An Alternative Semantics for Argumentative Systems

Alejandro G. Stankevicius, Guillermo R. Simari

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial
Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur
Bahía Blanca - Buenos Aires - ARGENTINA
{ags, grs}@cs.uns.edu.ar

Abstract

Defeasible argumentation is one of the approaches that addresses the challenges arising when we reason defeasibly, with several formalisms in the literature reaching a mature state. Considering that most of these theories eventually shifted their semantics towards dialectical characterizations, we believe that a sufficiently generic model of the process of reasoning in dialectical terms could also serve as an abstract model of what happens inside an argumentative system. To that end, we develop in this article a formal model of dialectical reasoning and explore its role as an alternative semantics for argumentation theories.

Keywords: knowledge representation, defeasible reasoning, argumentation systems, dialectics.

1 Introduction

The start of the 80's mark a revolution in the Knowledge Representation and Reasoning (KR&R) field. For thousands of years, KR&R formalism were required to be *monotonic*, in the sense that current results should not be invalidated by the addition of new premises. However, most intelligent beings appear to contradict this reasoning pattern, showing non-monotonic features in their regular behavior. Suddenly, monotonic reasoning became a burden, deemed to be avoided [6, 7, 16, 8]. To overcome this, a new kind of reasoning, called *defeasible reasoning* (a form of non-monotonic reasoning), emerged as a new field within the KR&R community. In this context, *defeasible argumentation* [9] is one of the approaches that addresses the challenges arising when we reason defeasibly.

Nowadays, defeasible argumentation has become one of the hottest topics in defeasible reasoning, with several formalisms in the literature reaching a mature state [19, 3, 25, 14, 10, 4]. Given its common origin, most of these system share several key aspects among them. For instance, in the article surveying the state of the art in argumentation systems [15], Prakken and Vreeswijk identified the following core notions as common to every formalism:

1. They provide an *underlying logic*.
2. They formally define the concept of *argument*.
3. They capture when two arguments are in *conflict*.
4. They define when that conflict leads to a *defeats*.
5. They provide a mechanism for determining the *ultimate state* of arguments.

Every argumentation system begins by defining an *underlying logic* where knowledge is expressed. Even though these theories behave non-monotonically,

most of these underlying logics tend to define a monotonic entailment relation. The rationale is that the addition of new premises should allow the construction of new *arguments* (possibly changing the conclusions sanctioned by the system as a whole), without requiring the withdrawal of previous deductions. These arguments, the second notion in common, are usually associated with proofs or deductions in the underlaying logic. Simply put, an argument is a tentative piece of reasoning supporting a given conclusion. However, not every conclusion drawn on this logic leads to a valid argument, since these systems usually impose additional restrictions on those derivations, such as being based on non-contradictory premises, being minimal, etc.

Regarding the third common aspect, every system usually allows us to construct arguments for *conflicting* conclusions. This relation among arguments is also called *attack* or *counter-argumentation* in some systems. Although the most obvious form of conflict is the support of complementary conclusions, an argument may conflict with another for other reasons, for example, when the first denies a premise of the second. Considering that this relation only captures disagreement between arguments, it cannot tell appart successful attacks from those that are not. The relation called *defeat*, fourth notion common to every argumentation theory, is a refinement of the previous relation that only accounts for successfull attacks. Note that this relation captures the conditions under which an argument is able to deny the conclusive force of another argument, effectively disabling the latter. On some systems, this relation is called *attack* or *interference* as well. Even though the conflict relation is usually symmetric, the defeat relation is usually not: some sort of argument comparison criterion is applied to determine which argument prevails in those reciprocal conflicts.

Finally, defeat alone is not enough to determine the *final state* of an argument, as it only states what the outcome of the conflict between two particular arguments is. This key notion, also common to every argumentative system, is used to determine the set of conclusions sanctioned by the formalism. Therefore, several alternatives have been explored in the literature, for instance fix-point vs. constructive semantics, single vs. multiple state assignments, skeptic vs. credulous stance, etc. All these proposals identify at least two disjoint sets: one containing those arguments that are *warranted*, and the other containing those that are not. In a sense, an arguments is warranted when it is not defeated, or when its defeaters are in turn defeated, since defeaters are arguments as well. Note that this process may continue recursively, as long as additional defeaters for any argument remain to be considered. This is the reason why warranted arguments are sometime called *undefeated* in some theories, though they have been termed *justified* or just *active* too.

During the 90's, several prominent formalisms started shifting their semantics towards dialectical characterizations, abandoning their original fix-point semantics or recursive definitions. For instance, the recursive definition of *arguments active on a given level* from Simari-Loui's theory [19] later became a *dialectical analysis structured as a tree* [18], or the *fixpoint semantics* of Prakken and Sartor's theory [13] was then replaced by a *dialogical game* [14]. Nowadays, every major argumentative system has been reformulated to accommodate a dialectical variant of its semantics,[1] which generally also became the preferred way of presenting it.

Considering how successful the shift towards dialectical characterizations turned out to be, we postulate that a sufficiently generic model of the process of reasoning in dialectical terms could also serve as an abstract model of what happens inside an argumentative systems. To that end, this article lays the foundations for such a generic model, whose particular instantiations may double as an alternative semantics for several concrete argumentation systems. This generic model is initially introduced in the next section. Then, Section 3 briefly explores other works in the literature which follow a like approach. Finally, Section 4 presents the conclusions reached and outlines the work ahead.

2 Dialectical Protocols

This section develops an abstract semantics for argumentation theories which is the result of the insight that most of these formalisms also contemplate dialectical recasts of their semantics as a part of their formal definition. To do so, we begin by discussing the key aspects of dialectical argumentation in lay terms, and then, in Section 2.2, we take into account the identified aspects while introducing the set of notions that compose a *dialectical protocol*. Finally, this notion (and its particular instantiations), induce an entailment relation, which constitute the desired semantics.

[1] with the notable exception of Pollock's theory [10].

2.1 Key Aspects of Dialectical Reasoning

Suppose we want to describe what is going on inside the mind of an intelligent reasoner that implements a theory of dialectical argumentation. Let us imagine the context where this reasoning usually takes place: a given agent comes up with a new claim, and it is about to weigh the chance that claim have of being sanctioned. This claim will be scrutinized under a careful dialectical analysis, where reasons for and against it are to be considered. Evidently, this agent must be willing to assume at least two distinctive roles. Sometimes it will seek out reasons supporting this claim, but often times it will become a sort of "devil's advocate", questioning those very same reasons it just put forth. Note that this duality can be observed in most argumentation systems too, where authors have agreed to call *proponent* the role where it supports the claim under consideration, and *opponent* the role where it questions it. These reasons being argued for and against the initial claim are all based upon the same knowledge base, namely the knowledge base of the agent performing both roles.

In order to introduce the remaining concepts, we have to resort to a useful analogy also used to present the dialectical flavor of the semantics of many argumentation systems: we will conceive the whole reasoning process in dialectical terms as if it were a game, where two contenders take turns to introduce further reasons, either supporting or attacking the initial claim. This intuitive analogy has been used to convey complex concepts such as those that make up the dialectical reformulations of Simari-Loui's system [18] or Prakken-Sartor's theory [14].

Under this conception, one may wonder what a *turn* or a *move* within this dialogical game stands for. It is clear that each move should either consolidate or attack the initial claim, according to the role being performed. Considering the nature of the systems being modeled, it is safe to assume that these reasons being put forward will be structured as *logical arguments*. With this in mind, the exchange of these arguments usually explores a given aspect of the topics being disputed to the fullest extent, until no further argument can be played addressing it, to then start discussing another aspect, and another aspect, until all the aspects of the initial topic are also exhausted. This behavior is modelled in several of the theories of defeasible argumentation as the notion of *argumentation line* [14, 10, 4]. An argumentation line is a mere sequence of related moves—an argument, followed by one of its defeaters, followed in turn by a defeater of this defeater, and so on.

We believe this concept plays a key role upon which we erect our generic model of reasoning in dialectical terms. In contrast, other alternative proposals previously explored in the literature (later discussed in Section 3) assign this central role to arguments. Our decision stem from the fact that the notion of argumentation line is the smallest piece of reasoning still showing dialectical features; once we go down to the argument level, it is quite difficult to identify which role dialectics plays there. For instance, we can describe in what state a given dispute is by providing the set of all the (partial) argumention lines explored so far. In a sense, this set of argumentation lines readily represents a *snapshot of the dispute* being conducted.

Once we are able to grasp the different states a dispute can be in, another issues should also be addressed, such as:

- Which player should move next in a given configuration?
- What is the set of legal moves available in a certain state?
- Is the dispute over? If so, which contender prevailed?

These important aspects should be carefully considered, since answering these questions in way entirely reasonable for a given context may preclude the model from being applicable in some other context. We can take this into account by not forcing these decisions, only providing a scheme which should later be instantiated according to the needs at hand. For instance, every argumentation system clearly states how proponent and opponent should alternate, which arguments can be formulated in a given context, or what has been the outcome of the dialectical analysis just performed.

Finally, being able to formally capture each of the states a given dispute may traverse is not enough, as we should also consider how them relate to each other. In a sense, all the notions briefly introduced so far just model the *static* side of a dispute (for instance, which move can be played next, or which argumentation lines have been explored so far), but we have not captured its *dynamics* yet. Not whitstanding, we can take its dynamics into account by means of a transition function, formally modelling the effect of playing a certain move in the context of a particular state of a dispute. Observe that this transition function is in fact capturing the dynamics of the dialogical game, provided it relates the state of a dispute before and after playing a move.

With this last concept we complete the discussion of all the relevant aspects involved in the process of reasoning in dialectical terms. In the next section, we present a set of formal definitions accounting for all the aspects of dialectical reasoning just overviewed.

2.2 Formal Definition

According to the discussion in the previous section, a dialectical dispute regarding a given claim $\mathcal{T}$, expressed in a certain knowledge representation language $\mathcal{L}$, involves two contenders: the proponent of the claim, P, and its opponent, O. Let KB be the knowledge base available to both contenders, also expressed in terms of $\mathcal{L}$ (we only require from this language to be able to express reasons or justifications in a sound way, called in this context arguments), and let `args` be a mapping from knowledge bases into set of arguments, which provides the link between a knowledge base and the set of all the arguments one can formulate from that knowledge. The particular conditions these arguments satisfy, or the process upon which they are constructed, are aspects relevant only to particular instantiations of our model, whereas this initial formulation only concerns the framework itself. Not withstanding, since the reasoning we are modelling is of an introspective nature, where the same agent assumes both roles, we can deduce that the set of arguments available to both P and O must be `args(`KB`)`.

These notions constitute the context where the dialectical reasoning will take place, formally defined as follows:

Definition 1. (**context**)
Let us call *context* the tuple $\mathcal{C} = (\mathcal{L}, KB, \texttt{args})$, where:

- $\mathcal{L}$ is the knowledge representation language being used, arbitrary but fixed, in which it is assumed one can formulate logical arguments.
- KB is the knowledge base, coded in terms of $\mathcal{L}$, available to both contenders.
- *args* is a mapping from knowledge bases into set of arguments, relating each knowledge base with the set of all the valid arguments that can be built from it.

For the remainder of this section we will assume the existence of an implicit context $(\mathcal{L}, KB, \texttt{args})$, arbitrary but fixed. Going back to the analogy briefly introduced in the previous section between dialectical reasoning and a dialogical game involving the exchange of arguments, we begin by formally defining what each play or move represents:

Definition 2. (**move**)
Let us call *move* the tuple $(contender, argument)$, where $contender$ is either P or O, and $argument \in$ `args(`KB`)`.

Given the move $m = (contender, argument)$, we may also refer to its contender as *player*(m) and to the argument being played as *arg*(m). Now, moves that relate with each other are usually structured as argumentation lines:

Definition 3. (**argumentation line**)
Let $\mathcal{T}$ be a claim formulated in $\mathcal{L}$. We say that an *argumentation line regarding* $\mathcal{T}$ is the sequence of moves $\langle m_0, \ldots, m_k \rangle$, $k \geq 0$, such that:

- $m_0 = (P, argument)$, where $argument$ is an argument supporting $\mathcal{T}$.

Note that this definition requires from argumentation lines to start with a move from P supporting the claim being disputed. However, this definition fails to capture a subtle aspect of argumentation lines, as shown in the following example:

Example 1. Consider the arguments:

- $\mathcal{A}$ = we should fire John because he is missing work.
- $\mathcal{B}$ = John is missing work because he is on a leave of absence.
- $\mathcal{C}$ = John's cat is fluffy.

In this setting, a potential argumentation line regarding whether we should fire John may involve the moves $\langle m_0, m_1, m_2 \rangle$, where $m_0 = (P, \mathcal{A})$, $m_1 = (O, \mathcal{B})$ and $m_2 = (P, \mathcal{C})$.

Granted, not every sequence of moves constitute an argumentation line worth considering. For instance, the sequence of moves described in Exam. 1 involves unrelated arguments, which probably do not represent the exploration of an actual issue. In order to only take into account related moves, the notion of argumentation line must be refined with the help of a function called `legal`, whose purpose is to determine the set of moves that are allowed to extend a given argumentation line. Let *Lines* be the set of all the possible

argumentation lines and *Moves* the set of all the valid moves, then `legal` should be a function from *Lines* into $\mathcal{P}(Moves)$. Strictly speaking, this signature is the sole restriction imposed over this function, since its concrete definition is one of tasks involved in the actual instantiation of the framework. The function `legal` allows us to refine the notion of argumentation line as follows:

Definition 4. (**revised argumentation line**)
Let $\mathcal{T}$ be a claim formulated in $\mathcal{L}$. We say that an *argumentation line regarding* $\mathcal{T}$ is the sequence of moves $\langle m_0, \ldots, m_k \rangle$, $k \geq 0$, such that:

- $m_0 = (P, argument)$, where $argument$ is an argument supporting $\mathcal{T}$, and
- for each i, $1 \leq i \leq k$, it holds that $m_i \in$ `legal` $(\langle m_0, \ldots, m_{i-1} \rangle)$.

According to the discussion in the previous section, having formalized the concept of argumentation line allows us to tackle the state of a dispute:

Definition 5. (**state of a dispute**)
Let $\mathcal{T}$ be a claim formulated in $\mathcal{L}$. We say that the *state of dispute regarding* $\mathcal{T}$ is a non empty set of argumentation lines regarding $\mathcal{T}$.

Even though this definition imposes few restrictions over the state of a dispute, not every state will be reachable in practice. For instance, those states representing the continuation of a debate already won by one of the contenders will not be reached during an actual dispute, despite of being valid according to Def. 5. Having modeled this notion, the same issues previously considered during the discussion of the key aspects of dialectical reasoning should also be addressed. For instance, it is quite natural to ponder who should play the next move, or which contender may have won in a given state of a dispute. On the one hand, recall that providing a concrete answer to any of these issues can end up restricting the applicability of the model being proposed. On the other hand, any decision with respect to how to address them drastically affect the behavior of the model, so the policy governing each of these issues will constitute a central aspect of the dynamics of the model. To reconcile both visions, we make use of a set of abstract functions, barely characterized in the generic model, yet required to be fully specified in every concrete instantiation.

In what follows, let *States* be the set of all the possible states of a given dispute. We will make use of the following auxiliary functions:

- `toMove`: defined from *States* into $\{P, O\}$, determining which contender must play next in a given state of a dispute.
- `winner`: defined from *States* into $\{P, O\} \cup \{$`none`$\}$ determining which contender (if any) has prevailed in a given state of a dispute. When no contender has prevailed yet, this function must return the constant `none`.

The function `toMove` models the so called *burden of the proof*,[2] which changes sides during the actual debate [17]. Since this function allows us to inspect one the features of the move about to be played (namely, which contender is going to play it), it must agree with the behavior of the function `legal`, which, in a sense, also inspect other features of that move. To illustrate this point, let us consider the following situation, where these functions do not concord:

Example 2. Consider the arguments:

- $\mathcal{A}$ = we should fire John because he is missing work.
- $\mathcal{B}$ = John is missing work because he is on a leave of absence.
- $\mathcal{C}$ = John is missing work because he is on vacation.

Suppose we adopt a definition for the function `legal` allowing contenders to play any of the available arguments as long as they do not replay them, and a definition for the function `toMove` stating that contenders must take alternating turns. In this context, it is possible to explore the argumentation line about whether we should fire John $\langle m_0, m_1, m_2 \rangle$, where $m_0 = (P, \mathcal{A})$, $m_1 = (O, \mathcal{B})$, and $m_2 = (O, \mathcal{C})$, since its first move is appropriate, and the subsequent moves uphold the restrictions imposed by `legal` (i.e., both $m_1 \in$ `legal` $(\langle m_0 \rangle)$ and $m_2 \in$ `legal` $(\langle m_0, m_1 \rangle)$). Not withstanding, this sequence of moves where the opponent plays two arguments consecutively does not uphold the ordering required by `toMove`.

Simply put, to avoid the kind of conflicts suggested by the previous example, we ought to make sure that `legal` and `toMove` agree with each other. The contender playing each of the moves compossing a given

[2] this notion usually has a different meaning in courts of law.

argumentation line must be exactly whoever `toMove` would have required, should that line be the only line explored so far. That is to say, we only consider a move as valid when it is being played by whoever had to play in that situation. Formally speaking, whenever $(contender, argument) \in$ `legal(`$line$`)`, then it must also be the case that `toMove(`$\{line\}$`)` $= contender$.

The reader might wonder why not let `legal` also determine which player should play next, both simplifying the framework and rendering the above restriction unnecessary. However, recall that `legal` only concerns the current line of argumentation, whereas the function `toMove` may have to consider all the lines of argumentation being explored in the current state of the dispute. Should we merge these notions, either `legal` must take into account every moves played so far, turning the notion of line of argumentation a bit superfluous, or `toMove` will have to only refer to the moves played in the line of argumentation being explored, making impossible to enforce, for instance, that contenders must take alternating turns.

Finally, once we are able to model all the states a given dispute may traverse, we should then capture how these states relate to each other. In a sense, all the notions briefly introduced so far just model the *static* side of a dispute (for instance, which move can be played next, or which argumentation lines have been explored so far), but we have not captured its *dynamics* yet. However, we can take it into account through a transition function, formally modelling the effect of playing a given move in the context of a particular state of a dispute. This transition function by relating the state of a dispute before and after playing a given move is actually capturing the dynamics of the dialogical game we intend to model.

To do so, we must introduce the component that captures the dynamics of the dispute: the transition function `next`. This function, defined from *States* $\times$ *Moves* $\times$ *Lines* into *States*, determines which state is the outcome of playing a move extending a given argumentation line in the context of a certain state of a dispute. Note that this function is the only component of this model capable of modifying the state of a dispute. No other component can *create*, *change*, or *eliminate* argumentation lines. Following a similar approach as before, we specify this function in an abstract manner, to avoid restricting the applicability of the model. Once again, characterizing this abstract function in a concrete way is one of the key tasks involved in its instantiation.

Finally, we are ready to get together all the definitions previously introduced along with the partially defined notions as follows:

Definition 6. (**dialectical protocol**)
Let $\mathcal{C}$ be a context. We say that a *dialectical protocol* for $\mathcal{C}$, noted $\mathbb{DP}_{\mathcal{C}}$, can be characterized through the tuple

(*Moves*, *Lines*, *States*, `legal`, `toMove`, `winner`, `next`)

where:

- *Moves* is the set of moves considered valid (Def. 2).
- *Lines* is the set of argumentation lines considered valid (Def. 4).
- *States* is the set of all the possible states of a dispute (Def. 5).
- `legal` is a function defined over *Lines* into $\mathcal{P}$(*Moves*), that provides the set of moves that are allowed to extend a given argumentation line.
- `toMove` is a function defined over *States* into $\{P, O\}$, that determines which contender should move next in a given state of a dispute.
- `winner` is a function from *States* into $\{P, O\} \cup \{$`none`$\}$, that establishes which contender has prevailed in a given state of a dispute, if any. We assume that the constant `none` denotes that no one has prevailed yet (that is, the dispute is still open).
- `next` is a function defined over *States* $\times$ *Moves* $\times$ *Lines* into *States*, which captures the dynamics of the dispute, that is, it describes the effect of playing a given move in the context of a certain state of a dispute.

Moreover, in order to preserve the internal consistency of the dialectical protocol being described, whenever $(contender, argument) \in$ `legal(`$line$`)`, then it must also hold that `toMove(`$\{line\}$`)` $= contender$.

In those cases where the context being referred to in a dialectical protocol is evident, we shall note the protocol $\mathbb{DP}_{\mathcal{C}}$ just as $\mathbb{DP}$.

We have at our disposal a set of definitions that can capture the interaction that takes place inside a generic theory of defeasible argumentation. Since this interaction in fact gives birth to the semantics of the theories, we should carefully consider what ought to be concluded from a concrete instance of a dialectical protocol.

Definition 7. (**entailment**)
Let $\mathbb{DP}_{\mathcal{C}}$ be the tuple

$(\mathit{Moves}, \mathit{Lines}, \mathit{States}, \texttt{legal}, \texttt{toMove}, \texttt{winner}, \texttt{next})$

which is a particular instance of a dialectical protocol, and let $\mathcal{T}$ be a claim formulated in $\mathcal{L}$. We say that $\mathcal{T}$ is *entailed* by $\mathbb{DP}_{\mathcal{C}}$ if, and only if, there exists a finite sequence $s_0, s_1, \ldots, s_n$ of states of a dispute regarding $\mathcal{T}$, such that:

- $s_0 = \{\langle m_0 \rangle\}$, where $\langle m_0 \rangle$ is an argumentation line regarding $\mathcal{T}$,
- for each i, $0 \leq i < n$, there exists an argumentation line $\langle m_0, \ldots, m_k \rangle \in s_i$ for which it is possible to find a j, $0 \leq j \leq k$, such that there exists a move $m \in \texttt{legal}(\langle m_0, \ldots, m_j \rangle)$, that satisfies the following conditions:
  - *player*$(m) = \texttt{toMove}(s_i)$, and
  - $\texttt{next}(s_i, m, \langle m_0, \ldots, m_j \rangle) = s_{i+1}$.
- for each i, $0 \leq i < n$, $\texttt{winner}(s_i) = \texttt{none}$, and
- $\texttt{winner}(s_n) = P$.

Given the abstract nature of the entailment relation induced by concrete instances of dialectical protocols, it is possible to capture the semantics of many argumentation systems within it. For instance, using dialectical protocols one can describe Simari-Loui's justified literals [19], Defeasible Logic Programming warranted beliefs [4], or the outcome of Prakken-Sartor's dialogue game [14] among others.

3 Related work

To begin with, this approach was first suggested in a groundbreaking work of R. Loui [5]. This author was the first to consider the existence of a generic model for argument-based dialectical reasoning. He successfully introduced a set of abstract notions that were able to capture the dynamics of many attractive theories of defeasible argumentation. However, Loui involvement with this line of research was rather tangential since he was mainly concerned with adding resource-boundedness into defeasible argumentation.

Later, H. Prakken took Loui's work in dialectical reasoning into what he calls *dynamic debates* involving several agents [12]. He proposed a model of dialectical argumentation in accord to Loui's designs, somewhat resembling the scenario depicted in Section 2.1. This author did not explore what could have been done with that model, using it only as a intermediate stage, to be later reinterpreted as if it were a model multiagent interaction. In particular, Prakken's main goal was to allow the agents taking part in this interaction to dynamically modify their knowledge bases, a key requirement, for instance, when the dispute being modeled take place inside a court of law.

In a recent article [11], the same author extended this line of research considering what properties are observed by dialogue games of argumentation systems. Given the dialectical nature of these dialogue games, some similarities with the model developed in this work are to be expected. For instance, Prakken's *turn-taking function* serve the same purpose as our `toMove` function. Not withstanding, the main goal of the framework introduced in that article was to study the properties dialogue game posses, rather than characterizing the set of conclusion a given dialogue game might entail.

Finally, we too have pursued a similar line of research over a series of articles [23, 20, 24, 21], striving to define an abstract model for the agent interaction in multiagent systems. Even though an intermediate model of dialectical reasoning was also introduced there, that model never became the main focus of our attention, whereas now we intend to explore its potential as a tool for studying the essence of argumentation.

4 Conclusions

In this article we have developed the concept of *dialectical protocol*, a framework whose particular instantiations can serve as an alternative semantics for argumentation systems. To that end, we observed that most theories of defeasible argumentation admits a dialectical recast of its semantics. This suggests that a sufficiently generic model of the process of reasoning in dialectical terms could also serve as an abstract model of what happens inside an argumentative system. Applying this strategy, we first discussed the key aspects of dialectical reasoning in lay terms, and then proposed such generic model contemplating these aspects.

As a future work, we would like to explore modelling the semantics of several argumentation theories using concrete instantiations of dialectical protocols, in order to study the relation between their original semantics and the set of entailed claims of their corresponding dialectical protocols. Should they be proved equivalent, we believe that dialectical protocols may

also serve as a testbed suitable for exploring the different properties of argumentation systems, a topic being actively researched [22, 1, 2].

Acknowledgments

This work was partially supported by Agencia Nacional de Promoción Científica y Tecnológica (PAV 2003 Nro. 076, PICT 2002 Nro. 13096, PICT 2003 Nro. 15043), CONICET (*Consejo Nacional de Investigaciones Científicas y Técnicas de la República Argentina*), and CIC (*Comisión de Investigaciones Científicas de la Provincia de Buenos Aires*).

References

[1] Martín Caminada and Leila Amgoud. An Axiomatic Account of Formal Argumentation. In *Proceedings of AAAI 2005 Conference*, Pittsburgh, United States, 2005.

[2] Marcela Capobianco, Carlos I. Chesñevar, and Guillermo R. Simari. An approach for identifying design principles in argument systems—a preliminary report. In *Proceedings del 8vo Workshop de Investigadores en Ciencias de la Computación (WICC)*, pages 179–183, 2006.

[3] Phan M. Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning and Logic Programming and n-Person Games. *Artificial Intelligence*, 77(2):321–357, 1995.

[4] Alejandro J. García and Guillermo R. Simari. Defeasible Logic Programming: An Argumentative Approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.

[5] Ronald P. Loui. Process and Policy: Resource-Bounded Nondemonstrative Reasoning. *Computational Intelligence*, 14:1–38, 1998.

[6] John L. McCarthy. Circumscription—A Form of Non-Monotonic Reasoning. *Artificial Intelligence*, 13(1–2):27–39, 1980.

[7] Drew V. McDermott and Jon Doyle. Nonmonotonic logic I. *Artificial Intelligence*, 13(1–2):41–72, 1980.

[8] Robert C. Moore. Semantical considerations in non-monotonic reasoning. *Artificial Intelligence*, 25(1):75–94, 1985.

[9] John L. Pollock. Defeasible Reasoning. *Cognitive Science*, 11(4):481–518, 1987.

[10] John L. Pollock. *Cognitive Carpentry: A Blueprint for How to Build a Person*. The MIT Press, 1995.

[11] H. Prakken. Coherence and Flexibility in Dialogue Games for Argumentation. *Journal of Logic and Computation*, 15:1009–1040, 2005.

[12] Hendrik Prakken. Relating Protocols for Dynamic Dispute with Logics for Defeasible Argumentation. *Synthese*, 127(1–2):187–219, 2001.

[13] Hendrik Prakken and Giovanni Sartor. A System for Defeasible Argumentation, with Defeasible Priorities. In *Proceedings of the International Conference on Formal and Applied Practical Reasoning (FAPR)*, pages 510–524, 1996.

[14] Hendrik Prakken and Giovanni Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-classical Logics*, 7:25–75, 1997.

[15] Hendrik Prakken and Gerard Vreeswijk. Logics for defeasible argumentation. In *Handbook of Philosophical Logic*, volume 4, pages 218–319. Kluwer Academic Publishers, 2002.

[16] Raymond Reiter. A Logic for Default Reasoning. *Artificial Intelligence*, 13(1–2):81–132, 1980.

[17] Nicholas Rescher. *Dialectics: A Controversy-Oriented Approach to the Theory of Knowledge*. State University of New York Press, 1977.

[18] Guillermo R. Simari, Carlos I. Chesñevar, and Alejandro J. García. The Role of Dialectics in Defeasible Argumentation. In *Proceedings of the XIV Conferencia Internacional de la Sociedad Chilena para Ciencias de la Computación*, pages 111–121, 1994.

[19] Guillermo R. Simari and Ronald P. Loui. A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence*, 53(1–2):125–157, 1992.

[20] Alejandro G. Stankevicius. Modelling Multiagent Deliberation from an Abstract Standpoint. In *Proceedings del 3er Workshop de Investigadores en Ciencias de la Computación (WICC)*, pages 74–76, 2001.

[21] Alejandro G. Stankevicius. Un Modelo Dialéctico para la Deliberación Multiagente. Master's thesis, Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, 2004.

[22] Alejandro G. Stankevicius, Marcela Capobianco, and Carlos I. Chesñevar. Logical Properties in DeLP—a preliminary report. In *Proceedings del 4to Workshop de Investigadores en Ciencias de la Computación (WICC)*, pages 61–65, 2002.

[23] Alejandro G. Stankevicius and Guillermo R. Simari. A Framework for Multiagent Deliberation Based on Dialectical Argumentation. In *Proceedings del 6to Congreso Argentino de Ciencias de la Computación (CACiC)*, pages 1469–1480, 2000.

[24] Alejandro G. Stankevicius and Guillermo R. Simari. An Abstract Model for the Process of Deliberation within Multiagent Systems. In *Proceedings del 7mo Congreso Argentino de Ciencias de la Computación (CACiC)*, pages 1017–1026, 2001.

[25] Gerard A. W. Vreeswijk. Abstract Argumentation Systems. *Artificial Intelligence*, 90(1–2):225–279, 1997.