	–	–

The Shapley value of each of these arguments is the following :

- $\pi_a^{\hat{\mathbf{p}}}(v) = -\frac{1}{2}$
- $\pi_b^{\hat{\mathbf{p}}}(v) = 1$
- $\pi_c^{\hat{\mathbf{p}}}(v) = -\frac{1}{2}$

■

The question is to figure out if the Shapley value is coherent in this context. We want to ensure that both the structure of the argumentation framework and the worth of opinions as measured by v are taken into account. To do so, we focus on properties such that a measure of relevance should satisfy.

Let $\langle \mathcal{A}, \mathcal{R}, v \rangle$ be a CAF. We recall that $\mathcal{R}_1^-(a)$ is the set of direct attackers of $a \in \mathcal{A}$, and denote $\mathcal{S}(a)$ the set of arguments attacked by a: $\mathcal{S}(a) = \{b \in \mathcal{A} | a \in \mathcal{R}_1^-(b)\}$.

Symmetry (SYM) The Symmetry property says that the relevance of an argument does not depend on its label.
Let $i, j \in \mathcal{A}$ be such that $\mathcal{R}_1^-(i) = \mathcal{R}_1^-(j)$, $\mathcal{S}(i) = \mathcal{S}(j)$ and $v(i) = v(j)$. A power index $\pi^{\mathbf{p}}$ satisfies the property of **Symmetry (SYM)** if and only if $\pi_i^{\mathbf{p}}(v) = \pi_j^{\mathbf{p}}(v)$.

Disconnected Argument (DA) The Disconnected Argument property says that disconnected arguments should receive as value of relevance precisely their utility.
Let $i \in \mathcal{A}$ be such that $\mathcal{R}_1^-(i) = \mathcal{S}(i) = \emptyset$. A power index $\pi^{\mathbf{p}}$ satisfies the property of **Disconnected Argument (DA)** if and only if $\pi_i^{\mathbf{p}}(v) = v(i)$.

Efficiency (EFF) The Efficiency property imposes an upper bound over the total amount of

relevance of arguments.
A power index $\pi^{\mathbf{p}}$ satisfies the property of **Efficiency (EFF)** if and only if $\sum_{i\in\mathcal{A}} \pi_i^{\mathbf{p}}(v) = \sum_{i\in M(\mathcal{A})} v(i)$.

Equal Impact of Attack (EIA) A consequence of an attack should equally affect both arguments involved in the attack.
Let $i, j \in \mathcal{A}$ with $i \neq j$. Consider a CAF $\langle \mathcal{A}, \mathcal{R} \cup \{(i,j)\}, v_{ij}\rangle$. A power index $\pi^{\mathbf{p}}$ satisfies the property of **Equal Impact of Attack (EIA)** if and only if $\pi_i^{\mathbf{p}}(v) - \pi_i^{\mathbf{p}}(v_{ij}) = \pi_j^{\mathbf{p}}(v) - \pi_j^{\mathbf{p}}(v_{ij})$

Additivity (ADD) Let $\text{CAF}_1 = \langle \mathcal{A}, \mathcal{R}_1, v_1\rangle$ and $\text{CAF}_2 = \langle \mathcal{A}, \mathcal{R}_2, v_2\rangle$ such that there exists $\text{CAF}_3 = \langle \mathcal{A}, \mathcal{R}_3, v_1 + v_2\rangle$. A power index $\pi^{\mathbf{p}}$ satisfies the property of **Additivity (ADD)** if and only if $\pi_i^{\mathbf{p}}(v_1) + \pi_i^{\mathbf{p}}(v_2) = \pi_i^{\mathbf{p}}(v_1 + v_2)$.
The sum of the relevance of each argument i in CAF_1 and CAF_2 must be equal to the relevance of i in CAF_3.

An axiomatic characterization can then be obtained:

Theorem 4.1 The Shapley value is the unique index that satisfies properties EFF, SYM, DA, EIA and ADD.

Moreover, if in general, the Shapley value is hard to calculate since it requires a number of operations that is exponential in the number of arguments, good computational properties can be derived for the specific class of CAFs introduced in this section.
Assigning to each argument the status of 'accepted' or 'rejected' can be seen as a simplistic manner to compare arguments in decision-making applications. The measure of importance for arguments provided by the Shapley value in a CAF can be interpreted as a novel measure of acceptability, provided that all the arguments are indifferent to the user.

5. Interaction in Task Allocation

With the rise of low-cost robotics and drones, multi-agent coordination (MAC) has proven very effective for robotic teamwork (e.g. Marcolino et al., 2013; Vig and Adams, 2006). Many MAC problems require multiple heterogeneous agents to concurrently perform a joint task, comprised of sub-tasks. For example, in search and rescue problems (Beck et al., 2016), robots with complementary capabilities perform a set of tasks that jointly address a global task. Yet, the vast majority of such solutions assume task independence. Moreover, as many communication disconnections or breakdowns may occur in real-world applications, we need multi-agent task allocation mechanisms in a decentralized and robust manner, to avoid the single-point failure possible in a centralized configuration.
Such MAC problems are commonly solved via agent coalition formation (Shehory and Kraus, 1998). Thus, the global task is accomplished by a set of coalitions comprising a coalition structure (Rahwan et al., 2015). Optimal coalition formation and coalition structure generation are exponentially complex. Recent progress led to complexity reduction in specific domains, however, optimal solutions remain exponential. Task interdependence further increases complexity as the formation of a coalition and its utility may depend on other coalitions. Distributed solutions that attempt to ease complexity (e.g Michalak et al., 2010) opt for an anytime approach, where quality improves as the formation process progresses.

In this chapter, we present a novel decentralized, anytime coalition formation and task allocation mechanism, that diverges from the state of the art in several ways. Specifically, we address coalition formation where sub-tasks and coalition utilities are interdependent, thus affecting the global utility of the coalition structure. To address this, our mechanism simultaneously considers local and global task requirements, accounting for interrelations thereof. Such interrelations are seldom considered in the prior art. Additionally, our approach explicitly represents both qualitative and quantitative information on agent characteristics and task requirements.
Our coalition formation and task allocation mechanism is fully decentralized, thus preventing a single point of failure. Initially, agents only know their own characteristics, the global task and its sub-tasks and their respective requirements, and the set of available agents. Gradually, agents may accumulate information on the characteristics of other agents, potential coalitions and coalition structures. Throughout the process, each agent matches task requirements against its characteristics (and characteristics of other agents it learned about) and accordingly decides which coalition it should join to maximize global utility.
Our mechanism comprises 2 stages. Stage I, denoted *Feasible Interdependent Coalition Structure Anytime Method (FICSAM)*, finds a feasible coalition structure if one exists. If a solution is found, in stage II, denoted *Improved Feasible Interdependent Coalition Structure Anytime Method (IFICSAM)*, agents incrementally improve it in a decentralized manner via replacements of single agents in the coalition structure, while maintaining feasibility (i.e. no single or global task requirements are violated). Thus, we guarantee at any time, the generation of a solution that systematically increases the global utility.

Note that the main part of this work has been done by Douae Ahmadoun (Ahmadoun, 2022) during her PhD thesis, in collaboration with Thales[1], and co-supervised with Pavlos Moraitis.

5.1 Interdependent task allocation

We consider the problem of decentralized allocation for a set of interdependent tasks $T = \{t_1, t_2, \ldots, t_{|T|}\}$ necessary to accomplish a global task T, to a set of heterogeneous cooperative agents $A = \{a_1, a_2, \ldots, a_{|A|}\}$. The global task T represents the global goal that the team of cooperative agents has to achieve.
A specific profile can describe each agent. For instance, in a given application, an agent can be described by its type, ownership or not of a resource, position on a grid, remaining fuel, or distances to the different tasks.

Definition 5.1 (Agent characteristics)
Consider a set of heterogeneous agents $A = \{a_1, a_2, \ldots, a_{|A|}\}$. An agent $a_i \in A$ is described by a set of attributes $X = \{x_1, \ldots, x_{|X|}\}$ where $x_k(a_i)$ denotes the value of a_i under attribute k. Without loss of generality, we consider each attribute as a function $x_k : A \rightarrow D_k$, D_k is the domain of values for attribute k. We call these attributes **agent characteristics**.

We notice here that our model characteristics capture the qualitative values, like the type and possession of a resource, as well as the quantitative values, like the position, quantity of the remaining fuel, and distances to tasks simultaneously.
The agents have to perform tasks as defined below:

Definition 5.2 (Single task requirements)
A task $t_j \in T$ is described by a set of attributes $\Gamma_{t_j}=\{\gamma_1, \ldots \gamma_m\}$ where $\gamma_l(t_j)$ is the value of task t_j under the attribute l. Without loss of generality, we consider each attribute as a function $\gamma_l : T \rightarrow K_l$, K_l is the domain of values for attribute l. We call the attributes Γ_{t_j} **single task requirements**.

Task requirements also depend on the application and can be either qualitative or quantitative. A requirement concerning a specific type of resource or agent is a qualitative requirement. By contrast, a minimum number of available agents or maximal distances are quantitative requirements.

Macro-level requirements may be necessary to express the presence of interdependencies and to check their manageability. As in single-task, task combination requirements are expressed by instantiated attributes and focus each on a specific subset of agent characteristics. For a specific combination of tasks, a joint demand is expected to be covered by the subsets of agents assigned to the tasks of the task combination. For example, for two specific tasks that each requires at least one resource of a certain type, the combination formed of those tasks may require the existence of a supplementary resource of the same type among one of the tasks. It doesn't matter which task has the supplementary resource, but it is enough if one of them has it. This constraint concern the two tasks and cannot be verified in all the situations for one of the two tasks without reckoning the other one.

Definition 5.3 (Task combination requirements)
Consider a set of attributes $\Lambda_{cbt_j}=\{\lambda_1, .., \lambda_n\}$ concerning task combinations such that $\lambda_l(cbt_j)$ is the value of task combination $cbt_j \in T_{cbt}$, $T_{cbt} \subseteq 2^T$, under attribute l. Without loss of generality, we consider each attribute as a function $\lambda_l : T_{cbt} \rightarrow F_l$, F_l is the domain of values for attribute l. We call the attributes Λ_{cbt_j} **task combinations requirements**.

[1] https://www.thalesgroup.com

Examples of single-task requirements can be the number of agents that allow accomplishing a task or other application-related constraints that have to be satisfied for the tasks to be accomplished. Task combination requirements can be seen as constraints that have to be satisfied to address the interdependence among tasks (for example, temporal constraints imposing accomplishment order). These task combination requirements are a general representation of the inter-task dependencies. In real-world applications, several tasks can have joint requirements. For example, in a given situation, two tasks (or more) may need to have a minimal number of resources as a single task requirement and need mutually an additive resource, that the tasks' coalitions might lend if needed. A dependency, or a task combinations requirement, can even concern all the tasks at once.

Task requirements and task combination requirements are easily translated into **constraints**. Single-task requirements can be seen as constraints defined locally on tasks. These must be satisfied for the tasks to be performed. Task combination requirements can be seen as constraints over the global task T that must be satisfied to address dependencies among tasks.

Combined characteristics of a set of agents may allow fulfilling task requirements, or generate conflicts with such requirements.

Definition 5.4 (Fulfillment relation)
Let $s = \{a_1, ..., a_{|s|}\}$ be a set of agents, $X_s = \{X_{a_1}, ..., X_{a_{|s|}}\}$ their characteristics, $t_j \in T$ a task and $\bowtie$ denoting the satisfaction (with respect to a mathematical operator) of the assignment of a value to a requirement $\gamma_l(t_j)$ by the assignment of a value to some characteristic $x_k(a_i)$.
We say that s can **fulfil** a requirement $\gamma_l \in \Gamma_{t_j}$ denoted $s_{cc} \circlearrowleft \gamma_l(t_j)$ if there exists a combination of characteristics $cc = \{x_k, ..., x_r\} \subseteq X_{a_1} \cup, ..., \cup X_{a_{|s|}}$ such that $\sum_{i=1}^{|s|} x_k(a_i) \bowtie \gamma_l(t_j)$ for some $x_k \in cc$ or $x_r(a_i) \bowtie \gamma_l(t_j)$ for some $x_r \in cc$ with $r \neq k$, saying (slightly abusing the notation) that $cc \bowtie \gamma_l$.
In contrast, we say that s_{cc} has a **conflict** with the requirement $\gamma_l \in \Gamma_{t_j}$, if $\exists x_k \in cc$ such that x_k is in conflict with this requirement γ_l denoted as $x_k \not\bowtie \gamma_l$.

When a subset of agents conflicts with a task requirement, it means that the subset cannot "contribute" to the task according to this specific requirement. The violation of the requirement cannot be amended adding any agent or agents of any possible combination of characteristics.

Agents can form *coalitions* to accomplish a task $t_j \in T$.

Definition 5.5 (Coalitions)
Let a set of agents $A = \{a_1, a_2, ..., a_{|A|}\}$, a set of tasks $T = \{t_1, t_2, ..., t_{|T|}\}$ composing the global task T and $\tau : A \to T$ a function assigning an agent a_i to a task t_j when $\exists x_k \in X_{a_i}$, $\exists \gamma_l \in \Gamma_{t_j}$ such that $x_k \bowtie \gamma_l$, and $\nexists x_m \in X_{a_i}$ such that $x_m \not\bowtie \gamma_p$ for any $\gamma_p \in \Gamma_{t_j}$ with $p \neq l$. A **coalition** $C_{t_j} \in 2^A$ whose task is t_j is a subset of agents $C_{t_j} \in 2^A$ such that $C_{t_j} = \{a_i \in A \mid \tau(a_i) = t_j\}$.

To perform a global task T we need a set of coalitions S, called *coalition structure*. Each coalition $C_{t_j} \in S$ is assigned a task $t_j \in T$. When S can accomplish T we call it a *feasible coalition structure*. Formally:

Definition 5.6 (Feasible coalition structure)
Let a global task $T = \{t_1, t_2, ..., t_{|T|}\}$ and a coalition structure $S = \{C_{t_1}, C_{t_2}, ..., C_{t_{|T|}}\}$ over the set of tasks T. S is a **feasible coalition structure** (called also a feasible solution) denoted S^f if and only if $\forall t_j \in T$ such that $C_{t_j} \in S^f$ it holds that $\forall \gamma_l \in \Gamma_{t_j}$, there exists a set $s_{cc} \subseteq C_{t_j}$ such that $s_{cc} \circlearrowleft \gamma_l(t_j)$ and if Λ_{cbt} is a set of requirements concerning the combination of tasks then $X_{S^f} \bowtie \Lambda_{cbt}$.

Note that agents are *single-task*, i.e., they can accomplish only one task at a time (see Gerkey and Matarić, 2004). Thus, an agent should exist in no more than one coalition of coalition structure.

From the perspective of coalitions, this is equivalent to stating that two different coalitions cannot contain the same agent, formally: $C_{t_j}, C_{t_k} \in S$, $j \neq k$ then $C_{t_j} \bigcap C_{t_k} = \varnothing$. However, some agents can have no task assigned to them, and thus not exist in any coalition of the structure, formally: $\bigcup_{j=1}^{|T|} C_{t_j} \subseteq A$.

A utility function is defined following the application's needs. It should consider the globality of the tasks to cover the interdependencies between tasks and not be simply a sum of utilities over tasks.

Definition 5.7 (Coalition Structure Evaluation)
Consider $CS = \{S_1, \ldots, S_n\}$ be the set of all possible coalition structures that could be assigned to a global task T. We introduce a set of criteria G such that for each $g_k \in G$ there exists a weak order G_k upon the set CS, $G_k \subseteq CS^2$ such that if $(S, S') \subseteq G_k$, then $S \succeq S'$ and $\exists g_k : CS \to \mathbb{R}$, $g_k(S) \geq g_k(S')$.

Note that we consider interdependent tasks. Hence, it is necessary to define a function that evaluates a coalition structure S as a whole considering the interdependence among the coalitions in the structure. Evaluating the whole structure based only on individual evaluations of the coalitions would not be sufficient due to interdependencies between tasks and thus between coalitions.
Provided the conditions of commensurability, compensation and preferential independence are satisfied among the criteria in G (see Bouyssou et al., 2000), a global additive value function u_{global} is applicable. However, our approach is generic and therefore other types of evaluation functions could be considered.

5.2 Decentralized coalition formation

As stated above, the main aim of our work is to find the best feasible coalition structure with respect to the global utility u_{global} if it exists or detects the non-existence of a feasible coalition structure as early as possible. We propose a decentralized solution approach based on token-passing among the agents, and candidate members of different coalitions assigned to the tasks composing the global task.
The process is divided into rounds. In each round, the token is circulated among the agents. The round ends when all agents have received the token once. Agent ordering depends on application-based criteria. For example, in an application with a specific hierarchy, the token may be sent from the most important, and thus possibly best candidate in the team, to the less important, the one with fewer resources and qualities for the application. In a spatial-placed application, the token can be passed from an agent to its nearest agent.

In the beginning, each agent knows its own characteristics and the global task T to be accomplished. Information about the other agents and the evolution of the coalition structure formation arrives via the token-passing process. When agent a_i sends the token to agent a_j, the next agent adds information on the best (with respect to u_{global}) feasible coalition structure S formed so far, the accumulated expertise (*i.e.* characteristics) X_S of the participating agents in S, and the number of agents nd who have not changed the coalition structure (have not decided to join a coalition).
Holding the token, agent a_i may decide to join (or initiate) a coalition C_{t_j} assigned to task t_j, if it can contribute to the accomplishment of task t_j or if its participation can increase the global utility function u_{global}. Where applicable, agent a_i updates the information it received with the changes it had applied. Then, agent a_i passes the token to the next agent. If it has not joined a coalition, it passes the token to the next agent by forwarding the information from the previous agent.

We assume that agents can exchange messages and we describe the communication protocol that organizes the messages exchanged between agents. Yet, the underlying communication infrastructure (*i.e.* studying questions such as whether there is a message board, the agents

acknowledge message receipts, communication is asynchronous, or there is any noise) is beyond this work's scope.

Concretely, our mechanism has two stages:

Stage I coincides with the first round. In this stage, agents try to find a feasible coalition structure S^f, if one exists. An agent joins a coalition only if it can satisfy some task requirements not yet satisfied by other coalition members, regardless of the impact on u_{global} (*i.e.*, the improvement of the u_{global} value is not a precondition). If no S^f is found in stage I, no such S^f exists considering the requirements of the tasks and the characteristics of the available agents.
We call the method implementing this stage **FICSAM**.

Stage II implements our second method incrementally improving the feasible coalition structure found so far. This method starts from round 2. Once a feasible coalition structure was found at the end of round 1, agents improve u_{global} via replacements or swaps between single or groups of agents. The resulting improved structure must preserve feasibility by respecting all the single-task and task combination requirements.
We call the method implementing the first stage combined with this second stage **IFICSAM**.

5.2.1 General Process

Here, we present the general process for the decentralization of the coalition formation generation.[2]
An agent participating in the process acts either as the process initiator or as a candidate member of some coalitions of a certain coalition structure S.
An initiator agent a_i starts by initializing the set of tasks it can perform T_{a_i}. It then picks one of them, say t_j, and initializes a coalition in the structure S that is assigned to t_j. We mention here that S is at the beginning a coalition structure with empty coalitions. Following this, the agent adds its characteristics to X_S (initially empty) that accumulate the characteristics of all the members of the coalitions in structure S. Next, it checks the feasibility of the structure S.
In the unlikely case where structure S is feasible (e.g. if T contains only the task t_j and t_j is fulfilled by this exact initiator agent), this coalition structure becomes an anytime coalition structure. Agent a_i considers S as a feasible coalition structure to explore and sends this proposal to all the agents in A. Then, in both cases, whether coalition structure S is feasible or not, agent a_i initializes the decentralized process of coalition structure formation by sending a message to the next agent a_j. With this message, agent a_i passes the token to agent a_j, who enters the process. a_i informs a_j about the structure S, the characteristics gathered so far X_S, the current round R, and the number of agents that did not change their decision in the last round (here zero). Figure 5.1 on the next page presents globally the main steps the first agent goes through in the process.

When an agent a_i acts as a candidate member of a coalition structure, its activity depends on the messages it receives from other agents. This is illustrated in Figure5.2 on page 123. Three types of messages exist.
In the case of a *propose 'ats'* message, if an anytime solution is required, the coalition formation process terminates with S as the best (with respect to u_{global}) feasible coalition structure found so far. This can occur either at the end of the first round or in the middle of any other round (*i.e.* all the available agents cannot check whether they can contribute to improving this feasible coalition structure).
Otherwise, the receiving agent considers that S is a feasible coalition structure that might be further improved. For that, the receiving agent uses procedure *decide*. The message *propose 'gt'* means that S is not a feasible coalition structure and the sender passes the token (*gt* meaning give token) to the receiving agent who uses procedure *decide* to look whether it can contribute to the search for

[2] An interested reader can find the detailed algorithms in Ahmadoun (2022); Ahmadoun et al. (2021).

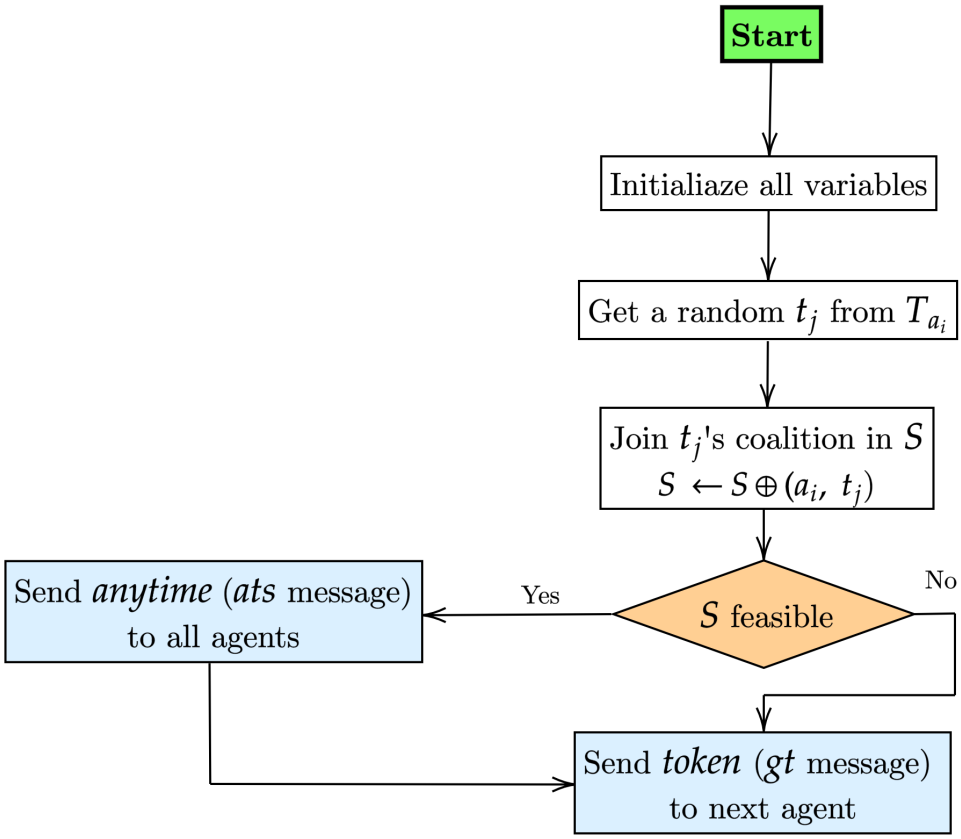


Figure 5.1: First agent process

a feasible coalition structure. The message *inform* with $S_{msg} = \emptyset$ informs the agents that no feasible coalition structure has been found at the end of the first round and therefore the process ends with failure, while the same message *inform* with $S_{msg} \neq \emptyset$ informs agents that the process ends with a feasible and improved (with respect to the feasible coalition structure found in the first round) solution, meaning that none of the available agents can furthermore improve the current feasible coalition structure.

Decision Process

First, agent a_i checks the feasibility of the received coalition structure S. This is done using a feasibility checking procedure according to the feasibility criteria of the given application, based on the single task and task combinations requirements of the application.
Then, if the process is in the first round, and the received coalition structure S is not yet feasible, the agent computes a feasible structure, accounting for the set of tasks T that have to be accomplished, task-level and combination-level requirements, and its potential contribution in case it joins S. This is done by implementing a CSP (Constraint Satisfaction Problem) to generate a feasible coalition structure if it exists.
When agent a_i holds the token, it adds the information about its characteristics to X_S, the set of agent characteristics gathered by preceding agents. Thus, when agent a_i gets the token in the first round, it has to solve a locally centralized coalition formation problem based on the knowledge accumulated so far. This knowledge concerns more precisely the agents that have already joined the coalition structure and their characteristics. Based on this knowledge, agent a_i tries to find whether preceding agents, regarding their characteristics in X_S, along with a_i are sufficient for satisfying all

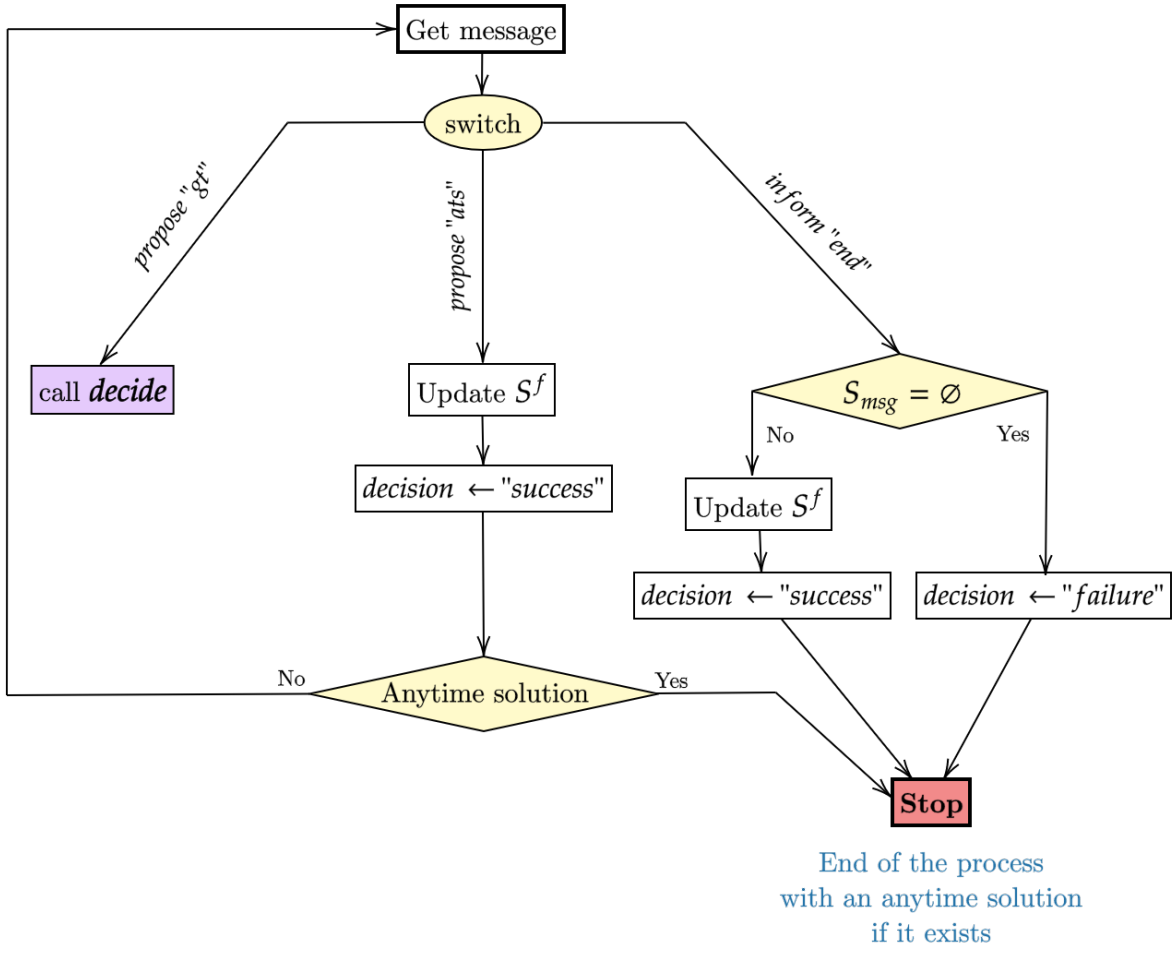


Figure 5.2: Global process

of the requirements of both individual tasks and task combinations.

Requirements are modeled as hard constraints (see, for example, Dechter (2003)), and a CSP problem is solved. Note that in the case where no feasible coalition structure is found in the first round of the decision process (*i.e.*, the CSP problem has no solution with the accumulated knowledge on agents until now), the process is to be terminated as there is no solution at all. In this case, the absence of a solution is proved the following way: as each agent transfers its characteristics in X_S with the token-passing, at the end of round 1 (marking a complete passage of the token to all the agents) the last agent has complete knowledge of all the agents' characteristics. If a solution existed, this last agent would have found it using this knowledge running the CSP. Therefore, if all the agents received the token and the last one did not find a solution, no solution exists and the process can stop by sending end-process messages to all the agents. Otherwise, the process can proceed to gradually improve the initial feasible structure.

This procedure allows to detect during the first round whether there is a feasible coalition structure, *i.e.* whether the characteristics of the available agents in A are sufficient to accomplish the tasks in T. Subsequently, either we seek afterward to improve the performance of the coalition structure based on a decentralized approach (knowing that we have at least one feasible coalition structure) or we abandon the effort by avoiding consuming resources unnecessarily. Thus, by not considering the maximization of u_{global} when looking for a feasible coalition structure during the first round, this procedure avoids losing time and computation by trying to maximize the global utility for a problem for which we could discover later on that it has no solution considering the available resources (*i.e.*,

the agents A). That also allows an anytime method, which finds an applicable solution in a minimal time and then improves it gradually.

Feasibility Check

To check if a coalition structure S is feasible or not, we also use a CSP. The modelization of the CSP consists of a set of variables representing the agents' assignments, their respective domains where each represents an empty set or the task that is allocated to the agent depending on the agent's assignment in the coalition structure S, and the constraints corresponding to local and global requirements.
If the coalition structure S is feasible, the solver returns it: this assignment satisfies the problem's constraints, meaning that all the requirements are fulfilled by the agents' coalitions in S and thus that S is feasible. Otherwise, it returns that no solution exists, meaning that the coalition structure violates one or many constraints of the problem, making it unfeasible.

Solution Generation Method

The *s-f-st* procedure allows the deciding agent to find a feasible coalition structure or to conclude the temporary non-existence of such an allocation. This agent uses the knowledge it has of agents characteristics to look for the distribution of the tasks among agents, represented by a coalition structure, where the requirements in Γ_T and Λ_{cbt} are fulfilled. It is a sort of locally centralized feasible coalition structure generator that only considers feasibility but not utility.
The goal is to find a feasible coalition structure where agents are assigned to tasks in T. A coalition structure feasibility results from fulfilling the tasks and task combinations requirements by its coalitions. Many centralized coalition formation solutions can answer this need, as the methods of Shehory and Kraus (1998). We have chosen to use a CSP to make use of the mirroring between task requirements and constraints.
A CSP is normally defined by three elements: a set of variables, a set of these variables' respective domains of values and a set of constraints. In our CSP model, variables represent agents' assignments and each variable's domain is the set of singletons of tasks in T and the empty set. This way, the decision of an agent, represented by its corresponding variable, can have as a value one of the tasks, if a task is allocated to him, or no task otherwise. This allocation of tasks can be seen as a coalition formation where each task has a coalition composed of agents whose variables have this task as a value. Finally, the constraints of our CSP implement the single task requirements and the task combination requirements satisfaction for the agents' characteristics. The satisfaction of a constraint is equivalent to the fulfillment of its requirement.

5.2.2 Improved FICSAM

As stated in the previous section, the algorithm's first stage consists in finding a feasible coalition structure to allow agents to have a decision that deals with the problem's constraints represented by the task requirements. This decision corresponds to an anytime solution. The second stage incrementally improves feasible coalition structures in terms of utility. It starts from the second round, once S^f was found in round 1.

In this stage, agents examine improvements in u_{global} through replacements or swaps between single or groups of agents. The resulting improved structure must preserve the feasibility of the improved solution for both single tasks and task combinations requirements.
To this end, agent a_i can use different algorithms to search for better solutions. It can try to swap its place with another agent or replace another agent allocated to a different task. It may also swap its place with a certain agent in another coalition. And after each swapping trial, it should check if the coalition structure that results from this possible change has a greater utility than the old coalition structure. The next sections present in-depth that different versions exist of the

improvement trials. After the call of one of these improvement algorithms, the agent checks the returned coalition structure's feasibility. If the algorithm returns an improved feasible coalition structure, then this coalition structure becomes the current best feasible one and it corresponds to a new anytime feasible coalition structure for all the agents. If the returned structure is not feasible, the agent reconsiders the feasible coalition structure that it received from the previous agent.

Swap Improvement Process

Once agent a_i receives the token, it assumes that the received structure S maximizes u_{global}. It then checks whether it can contribute to improving the global utility u_{global} by exploring two possibilities.
The first possibility consists of assigning agent a_i to another task (and thus to another coalition). To do so, agent a_i examines the possibility to move to a coalition of one of the tasks that it can contribute to. For each of these tasks, let's say t_j, it checks if its move to coalition C_{t_j} can result in a coalition structure that has a greater utility than the coalition structure with the best utility until now. This is built on the idea that there may be a task $t_j \in T_{a_i}$ that, if a_i contributes to its performance instead of its contribution to its current task in S, u_{global} increases.
The second possibility consists of switching the agents. Two possible scenarios are tested: the deciding agent can either take the place of another agent already allocated in a coalition, whereas the replaced agent leaves the coalition structure; or swap places with another agent. If such a change increases the global utility u_{global} of the resulting coalition structure, the resulting coalition structure is kept as the best current coalition structure.

Combinations Improvement Process

Instead of the one-to-one swap proposed in the previous algorithm, agents may try to swap with many agents in their efforts to improve the coalition structure utility.
Contrary to the previous algorithm where deciding agents consider only one-to-one swapping with other agents, we propose now to increase the search space by considering many-to-one swaps. For this, an agent a_i checks whether, for each task $t_j \in T_{a_i}$, it is beneficial to replace each member of the coalition C_{t_j} assigned to t_j by a combination cb of agents that includes a_i whose characteristics match the requirements of task t_j. As previously, these changes should only take place if they increase the global utility of the coalition structure.
This algorithm, with its many-to-one swapping trials, allows for discovering a wider part of the search space on the possible coalition structures. Nonetheless, due to its combinatorial complexity, it does not scale well to large problems.

CSP Improvement Process

To deal with the highly combinatorial problem related to the many-to-one or many-to-many swaps, we propose to use a CSP approach, by using single task requirements as hard constraints (*i.e.*, required to be satisfied) to reduce the search space. The idea is to eliminate the coalition structures in the search space that do not satisfy the problem's constraints and thus the tasks' requirements before processing them and calculating their utility.
Selecting the optimal agents' subset according to the requirements of the tasks and the preferences dictated by the utility function in the application amounts to computing optimal subsets of a set of items. For that, Binshtok et al. (2007) proposed the BB-CSP algorithm that allows finding the optimal items' subset regarding a preference specification.
A preference specification is a description of the problem's elements, the item subsets, and a preference order over the properties of these subsets. BB-CSP introduces a formalism for the preference specification. It starts with the item properties designation based on which the set

properties $(P_k)_{k\leq n}$ are defined. Once the set properties are defined, set preferences are built over the comparison of the values of these set properties through conditional value preference statements or relative importance statements. The conditional value preference statement is when for specific values of the properties $P_{i_1}, P_{i_2}, ..., P_{i_j}$, we prefer a value p_k for the property P_k over a value p'_k. The relative importance statements are when for specific values of the properties $P_{i_1}, P_{i_2}, ..., P_{i_j}$, the property P_k is more important than the property P_l and thus we prefer a better value for P_k even if we compromise on P_l's value. We can set preferences defining an order over the properties and their values based on these comparison statements. This order generates a tree with different properties combinations. Each node in this tree represents a combination of properties and is associated with a set of candidate subsets. The leaf nodes represent combinations composed of all the properties and assign a value for each property in $(P_k)_{k\leq n}$.
For example, let us consider a preference specification where there is only two set properties P_1 and P_2, that are both boolean (*i.e.*, they either take the value true or false and we note P_i when it is true and $\overline{P_i}$ otherwise) and where P_1 is more important than P_2 and true properties are the preferred ones. The resulting tree is as presented in Figure 5.3.

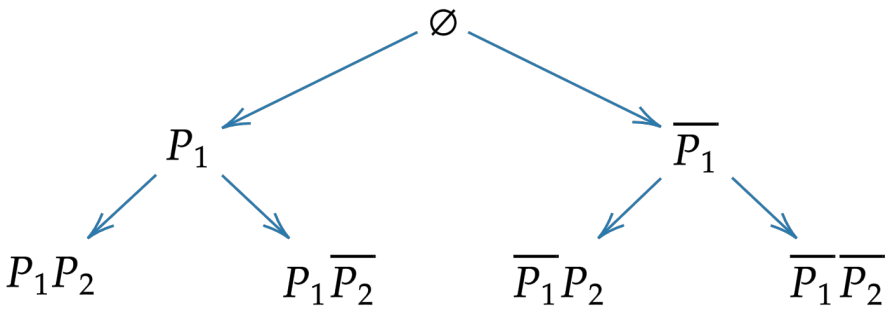


Figure 5.3: Example of a tree of CSP in Binshtok et al. (2007)

In searching for an optimal set, Binshtok et al. (2007) proposed a Branch & Bound search. This is used to prune the nodes whose property combinations are sub-optimal. Then, a CSP is run in each tree node and in the specified order by the preference tree. In this CSP, the variables representing items take 1, if the item in question appears in the output subset or 0 otherwise. The CSP aims to look for a subset of items that have associated preferred properties to its node. Since each tree node is mapped to a CSP, the entire tree is viewed as a tree of CSP. The B&B on the tree-of-CSP enables finding an optimal subset of items denoted W_{opt} regarding the preference specification.

In our model, the items are the agents, the item properties are the agent characteristics, and the set properties are the requirements' state of fulfillment by the set of agents in the set. We can, in addition, define some new desired requirements that can lead to utility increases. For example, if a given task requires at least two agents, but the more agents we have, the greater the utility function, we can define the desired requirement of having four agents.
In Binshtok et al. (2007), only preferences are considered. Hence, the set properties preferred values can be seen as soft constraints, and the goal is to satisfy the maximum preferred ones. In our problem, however, for a given task $t_j \in T$, t_j' requirements are hard constraints that we denote H_{t_j}. Besides, we add some soft constraints, denoted as P_{t_j} for each task $t_j \in T$, in a specific decreasing order of importance (*i.e.* from the more important to the less important), helping orient the search towards the subset of agents that, by doing the concerned task, increases the global utility. Hence, the main adaptation concerns the addition of the hard constraints in the preference specification and thus in the CSP's tree. Instead of having an empty root for the tree of CSP, we are starting with the set of true properties representing the hard constraints we want our subsets of agents to satisfy. This guarantees the assigned coalition's local feasibility. In Figure 5.4 on the facing page, an example

of the tree of CSP over which we apply our adaptation of the algorithm in Binshtok et al. (2007), where a task has three task requirements ($|\Gamma_{t_j}| = 3$), and thus three hard constraints and two soft constraints were added based on the utility function.

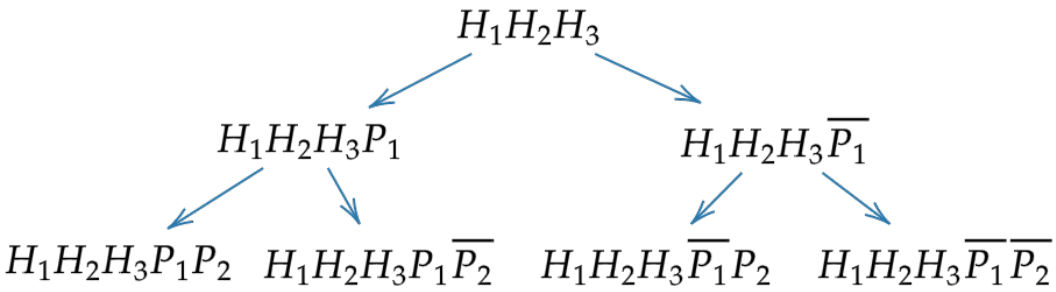


Figure 5.4: Example of the application tree

5.3 Experimental results

We illustrate the added value of our approach and evaluate its performance by benchmarking on a sample application. We also compare with performances obtained using a centralized method.

Scenarios

We evaluate our approach with a set of scenarios generated automatically. These are used to benchmark *FICSAM*, *IFICSAM* (the one-to-one and many-to-one variants) and a centralized solution.

The scenarios are generated according to the following settings: a fleet of Unmanned Aerial Vehicles (UAVs) is assigned a mission in a seaport. The (UAV) agents must inspect the hulls of boats in the port. The UAVs are relying on Unmanned Surface Vehicles (USVs), where they can charge. There are typically tens of UAV agents. However, to stretch-test our approach and compare it to a centralized approach, we experimented with up to 100 agents and 20 tasks. The USVs are scattered across the port so that the UAVs can easily charge for handling new tasks. Our scenario generator implements this by randomly positioning USVs (with a uniform distribution) on the port grid. We assume that the drones are initially uniformly positioned, but different distributions can be plausible depending on the application.

Inspection tasks may require various sensors. Here, we rely on two sensor types: HD cameras and LASERs (see for example, Agnisarman et al. (2019)). The quantity of each resource required by a task depends on the boat hull. In our scenario generator, the number of resources of each type is uniformly sampled between 0 and $\frac{n_{agents}}{2 \cdot n_{tasks}}$. This maintains scenario diversity and simplifies comparison across scenarios and settings, as averages are the same and can be compared without normalization. In addition to resource constraints, the generator introduces task interdependence via constraints on sets of tasks. Finally, each task has a deadline.

The scenario generator also generates UAV agents. For the sake of simplicity, all UAVs have the same maximum speed. Therefore, the travel time to a task is proportional to the distance between the task and the UAV. Given a grid size G, the generator randomly and uniformly draws UAV distances from $[G/2, G]$.

The agents are provided with sensors such that 25% of them have both sensors, 37.5% have only a LASER, and 37.5% have only an HD camera.

The token-passing strategy implemented is based on inter-agent distances. An agent holding the token sends it to the nearest agent that has not yet received the token in the current round.

Finally, the global utility function is a normalized additive function computed for tasks and task combinations by accumulating their values, meeting their requirements, matched against coalitions' and agents' characteristics. Specifically, the utility function we used is global and includes the agents' different properties regarding their allocated tasks. It sums the average of each of the number of agents that respect the maximal distance defined by their tasks by a certain margin, the number of allocated agents that have at least a resource, the number of agents, the number of tasks with a supplementary agent in their coalitions and the number of tasks with coalitions containing agents with a lifespan that exceeds a certain threshold.
The generator produces both feasible and infeasible scenarios. The latter is of interest for method comparison, as it requires that the algorithms prove unsatisfiability, which might take a long time. It is to be reminded that a centralized approach does not apply to the problem settings of these scenarios, mainly to avoid the single point of failure problem as a result of the distributed nature of the drones and the possibility of communication failures.

Setup

To understand the impact of the number of agents $|A|$ and tasks $|T|$, simulations are performed considering sample mission scenarios with $|A| \in [5,100]$ and $|T| \in [2,20]$. Tasks are handled by multiple agents. That is why we consider that the number of agents is always larger than the number of tasks, hence $|A| > |T|$. The reported results include algorithms execution time taken by the algorithms to terminate, the utilities of the structures they return and the number of exchanged messages.
For each $\{|A|,|T|\}$ pair, we executed 200 runs. In each run, the agent characteristics and task requirement attributes are randomly generated.
The execution platform was a 3.70 GHz Intel(R) Core(TM) i9-10900X CPU running Python, the MiniZinc toolchain and the Chuffed solver.

Centralised

To compare the results and performances of our algorithms, we developed a decentralized method. Our decentralized approach allows agents to make local decisions and avoid mission crashes because of a single point of failure. However, the comparison to a centralized solution facilitates evaluating our solution's distance from optimum and execution time.
For the centralized method, we modeled our allocation problem as an lCOP and solved it with the Lazy Clause Generation-based constraint solver Chuffed (Chu et al., 2018) through the constraint modeling language MiniZinc (Nethercote et al., 2007).
Since the proof of unsatisfiability or the proof of optimality might be very long, we instrumented our code with a timeout. Whenever the search is interrupted, we consider the problem as unsatisfiable for the first case and as the best solution found so far for the second case. In addition, to prevent pathological cases, we filter the instances with a series of necessary and sufficient conditions (for example, if the number of available agents is less than the sum of the required number of agents in all the tasks, it is useless to run the COP). Hence, no need to call the solver in such cases, which might take a long time to prove unsatisfiability.

Swap Improvement Process

We present in the following tables 5.1 and 5.2 the results of the application of our algorithms FICSAM, IFICSAM with the one-to-one swapping variant, and the aforementioned centralized approach for each metric. Every single result in the table is an average of experiments with 200 randomly generated scenarios by the scenario generator. The three methods were all tested on the same scenarios. The remainder of this section presents the results in terms of global utility value

for the system, runtime, and the number of messages exchanged for the decentralized version (this metric has no meaning for the centralized approach).

Nb of Agents	Nb of Tasks	Average Utilities			Average Execution Time (in s)		
		FICSAM	IFICSAM	COP	FICSAM	IFICSAM	COP
5	2	0.58	0.73	0.83	0.9 (± 0.0)	1.4 (± 0.0)	0.2 (± 0.0)
10	2	0.59	0.79	0.85	1.6 (± 0.0)	3.0 (± 0.1)	0.2 (± 0.0)
10	5	0.55	0.70	0.78	1.7 (± 0.0)	2.9 (± 0.1)	48.0 (± 5.3)
20	2	0.53	0.73	0.85	3.6 (± 0.01)	6.8 (± 0.1)	0.2 (± 0.0)
20	5	0.53	0.75	0.86	3.4 (± 0.1)	6.1 (± 0.1)	0.2 (± 0.0)
20	10	0.53	0.70	0.79	42.0 (± 6.9)	43.9 (± 7.0)	1184.4 (± 9.0)
50	2	0.51	0.67	0.85	7.0 (± 0.2)	11.6 (± 0.3)	4.6 (± 6.0)
50	5	0.49	0.67	0.86	69.1 (± 11.1)	76.4 (± 11.3)	0.3 (± 0.0)
50	10	0.54	0.71	0.86	387.6 (± 25.4)	398.4 (± 25.4)	25.1 (± 7.2)
50	20	0.44	0.61	0.80	212.6 (± 25.1)	222.2 (± 25.1)	1195.7 (± 4.5)
100	2	0.48	0.62	0.85	70.2 (± 14.2)	83.4 (± 14.3)	1.8 (± 0.4)
100	5	0.48	0.61	0.86	189.7 (± 24.1)	209.3 (± 24.0)	1.2 (± 7.3)
100	10	0.48	0.63	0.86	446.5 (± 24.7)	468.9 (± 24.4)	32.0 (± 7.3)
100	20	0.53	0.65	0.83	1043.9 (± 45.8)	1073.8 (± 45.8)	1158.6 (± 12.6)

Table 5.1: Experimental results on average time and utilities for FICSAM and IFICSAM in its swap (1-to-1) variant compared with the centralized method

Not surprisingly, the utilities of the solutions generated by the decentralized approaches are below those of the centralized approach (that are optimal, except for cases when the time limit is reached). However, impressively, they are rather close to that optimum. FICSAM solution utilities, being the first feasible coalition structures agents find, are below those of IFICSAM solutions where agents continue searching for other feasible coalition structures with better utilities. The utilities of FICSAM are consistently above 50% of the utilities of the centralized approach. IFICSAM utilities are always above 70% of the utilities of the centralized approach utilities and are at 75% from optimum on average. IFICSAM's search for a better solution through inversion shows that even without reaching optimality, this policy allows finding significantly better solutions in a decentralized manner. We observed that performance slightly degrades for a large number of agents

Number of Agents	Number of Tasks	Average Number of Messages	
		FICSAM	IFICSAM
5	2	18.4	25.9
10	2	36.8	71.7
10	5	38.0	61.8
20	2	71.5	177.2
20	5	71.8	170.7
20	10	75.6	158.2
50	2	178.3	557.6
50	5	176.0	576.3
50	10	181.7	526.8
50	20	180.3	511.7
100	2	351.3	1349.9
100	5	349.7	1204.1
100	10	350.6	1419.5
100	20	362.2	1147.1

Table 5.2: Experimental results on the average number of messages for FICSAM and IFICSAM in its swap (1-to-1) variant compared with the centralised method

(50 or 100). We believe that this may result from difficulties in finding an initial solution.
Notice that our algorithms terminate quickly. For the largest instances with 100 agents and 20 tasks, despite the message exchange overhead and the computations performed by disparate agents, both FICSAM and IFICSAM are faster than the centralized algorithm on almost all the tests we performed. Further, the latter, given a 1200 seconds timeout, sometimes terminates without reaching an optimal solution. That is the case where the problem is the most combinatorial, typically when the ratio of the number of available agents per task is the tightest. Otherwise, the runtime varies across scenarios. It depends not only on the number of agents and the number of tasks but also on scenario complexity. For instance, in scenarios with a larger number of tasks, a smaller average number of agents is needed for each task. Therefore, the problem becomes simpler in some cases as its combinatorial complexity is lower, which favors decentralized algorithms. For instance, for 50 agents, scenarios with 20 tasks take less time than those with 10 tasks. Eventually, there are scenarios where the centralized solution is clearly out of the question for the execution time. Consequently, not only our approach comes close to the optimal in terms of the solution's utility but it can still provide solutions in realistic times when the centralized method fails to.
Additional tasks, keeping the number of agents fixed, increase the difficulty for agents to find the first solution and send an anytime message to other agents. This can explain the drop in the number of exchanged messages when there are more tasks for the same number of agents. When the number of tasks is very small, the problem is inverted; the number of agents required for each task is larger. This agent multiplicity produces many symmetries among the variables representing the agents in the centralized COP solution, imposing additional computation. This can explain why the centralized algorithm for 50 and 100 agents, requires more time to solve scenarios with two tasks compared to scenarios with five.
As mentioned above, some of the generated scenarios are infeasible because task requirements cannot be covered by the generated set of agents. In such cases, the number of exchanged messages in our mechanism is always twice the number of agents. This results from the number of token-passing messages, to which we add the number of end messages with the mentioned failure. The token is passed via a message to all agents, from each to the next one. Then the last one that receives the token without succeeding in finding a solution sends to its previous end message and so on until the first agent is reached.
Moreover, FICSAM and IFICSAM take significantly less time than the centralized algorithm to terminate when there is a large number of agents and tasks. For 20 agents and 10 tasks, for example, they terminate after 70 seconds on average, while the centralized algorithm takes 400 seconds. For 50 agents and 10 tasks, they terminate after 360 seconds while the centralized, interrupted by the timeout, takes 1200 seconds. This observation sheds light on the cost of computing an unsatisfiability certificate by COP methods. This cost appears significantly larger than the computational cost exhibited by the decentralized approaches presented in this paper.

CSP Improvement Process

We also wanted to put in evidence the added value of the CSP Improvement Process. We recall that this algorithm uses CSP techniques for each task to find the better coalition locally. In this way, the decision-maker agent performs a many-to-one swapping of this coalition with the one that is already assigned to that task, aiming to find another coalition structure with a greater utility. We used the same generator for the experiments, and we replaced the LASER sensors with arms. The agents with the arms can lift objects, and two armed agents are needed to lift each object. This is why we added a constraint requiring a couple of armed agents for each task since one armed agent cannot perform the lifting alone.
In addition to the application's hard constraints, we also used an ordered list of preferences (*i.e.*, soft constraints). With the first preference, we considered only agents that are not assigned to the current coalition structure to minimize the chances of breaking the feasibility of the resulting

coalition structure after the swapping. With the second preference, we considered only agents whose positions are closer to the task's position by also considering a supplementary margin. With the third preference, we considered only agents with greater autonomy than the minimum required for this task by also considering an additional margin. These preferences helped us orient the research towards feasible coalition structures with better utilities.

For each $\{|A|, |T|\}$ couple, we executed 1000 runs.

Number of Agents	Number of Tasks	Average Utilities				Cases where IFICSAM (many to 1) solutions are optimal
		FICSAM	IFICSAM		COP	
			1 to 1	many to 1		
5	2	0.46	0.48	0.85	0.88	60%
10	2	0.44	0.53	0.64	0.89	8%
10	5	0.52	0.53	0.52	0.86	6%

Table 5.3: Experimental results on average time and utilities for FICSAM and IFICSAM in its two variants (1-to-1 and many-to-many) compared with the centralised method

For small instances of our scenarios (where the number of agents does not exceed 10), the use of the CSP Improvement Process allows to find the locally best subset that increases the global utility in the five first iterations. However, for instances with more agents, the scenarios generator must be updated for generating scenarios with more constraints on combinations involving several agents, which are recurrent in real-world applications. Moreover, the number of iterations has to be configured (*i.e.*, increased) for the application time limits and the problem's size for approaching the optimal solution. Finally, more soft constraints can be added to help this approach become more efficient. This is an ongoing task.
Nevertheless, the results on small instances show that, on average, the CSP Improvement Process outperforms the one-to-one swapping in many instances just by adding one single task requiring specific combinations of agent characteristics. In this case, the first variant of our IFICSAM algorithm (*i.e.*, the one-to-one swapping) fails to obtain a coalition structure with a greater utility. The reason is that this algorithm can only add or exchange an agent with one agent each time and cannot add couples of armed agents to the coalitions.
Consequently, the CSP Improvement Process may improve global utility. Our experiments focused on small instances, but our objective for future work is to create an optimized scenario generator to demonstrate its efficiency also for larger instances.

6. Conclusion and Perspectives

This final chapter concludes this manuscript with a discussion of the various perspectives I hope to explore over the next few years.

6.1 Conclusion

Thanks to my position at Université Paris Cité and my collaborations, I have had the opportunity to work on a variety of subjects, which all have in common the study of **interactions** among intelligent agents.

The starting point for my research work was the study of Boolean games during my PhD. My recruitment to the DAI team, alongside colleagues working in the field of argumentation, led me to draw a link between games and argumentation frameworks. I then started to study interactions in a multi-agent system through argumentation, firstly in the context of the study of argumentative protocols.

I then turned my attention to argumentative semantics, and more particularly to the study of ranking-based semantics, their properties, and the proposal of new families of semantics.

More recently, I have been interested in the study of interactions in task allocation and proposed a decentralized, anytime coalition formation and task allocation mechanism.

6.2 Perspectives

I present in this section different perspectives for more or less long-term work. In the interest of brevity, I will not mention all the possibilities in all the areas of research cited in this manuscript. One of the common features of the proposals presented in this section is the desire to seize the opportunities offered by the latest advances in artificial intelligence, explainability and machine learning.

6.2.1 Explainability of argumentation protocols

We place ourselves within the framework of argumentation protocols (see e.g. Bonzon and Maudet (2011); Dupuis de Tarlé et al. (2022)). One of the aims of such argumentative debates is to enable consensual decision-making between agents with different points of view. These exchanges are also likely to enable other agents to learn and possibly adopt new beliefs (arguments). A direct application of this work is linked to collective decision-making in societies of agents (town planning, project funding within a university or laboratory...). In recent years, several debate systems have been developed on the Internet to enable users to exchange opinions (e.g. Debatepedia and Debategraph). The success of these platforms, in their current form, seems to suggest that they could become an important source of exchange and information, just as Wikipedia is at this moment.

Recently, the notion of Explainable Artificial Intelligence (XAI) has seen a resurgence of attention

from researchers. This resurgence is motivated by the fact that many AI applications have limited use, or are not appropriate at all, due to ethical concerns and a lack of confidence on the part of their users. To enable users to adopt argumentation-based systems, it is essential to be able to explain the conclusions of debates. It is necessary to be able to justify the decision proposed, or the result returned, in most applications using argumentation techniques, particularly when these applications are intended for a non-specialist audience. However, to the best of our knowledge, no work has been done to explain the acceptability of the outcome of a debate to an audience that is either expert or non-expert in argumentation.
The scientific challenges we have identified are as follows:

Enriching existing argumentative protocols by allowing agents to vote for or against arguments. Adding a voting mechanism would bring us closer to existing protocols in online platforms, most of which allow users to vote for or against a given argument. Such voting mechanisms raise several questions, particularly in terms of the semantics used: the meaning of votes is often unclear and can vary greatly depending on the platform in question. Does a vote mean that an argument is valid, that it is relevant, or that it should be accepted? All these interpretations have a meaning and can give rise to different semantics. Questions are also raised about the protocol itself: for example, is voting allowed just after an argument has been presented, or only at the very end of the debate, once all the arguments have been given? Can users delete votes? Issues relating to strategic decision-making in the arguments put forward by agents (Hadoux et al., 2015) or in votes can also be studied.

Notion of impact In order to explain the results of a protocol, it will be necessary to list and quantify the impact of the elements involved in the acceptability of arguments and the outcome of the debate. These elements concern arguments via existing relationships (attack or support), votes (positive or negative) attributed by users to arguments and/or attacks, but also, if the information is available, data on the agents taking part in the debate. Extending or defining methods based on power indices (e.g. the Shapley value) could be useful for defining several families of methods for assessing the influence of these different elements.

Evaluation Finally, we will need to test and evaluate both the protocols and the explanation methods on real data (online debates for which data is available). This will require the automatic (or semi-automatic) provision of 'good' explanations for the results of the argumentation protocols. One possible way forward comes from Miller (2019), which brings together and analyses a large body of existing work from the social sciences on the subject of human explicability.

6.2.2 Automated negotiations for online dispute resolution

In the last few years a new way of solving conflicts in different domains, called Online Dispute Resolution (ODR), has appeared as an alternative solution to the traditional way of courts. ODR platforms can help solve a dispute in an amicable, inexpensive way, without going to court. If this may be implemented differently by different administrators and may evolve, it often includes at least two stages: (i) negotiation (the claimant and respondent negotiate directly with each other through the ODR platform) and, if (i) fails, (ii) facilitated settlement (the ODR administrator appoints a neutral representative, who communicates with the parties in an attempt to settle). Therefore, in ODR the opponents still discuss and interact directly, although this is done through an online platform. The time loss, but also the anxiety and the emotional charge may still be present.
To completely avoid those factors, we propose[1] to push even further the online dispute resolution by proposing automation of the ODR process through automated argumentation-based negotiations. The links between formal argumentation and legal reasoning have already been highlighted in

[1]This idea has been proposed in an ANR (Agence nationale de la recherche) project that is currently under review

various works, such as (Al-Abdulkarim et al., 2014; Bench-Capon et al., 2005, 2009; Collenette et al., 2020; Gordon, 2005). In particular, the approach of Collenette et al. (2020) has shown that the use of argumentation to model decisions of the European Court of Human Rights was not only more efficient than approaches based on machine learning (in terms of the accuracy of the results provided) but also more easily explained.
Thus, we aim to develop an online platform where the litigants will not interact anymore directly with their opponents, but through software agents that will be acting as artificial lawyers representing the litigants. If the negotiation between (virtual) lawyers fails, then a (virtual) judge will take into consideration the arguments of the opposing parties for making a decision. The litigants will then have the possibility either to accept the agreement found by the artificial lawyers or the decision taken by the artificial judge or to refuse and go for a real trial.
In that latter case, the platform will be able to provide human lawyers representing litigants in a real trial with the appropriate laws/past cases and the argument developed during the automated dispute resolution.

The scientific challenges we have identified are as follows:

Argumentation-based reasoning mechanism The unified reasoning mechanism for the artificial lawyers and judges we consider is the result of an original integration of two different mechanisms, namely (i) a mechanism for reasoning with law/jurisdiction and (ii) a mechanism for reasoning with past cases. For (i), to express the strength of arguments, we consider using argumentation-based reasoning with rules and preferences (Kakas and Moraitis, 2003). For (ii), as the decisions on past cases should have followed some rules and considered preferences, we will extract those rules and preferences from reasoning with cases, via its argumentation-based incarnation. However, the knowledge to be used in (i) and (ii) is very broad (many jurisdictions and cases for different areas) and must be acquired by legal experts. Thus, we consider automatically extracting this knowledge from texts stored in public repositories using machine learning (ML) techniques (LeCun et al., 2015; Mochales and Moens, 2011) which will then be validated by legal experts.

Negotiation protocols and strategies For the automated negotiations, we will need to implement protocols defining the rules for the exchange of arguments by the virtual layers. For this, we will adapt the work proposed in (Kakas and Moraitis, 2006b). However, to choose the winning arguments in a negotiation, negotiators need to have a certain (often incomplete) knowledge of their opponents to adapt their strategy. It is therefore desirable to provide lawyers with such incomplete knowledge of the possible arguments that could be used by their opponents. For the representation of this incomplete knowledge, we will use the work proposed in (Dimopoulos et al., 2018). As this framework is an abstract one, the challenge and originality here will be twofold: we will have to propose a structured version of the abstract framework proposed in (Dimopoulos et al., 2021) using a deductive logic-based representation of the arguments, and we will need to propose techniques for learning/inferring argument structures with the additional requirement that certain arguments and especially certain types of attacks be considered by uncertainty.

6.2.3 Generative argumentative Artificial Intelligence

Generative pre-trained Large Language Models (LLM) have garnered much attention and interest in recent years. They have significantly contributed to advancing the state of the art in Natural Language Processing (NLP) for an impressive number of tasks as suggested by (Bang et al., 2023). However, as identified by these authors, LLMs are unreliable in performing reasoning tasks. Our objective[2] is to propose a generative argumentative AI that would combine current advances in

[2]This idea has been proposed in an ANR project that is currently under review

language modeling with "classic" generative AI methods and work carried out in computational argumentation for automated human-like reasoning. In doing so, we hope to (1) better characterize and overcome the limitations of statistical language models in reasoning tasks and (2) connect argumentation theory to NLP to automatically provide the necessary inputs to an argumentative reasoning mechanism.
Current practices focus on learning argumentation patterns from data, to use them in future situations that appear similar to those learned from. However, those methods do not facilitate justification of the reasoning and explanation of claims provided by the system. We aim to deliver learning and reproduction of human reasoning from texts, with an emphasis on argumentative reasoning. Therefore, our research hypothesis is that once we succeed in learning structures and relations between arguments, in learning to construct new arguments from learned arguments, and finally in learning patterns of argumentative reasoning, we will be able to reproduce argumentative reasoning in an automated manner. This should be done through modeling within a structured preference-based argumentation framework (e.g. Kakas and Moraitis, 2003). We will then be able to automatically build a generative AI model capable of learning to reason, to explain its reasoning trace and the knowledge on which the reasoning is based, thus explaining its decisions.

The scientific challenges we have identified are as follows:

Automatically parsing argumentation structures As a first step, we intend to take advantage of existing data sets annotated with argumentation structure to train a statistical parser.

Automated generation of new arguments The system must be able to generate new arguments to automatically improve its current knowledge by combining parts of the knowledge provided by the learned arguments and by interaction with the user.

Automated revision of argumentation theories This is necessary when some claims of the system are contradicted by new knowledge acquired either through learning from texts or through interaction with the user who could put forward compelling arguments.

Bibliography

S. Agnisarman, S. Lopes, K. Madathil, K. Piratla, and A. Gramopadhye. A survey of automation-enabled human-in-the-loop systems for infrastructure visual inspection. *Automation in Construction*, 97:52–76, 2019. (Cited on page 127.)

D. Ahmadoun. *Interdependent Task Allocation via Coalition Formation for Cooperative Multi-Agent Systems*. PhD thesis, Paris Cité, 2022. (Cited on pages 118 and 121.)

D. Ahmadoun, E. Bonzon, C. Buron, P. Moraitis, P. Savéant, and O. Shehory. Decentralized coalition structure formation for interdependent tasks allocation. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 73–80, 2021. (Cited on page 121.)

S. Airiau, E. Bonzon, U. Endriss, N. Maudet, and J. Rossit. Rationalisation of Profiles of Abstract Argumentation Frameworks. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'16)*, 2016. (Cited on page 88.)

S. Airiau, E. Bonzon, U. Endriss, N. Maudet, and J. Rossit. Rationalisation of profiles of abstract argumentation frameworks: Characterisation and complexity. *Journal of Artificial Intelligence Research*, 60:149–177, 2017. (Cited on page 88.)

L. Al-Abdulkarim, K. Atkinson, and T. Bench-Capon. Abstract dialectical frameworks for legal reasoning. In *Legal Knowledge and Information Systems*, pages 61–70. IOS Press, 2014. (Cited on page 135.)

L. Amgoud, Y. Dimopoulos, and P. Moraitis. A Unified and General Framework for Argumentation-based Negotiation. In *AAMAS07*, pages 963–970. ACM Press, 2007. (Cited on page 83.)

L. Amgoud and S. Vesic. A formal analysis of the role of argumentation in negotiation dialogues. *Journal of Logic and Computation*, 2011. (Cited on page 86.)

L. Amgoud. A replication study of semantics in argumentation. In S. Kraus, editor, *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19)*, pages 6260–6266. ijcai.org, 2019. (Cited on page 28.)

L. Amgoud and J. Ben-Naim. Ranking-based semantics for argumentation frameworks. In *Proc. of the 7th International Conference on Scalable Uncertainty Management, (SUM'13)*, pages 134–147, 2013. (Cited on pages 6, 12, 13, 14, 15, 18, 22, 25, 34 and 46.)

L. Amgoud and J. Ben-Naim. Axiomatic foundations of acceptability semantics. In *KR'16*, pages 2–11, 2016. (Cited on page 57.)

L. Amgoud, J. Ben-Naim, D. Doder, and S. Vesic. Ranking arguments with compensation-based semantics. In *Proc. of the 15th International Conference on Principles of Knowledge Representation and Reasoning, (KR'16)*, pages 12–21, 2016. (Cited on pages 6, 12, 13, 25 and 30.)

L. Amgoud, J. Ben-Naim, D. Doder, and S. Vesic. Acceptability semantics for weighted argumentation frameworks. In *IJCAI'2017*, pages 56–62, 2017a. (Cited on pages 57, 58, 78 and 79.)

L. Amgoud, P. Besnard, and S. Vesic. Equivalence in logic-based argumentation. *Journal of Applied Non-Classical Logics*, 24(3):181–208, 2014. (Cited on page 55.)

L. Amgoud, E. Bonzon, M. Correia, J. Cruz, J. Delobelle, S. Konieczny, J. Leite, A. Martin, N. Maudet, and S. Vesic. A note on the uniqueness of models in social abstract argumentation. *CoRR*, abs/1705.03381, 2017b. (Cited on page 25.)

L. Amgoud, E. Bonzon, J. Delobelle, D. Doder, S. Konieczny, and N. Maudet. Gradual semantics accounting for similarity between arguments. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference (KR'18)*, pages 88–97, 2018. (Cited on pages 56 and 59.)

L. Amgoud and C. Cayrol. Inferring from inconsistency in preference-based argumentation frameworks. *International Journal of Automated Reasoning*, 29(2):125–169, 2002a. (Cited on page 54.)

L. Amgoud and C. Cayrol. Inferring from inconsistency in preference-based argumentation frameworks. *Journal of Automated Reasoning*, 29:125–169, 2002b. (Cited on page 62.)

L. Amgoud, Y. Dimopoulos, and P. Moraitis. Making decisions through preference-based argumentation. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference (KR'08)*, pages 113–124, 2008. (Cited on pages 84, 87, 108 and 109.)

L. Amgoud and S. Vesic. Rich preference-based argumentation frameworks. *International Journal of Approximate Reasoning*, 55(2):585–606, 2014. (Cited on page 52.)

Y. Bang, S. Cahyawijaya, N. Lee, W. Dai, D. Su, B. Wilie, H. Lovenia, Z. Ji, T. Yu, W. Chung, et al. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*, 2023. (Cited on page 135.)

P. Baroni, M. Caminada, and M. Giacomin. An introduction to argumentation semantics. *The Knowledge Engineering Review*, 26(4):365–410, 2011. (Cited on pages 6, 9, 11, 12 and 17.)

P. Baroni and M. Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007. (Cited on page 53.)

P. Baroni, M. Romano, F. Toni, M. Aurisicchio, and G. Bertanza. Automatic evaluation of design alternatives with quantitative argumentation. *Argument & Computation*, 6(1):24–49, 2015. (Cited on page 58.)

R. Baumann. What does it take to enforce an argument? Minimal change in abstract argumentation. In *Proceedings of the 20th European Conference on Artificial Intelligence (ECAI-2012)*. IOS Press, 2012. (Cited on page 87.)

Z. Beck, W. Teacy, N. Jennings, and A. Rogers. Online planning for collaborative search and rescue by heterogeneous robot teams. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016)*. Association of Computing Machinery, 2016. (Cited on page 117.)

T. Bench-Capon. Value based argumentation frameworks. *arXiv preprint cs/0207059*, 2002. (Cited on page 62.)

T. Bench-Capon, K. Atkinson, and A. Chorley. Persuasion and value in legal argument. *Journal of Logic and Computation*, 15(6):1075–1097, 2005. (Cited on page 135.)

T. Bench-Capon, H. Prakken, and G. Sartor. *Argumentation in legal reasoning*. Springer, 2009. (Cited on page 135.)

T. J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003. (Cited on page 88.)

T. J. M. Bench-Capon, S. Doutre, and P. E. Dunne. Audiences in argumentation frameworks. *Artificial Intelligence*, 171(1):42–71, 2007. (Cited on pages 87 and 96.)

C. Berge. *Graphs and Hypergraphs*. North-Holland Mathematical Library, 1973. (Cited on page 108.)

P. Besnard, V. David, S. Doutre, and D. Longin. Subsumption and Incompatibility between Principles in Ranking-based Argumentation. In *29th International Conference on Tools with Artificial Intelligence (ICTAI 2017)*, pages 853–859, Boston, MA, United States, November 2017. IEEE. doi: 10.1109/ICTAI.2017.00133. (Cited on pages 14 and 18.)

P. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128 (1-2):203–235, 2001. (Cited on pages 6, 12, 25 and 79.)

P. Besnard and A. Hunter. *Elements of argumentation*, volume 47. MIT press Cambridge, 2008. (Cited on pages 13 and 35.)

M. Binshtok, R. I. Brafman, S. E. Shimony, A. Martin, and C. Boutilier. Computing optimal subsets. In *AAAI*, pages 1231–1236, 2007. (Cited on pages 125, 126 and 127.)

G. A. Bodanza and M. R. Auday. Social argument justification: Some mechanisms and conditions for their coincidence. In *Proceedings of the 10th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-2009)*. Springer-Verlag, 2009. (Cited on page 87.)

G. Boella, D. Gabbay, A. Perotti, L. van der Torre, and S. Villata. Conditional labelling for abstract argumentation. In *Proc. of TAFA'11*, 2011. (Cited on page 67.)

E. Bonzon. *Modélisation des interactions entre agents rationnels : les jeux booléens*. PhD thesis, Université Paul Sabatier, 2007. (Cited on page 104.)

E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. A Comparative Study of Ranking-based Semantics for Abstract Argumentation. In *Proc. of the 30th AAAI Conference on Artificial Intelligence (AAAI'16)*, pages 914–920, 2016a. (Cited on pages 14, 16, 17, 18 and 25.)

E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. Argumentation ranking semantics based on propagation. In *Proceedings of the 5th International Conference on Computational Models of Argument (COMMA'16)*, pages 139–150, 2016b. (Cited on pages 6, 12, 13 and 19.)

E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. A parametrized ranking-based semantics for persuasion. In *Proceedings of the International Conference on Scalable Uncertainty Management (SUM'17)*, pages 237–251, 2017. (Cited on page 35.)

E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. Combining extension-based semantics and ranking-based semantics for abstract argumentation. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference (KR'18)*, pages 118–127, 2018. (Cited on page 46.)

E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. A parametrized ranking-based semantics compatible with persuasion principles. *Argument & Computation*, 12(1):49–85, 2021. (Cited on page 35.)

E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. An empirical and axiomatic comparison of ranking-based semantics for abstract argumentation. *Journal of Applied Non-Classical Logics*, pages 1–59, 2023. (Cited on pages 18 and 25.)

E. Bonzon, C. Devred, and M.-C. Lagasquie-Schiex. Translation of an argumentation framework into a CP-Boolean game. In *21st IEEE International Conference on Tools with Artificial Intelligence (ICTAI'09)*, pages 522–529, 2009a. (Cited on pages 108 and 109.)

E. Bonzon, C. Devred, and M.-C. Lagasquie-Schiex. Argumentation and cp-boolean games. *International Journal on Artificial Intelligence Tools (IJAIT)*, 19(4):487–510, 2010. (Cited on pages 108 and 109.)

E. Bonzon, Y. Dimopoulos, and P. Moraitis. Knowing each other in argumentation-based negotiation. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'12)*, pages 1413–1414, June 2012. Short Paper. (Cited on page 83.)

E. Bonzon, M.-C. Lagasquie-Schiex, J. Lang, and B. Zanuttini. Boolean games revisited. In *17th European Conference on Artificial Intelligence (ECAI'06)*, pages 265–269. Springer-Verlag, 2006. (Cited on page 104.)

E. Bonzon, M.-C. Lagasquie-Schiex, J. Lang, and B. Zanuttini. Compact preference representation and Boolean games. *Journal of Autonomous Agents and Multi-Agent Systems*, 18(1):1–35, 2009b. (Cited on pages 104, 105, 106 and 107.)

E. Bonzon and N. Maudet. On the outcomes of multiparty persuasion. In *Proceedings of the 10th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'11)*, pages 47–54, May 2011. (Cited on pages 61 and 133.)

E. Bonzon, N. Maudet, and S. Moretti. Coalitional games for abstract argumentation. In *Proceedings of the 5th International Conference on Computational Models of Argument (COMMA'14)*, pages 161–172, 2014. (Cited on page 114.)

R. Booth, S. Kaci, and T. Rienstra. Property-based preferences in abstract argumentation. In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT-2013)*. Springer-Verlag, 2013. (Cited on page 87.)

C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, and D. Poole. CP-nets : A Tool for Representing and Reasoning with Conditional *Ceteris Paribus* Preference Statements. *Journal of Artificial Intelligence Research*, 21:135–191, 2004a. (Cited on pages 105 and 107.)

C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, and D. Poole. Preference-Based Constrained Optimization with CP-nets. *Computational Intelligence*, 20(2):137–157, 2004b. Special Issue on Preferences. (Cited on page 105.)

C. Boutilier, R. I. Brafman, H. H. Hoos, and D. Poole. Reasoning with Conditional Ceteris Paribus Preference Statements. In *Uncertainty in Artificial Intelligence (UAI'99)*, 1999. (Cited on page 105.)

D. Bouyssou, T. Marchant, M. Pirlot, P. Perny, A. Tsoukias, and P. Vincke. *Evaluation and decision models: a critical perspective*, volume 32. Springer Science & Business Media, 2000. (Cited on page 120.)

G. Brewka. Dynamic argument systems: A formal model of argumentation processes based on situation calculus. *Journal of logic and computation*, 11(2):257–282, 2001. (Cited on page 62.)

M. Caminada and G. Pigozzi. On judgment aggregation in abstract argumentation. *Autonomous Agents and Multi-Agent Systems*, 22(1):64–102, 2011. (Cited on page 75.)

M. Caminada. On the issue of reinstatement in argumentation. In *Proc. of the 10th European Conference on Logics in Artificial Intelligence (JELIA'06)*, pages 111–123, 2006. (Cited on pages 9 and 11.)

C. Cayrol, S. Doutre, and J. Mengin. On decision problems related to the preferred semantics for argumentation frameworks. *Journal of logic and computation*, 13:377–403, 2003. (Cited on page 108.)

C. Cayrol, S. Doutre, M. Lagasquie-Schiex, and J. Mengin. "Minimal defence": A refinement of the preferred semantics for argumentation frameworks. In *Proceedings of the 9th International Workshop on Non-Monotonic Reasoning (NMR-2002)*, pages 408–415, 2002. (Cited on page 88.)

C. Cayrol and M. Lagasquie-Schiex. Graduality in argumentation. *Journal of Artificial Intelligence Research*, 23:245–297, 2005. (Cited on pages 6, 12, 13, 14, 16, 25, 26, 27, 34 and 59.)

C. Cayrol and M.-C. Lagasquie-Schiex. Weighted argumentation systems: A tool for merging argumentation systems. In *Proc. of ICTAI'11*, pages 629–632, 2011. (Cited on page 75.)

F. Cerutti, M. Vallati, and M. Giacomin. Afbenchgen2: A generator for random argumentation frameworks. 2017. (Cited on page 29.)

G. Chu, P. J. Stuckey, A. Schutt, T. Ehlers, G. Gange, and K. Francis. Chuffed, a lazy clause generation solver. https://github.com/chuffed/chuffed, 2018. (Cited on page 128.)

M. Clavel, F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer, and J. F. Quesada. The maude system. In *RTA'99*, pages 240–243, 1999. (Cited on page 68.)

J. Collenette, K. Atkinson, and T. J. Bench-Capon. An explainable approach to deducing outcomes in european court of human rights cases using adfs. In *COMMA*, pages 21–32, 2020. (Cited on page 135.)

S. Coste-Marquis, C. Devred, S. Konieczny, M.-C. Lagasquie-Schiex, and P. Marquis. On the Merging of Dung's Argumentation Systems. *Artificial Intelligence*, 171:740–753, 2007. (Cited on pages 62, 64, 75, 87 and 95.)

S. Coste-Marquis, S. Konieczny, P. Marquis, and M.-A. Ouali. Selecting extensions in weighted argumentation frameworks. In *Proc. of the 4th International Conference on Computational Models of Argument (COMMA'12)*, pages 342–349, Vienna, sep 2012. (Cited on page 52.)

A. Cournot. *Recherches sur les Principes Mathematiques de la Theorie des Richesses*. 1838. (Cited on page 100.)

C. da Costa Pereira, A. Tettamanzi, and S. Villata. Changing one's mind: Erase or rewind? In *Proc. of the 22nd International Joint Conference on Artificial Intelligence, (IJCAI'11)*, pages 164–171, 2011. (Cited on pages 6, 12, 26 and 58.)

R. Dechter. *Constraint processing*. Elsevier Morgan Kaufmann, 2003. (Cited on page 123.)

J. Delobelle. *Ranking-based Semantics for Abstract Argumentation*. Theses, Université d'Artois, December 2017. (Cited on pages 9, 17 and 18.)

Y. Dimopoulos, J.-G. Mailly, and P. Moraitis. Control argumentation frameworks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. (Cited on page 135.)

Y. Dimopoulos, J.-G. Mailly, and P. Moraitis. Arguing and negotiating using incomplete negotiators profiles. *Autonomous Agents and Multi-Agent Systems*, 35:1–40, 2021. (Cited on page 135.)

S. Doutre and J. Mengin. Preferred Extensions of Argumentation Frameworks: Computation and Query Answering. In A. L. R. Goré and T. Nipkow, editors, *IJCAR 2001*, volume 2083 of *LNAI*, pages 272–288. Springer-Verlag, 2001. (Cited on page 108.)

S. Doutre. *Autour de la sémantique préférée des systèmes d'argumentation*. Thèse, Université Paul Sabatier, IRIT, 2002. (Cited on page 109.)

P. M. Dung, P. Thang, and F. Toni. Towards argumentation-based contract negotiation. In *COMMA08*, pages 134–146, 2008. (Cited on page 83.)

P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-persons games. *Artificial Intelligence*, 77:321–357, 1995. (Cited on pages 6, 9, 10, 46, 58, 61, 67, 76, 97, 103 and 109.)

P. Dunne, A. Hunter, P. McBurney, S. Parsons, and M. Wooldridge. Inconsistency tolerance in weighted argument systems. In *Proc. of the 8th Int. Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'09)*, pages 851–858, 2009. (Cited on page 62.)

P. E. Dunne and T. J. Bench-Capon. Complexity and combinatorial properties of argument systems. Technical report, University of Liverpool, Department of Computer Science (U.L.C.S.), 2001. (Cited on page 109.)

P. E. Dunne and T. J. Bench-Capon. Coherence in finite argument system. *Artificial Intelligence*, 141(1-2):187–203, 2002. (Cited on page 109.)

P. E. Dunne, W. Dvořák, T. Linsbichler, and S. Woltran. Characteristics of multiple viewpoints in abstract argumentation. In *Proceedings of the 14th International Conference on Principles of Knowledge Representation and Reasoning (KR-2014)*, 2014. (Cited on pages 96 and 97.)

P. E. Dunne, P. Marquis, and M. Wooldridge. Argument aggregation: Basic axioms and complexity results. In *Proceedings of the 4th International Conference on Computational Models of Argument (COMMA-2012)*. IOS Press, 2012. (Cited on page 87.)

P. E. Dunne and W. van der Hoek. Representation and Complexity in Boolean Games. In *Proceedings of the Ninth European Conference on Logics in Artificial Intelligence (JELIA'04)*, volume LNCS 3229, pages 347–359. José Júlio Alferes et João Alexandre Leite (eds), 2004. (Cited on page 104.)

L. Dupuis de Tarlé, E. Bonzon, and N. Maudet. Multiagent dynamics of gradual argumentation semantics. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'22)*, 2022. A completer. (Cited on pages 78 and 133.)

F. Y. Edgeworth. La Teoria pura del monopolio. *Giornale degli Economisti*, pages 13–31, 1897. (Cited on page 100.)

S. Egilmez, J. G. Martins, and J. Leite. Extending social abstract argumentation with votes on attacks. In E. Black, S. Modgil, and N. Oren, editors, *Proceeding of the 2nd Workshop on Theory and Applications of Formal Argumentation (TAFA'13)*, volume 8306 of *Lecture Notes in Computer Science*, pages 16–31. Springer, 2013. (Cited on page 28.)

U. Endriss and U. Grandi. Graph aggregation. *Artificial Intelligence*, 245:86–114, 2017. (Cited on page 87.)

D. Gabbay and O. Rodrigues. Equilibrium states in numerical argumentation networks. *Logica Universalis*, 9(4):411–473, 2015. (Cited on page 58.)

D. M. Gabbay. Equational approach to argumentation networks. *Argument & Computation*, 3(2-3): 87–142, 2012. (Cited on page 25.)

S. Gabbriellini and P. Torroni. Arguments in social networks. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2013)*. IFAAMAS, 2013. (Cited on page 87.)

B. P. Gerkey and M. J. Matarić. A formal analysis and taxonomy of task allocation in multi-robot systems. *The International journal of robotics research*, 23(9):939–954, 2004. (Cited on page 119.)

T. F. Gordon. A computational model of argument for legal reasoning support systems. In *Argumentation in Artificial Intelligence and Law, IAAIL Workshop Series*, pages 53–64, 2005. (Cited on page 135.)

D. Grossi and S. Modgil. On the graded acceptability of arguments. In *Proc. of the 24th International Joint Conference on Artificial Intelligence, (IJCAI'15)*, pages 868–874, 2015. (Cited on pages 6, 12, 13, 26 and 27.)

D. Grossi and S. Modgil. On the graded acceptability of arguments in abstract and instantiated argumentation. *Artificial Intelligence*, 275:138 – 173, 2019. (Cited on page 26.)

D. Grossi and W. van der Hoek. Audience-based uncertainty in abstract argument games. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI-2013)*, 2013. (Cited on page 87.)

N. Hadidi, Y. Dimopoulos, and P. Moraitis. Argumentative alternating offers. In *AAMAS'10*, pages 441–448, 2010. (Cited on pages 84, 85, 86 and 87.)

E. Hadoux, A. Beynier, N. Maudet, P. Weng, and A. Hunter. Optimization of probabilistic argumentation with markov decision models. In *International Joint Conference on Artificial Intelligence*, 2015. (Cited on page 134.)

P. Harrenstein. *Logic in Conflict*. PhD thesis, Utrecht University, 2004. (Cited on page 104.)

P. Harrenstein, W. van der Hoek, J.-J. Meyer, and C. Witteveen. Boolean Games. In J. van Benthem, editor, *Proceedings of the 8th International Conference on Theoretical Aspects of Rationality and Knowledge (TARK'01)*, volume Theoretical Aspects of Rationality and Knowledge, pages 287–298. San Francisco, Morgan Kaufmann, 2001. (Cited on page 104.)

A. Kakas and P. Moraitis. Adaptive agent negotiation via argumentation. In *AAMAS06*, pages 384–391, 2006a. (Cited on page 83.)

A. Kakas and P. Moraitis. Argumentation based decision making for autonomous agents. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '03, page 883–890, 2003. (Cited on pages 135 and 136.)

A. Kakas and P. Moraitis. Adaptive agent negotiation via argumentation. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 384–391, 2006b. (Cited on page 135.)

J. Kemeny. Mathematics without numbers. *Daedalus*, 88(4):577–591, 1959. (Cited on page 95.)

M. G. Kendall. A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93, 1938. (Cited on pages 29 and 43.)

B. Konat, K. Budzynska, and P. Saint-Dizier. Rephrase in argument structure. In *Foundations of the Language of Argumentation – the workshop at the 6th International Conference on Computational Models of Argument (COMMA 2016)*, pages 32–39, 2016. (Cited on page 55.)

S. Konieczny, P. Marquis, and S. Vesic. On supported inference and extension selection in abstract argumentation frameworks. In *Proc. of the 13th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, (ECSQARU'15)*, pages 49–59, 2015. (Cited on pages 51, 52 and 53.)

D. Kontarinis. *Debate in a multi-agent system: multiparty argumentation protocols*. PhD thesis, Paris 5, 2014. (Cited on page 61.)

D. Kontarinis, E. Bonzon, N. Maudet, and P. Moraitis. Picking the right expert to make a debate uncontroversial. In *Proceedings of the 4th International Conference on Computational Models of Argument (COMMA'12)*, pages 486–497, September 2012. (Cited on page 75.)

D. Kontarinis, E. Bonzon, N. Maudet, and P. Moraitis. Empirical evaluation of strategies for multiparty argumentative debates. In *15th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA'14)*, pages 105–122, 2014a. (Cited on page 70.)

D. Kontarinis, E. Bonzon, N. Maudet, and P. Moraitis. On the use of target sets for move selection in multi-agent debates. In *21st European Conference on Artificial Intelligence (ECAI'14)*, pages 1047–1048, 2014b. Short Paper. (Cited on page 74.)

D. Kontarinis, E. Bonzon, N. Maudet, A. Perotti, L. van der Torre, and S. Villata. Rewriting rules for the computation of goal-oriented changes in an argumentation system. In *14th International Workshop on Computational Logic in Multi-Agent Systems (CLIMA'13)*, pages 51–68, September 2013. (Cited on page 68.)

S. Kumar, J. Cheng, J. Leskovec, and V. S. Subrahmanian. An army of me: Sockpuppets in online discussion communities. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 857–866, 2017. (Cited on page 56.)

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. (Cited on page 135.)

J. Leite and J. Martins. Social abstract argumentation. In *Proc. of the 22nd International Joint Conference on Artificial Intelligence, (IJCAI'11),*, pages 2287–2292, 2011. (Cited on pages 6, 12, 25, 46 and 78.)

M. Lesot, M. Rifqi, and H. Benhadda. Similarity measures for binary and numerical data: a survey. *International Journal of Knowledge Engineering and Soft Data Paradigms*, 1(1):63–84, 2009. (Cited on page 56.)

R. Loui. Process and policy: Resource-bounded nondemonstrative reasoning. *Computational Intelligence*, 14(1):1–38, 2002. (Cited on page 61.)

L. S. Marcolino, A. X. Jiang, and M. Tambe. Multi-agent team formation: Diversity beats strength? In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, International Joint Conference on Artificial Intelligence (IJCAI '13), page 279–285. AAAI Press, 2013. ISBN 9781577356332. (Cited on page 117.)

P. Matt and F. Toni. A game-theoretic measure of argument strength for abstract argumentation. In *Proc. of the 11th European Conference on Logics in Artificial Intelligence, (JELIA'08)*, pages 285–297, 2008. (Cited on pages 6, 12, 13, 14, 26 and 27.)

T. Michalak, J. Sroka, T. Rahwan, M. Wooldridge, P. McBurney, and N. R. Jennings. A distributed algorithm for anytime coalition structure generation. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 1 - Volume 1*, AAMAS '10, page 1007–1014, 2010. (Cited on page 117.)

T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019. (Cited on page 134.)

R. Mochales and M.-F. Moens. Argumentation mining. *Artificial Intelligence and Law*, 19:1–22, 2011. (Cited on page 135.)

J. Nash. Equilibrium points in *n*-person games. *Procedings of the National Academy of Sciences*, 36:48–49, 1950. (Cited on pages 100 and 102.)

D. Navarro. *Learning statistics with R: A tutorial for psychology students and other beginners*. Open Textbook Library, 2018. (Cited on page 81.)

N. Nethercote, P. J. Stuckey, R. Becket, S. Brand, G. J. Duck, and G. Tack. MiniZinc: Towards a Standard CP Modelling Language. In C. Bessière, editor, *Principles and Practice of Constraint Programming – CP 2007*, pages 529–543, 2007. (Cited on page 128.)

M. J. Osborne. *An Introduction to Game Theory*. Oxford University Press, 2004. (Cited on pages 99 and 102.)

M. J. Osborne and A. Rubinstein. *A course in game theory*. MIT Press, 1994. (Cited on pages 99 and 102.)

S. Parsons, C. Sierra, and N. R. Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261—292, 1998. (Cited on page 83.)

S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of some formal inter-agent dialogues. *Journal of Logic and Computation*, 13(3):347–376, 2003. (Cited on page 61.)

T. Patkos, A. Bikakis, and G. Flouris. A multi-aspect evaluation framework for comments on the social web. In *Proc. of the 15th International Conference on Principles of Knowledge Representation and Reasoning (KR'16)*, pages 593–596, 2016. (Cited on page 12.)

F. H. Poletiek. *Hypothesis-testing behaviour*. Psychology Press, 2013. (Cited on page 80.)

H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, 15(6):1009–1040, 2005. (Cited on page 67.)

H. Prakken. Formal systems for persuasion dialogue. *The Knowledge Engineering Review*, 21(02): 163–188, 2006. (Cited on pages 61, 62 and 79.)

F. Pu, J. Luo, and G. Luo. Some supplementaries to the counting semantics for abstract argumentation. In *Proceedings of the 27th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'15)*, pages 242–249, 2015a. (Cited on page 26.)

F. Pu, J. Luo, Y. Zhang, and G. Luo. Argument ranking with categoriser function. In *Proc. of the 7th International Conference on Knowledge Science, Engineering and Management, (KSEM'14)*, pages 290–301, 2014. (Cited on pages 6, 13 and 25.)

F. Pu, J. Luo, Y. Zhang, and G. Luo. Attacker and defender counting approach for abstract argumentation. In *Proc. of the 37th Annual Meeting of the Cognitive Science Society, (CogSci'15)*, 2015b. (Cited on pages 12, 13, 26 and 37.)

S. Pulfrey-Taylor, E. Henthorn, K. Atkinson, A. Wyner, and T. J. M. Bench-Capon. Populating an online consultation tool. In *Proceedings of the 24th Annual Conference on Legal Knowledge and Information Systems (JURIX-2011)*, pages 150–154. IOS Press, 2011. (Cited on page 96.)

A. Rago and F. Toni. Quantitative argumentation debates with votes for opinion polling. In *PRIMA 2017: Principles and Practice of Multi-Agent Systems - 20th International Conference, Nice, France, October 30 - November 3, 2017, Proceedings*, volume 10621 of *Lecture Notes in Computer Science*, pages 369–385. Springer, 2017. (Cited on pages 82 and 96.)

A. Rago, F. Toni, M. Aurisicchio, and P. Baroni. Discontinuity-free decision support with quantitative argumentation debates. In *KR'16*, pages 63–73, 2016. (Cited on page 58.)

I. Rahwan and K. Larson. Pareto optimality in abstract argumentation. In *Proc. of the 23rd Conference on Artificial Intelligence (AAAI'08)*, pages 150–155, 2008. (Cited on pages 62, 64 and 75.)

I. Rahwan and K. Larson. *Argumentation and Game Theory*, chapter Argumentation in Artificial Intelligence, pages 321–339. Springer, 2009. (Cited on page 61.)

I. Rahwan and F. A. Tohmé. Collective argument evaluation as judgement aggregation. In *Proc. of the 10th Int. Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'10)*, pages 417–424, 2010. (Cited on page 62.)

T. Rahwan, T. P. Michalak, M. Wooldridge, and N. R. Jennings. Coalition structure generation: A survey. *Artificial Intelligence*, 229:139–174, 2015. (Cited on page 117.)

A. Rubinstein. Perfect Equilibrium in a Bargaining Model. *Econometrica*, 50:97–110, 1982. (Cited on page 85.)

L. S. Shapley. A value for n-person games. In H. Kuhn and A. Tucker, editors, *Contributions to the Theory of Games, Vol. II*, volume 28 of *Annals of Mathematics Studies*, pages 307–317. Princeton University Press, Princeton, NJ, 1953. (Cited on page 103.)

O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. *Artificial intelligence*, 101(1-2):165–200, 1998. (Cited on pages 117 and 124.)

B. Stein. Report of dagstuhl seminar debating technologies. Technical report, Report of Dagstuhl Seminar Debating Technologies, 2016. (Cited on page 55.)

C. Tan, V. Niculae, C. Danescu-Niculescu-Mizil, and L. Lee. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th International Conference on World Wide Web, (WWW'16)*, pages 613–624, 2016. (Cited on page 37.)

P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*, chapter Cluster Analysis: Basic Concepts and Algorithms. (Cited on page 30.)

M. Thimm and G. Kern-Isberner. On controversiality of arguments and stratified labelings. In *Proc. of the 5th International Conference on Computational Models of Argument, (COMMA'14)*, pages 413–420, 2014. (Cited on page 34.)

M. Thimm and S. Villata. First international competition on computational models of argumentation (iccma'15). see http://argumentationcompetition.org/2015/, 2015. (Cited on page 43.)

F. A. Tohmé, G. A. Bodanza, and G. R. Simari. Aggregation of attack relations: A social-choice theoretical analysis of defeasibility criteria. In *Proceedings of the 5th International Symposium on Foundations of Information and Knowledge Systems (FoIKS-2008)*, pages 8–23. 2008. (Cited on page 87.)

F. Toni and P. Torroni. Bottom-up argumentation. In *TAFA*, pages 249–262, 2011. (Cited on page 75.)

J. Tremblay and I. Abi-Zeid. Value-based argumentation for policy decision analysis: Methodology and an exploratory case study of a hydroelectric project in Québec. *Annals of Operations Research*, 236(1):233–253, 2016. (Cited on page 97.)

S. Vesic, B. Yun, and P. Teovanovic. Graphical representation enhances human compliance with principles for graded argumentation semantics. In *21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS'22)*, pages 1319–1327, 2022. (Cited on page 34.)

L. Vig and J. Adams. Multi-robot coalition formation. *IEEE Transactions on Robotics*, 22(4): 637–649, 2006. doi: 10.1109/TRO.2006.878948. (Cited on page 117.)

J. von Newmann and O. Morgenstern. *Theory of Games and Economic Behaviour*. Princeton University Press, 1944. (Cited on page 100.)

M. Wooldridge, P. E. Dunne, and S. Parsons. On the complexity of linking deductive and abstract argument systems. In *AAAI-06*, pages 299–304, 2006. (Cited on page 55.)

Y. Wu and M. Caminada. A labelling-based justification status of arguments. *Studies in Logic*, 3 (4):12–29, 2010. (Cited on pages 12 and 48.)

W. S. Zwicker. Introduction to the theory of voting. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 2. Cambridge University Press, 2016. (Cited on page 95.)