Dialogue game in defeasible logic

By

Subhasis Thakur

A thesis submitted to the University of Queensland
for the degree of Doctor of Philosophy
Department of ITEE
November 2008

Examiner's Copy

Typeset in $\LaTeX\,2_\varepsilon$.

Except where acknowledged in the customary manner, the material presented in this thesis is, to the best of my knowledge, original and has not been submitted in whole or part for a degree in any university.

Subhasis Thakur

List of Publications

- Subhasis Thakur, Guido Governatori, Vineet Padmanabhan, and Jenny Eriksson Lundstrm. Dialogue games in defeasible logic. In Mehmet A. Orgun and John Thornton, editors, 20th Australian Joint Conference on Artificial Intelligence, AI 2007, Gold Coast, 2-6 December 2007

- Jenny Eriksson-Lundstrom, Guido Governatori, Subhasis Thakur and Vineet Padmanabhan. An Asymmetric Protocol for Argumentation Games in Defeasible Logic. PRIMA-07, 2007

Abstract

Modeling interaction among artificial agents is a challenging task. Argumentation, auction, dialogue games and agent communication languages are proposed as mechanisms to model agent interaction. Among different mechanisms to model interaction among agents, dialogue game protocols is most significant. In this thesis I model a dialogue game in defeasible logic. I have two research goals. First, previous development of dialogue game protocols does not consider the strategic behaviour of agents. In this thesis I model the strategic behaviour of agents. Second, as more and more dialogue game protocols are developed there is a need to compare these protocols so that an appropriate protocol can be chosen for a given problem. Previous attempts to compare dialogue game protocols are based on functional comparison of components of a protocol. In this thesis I develop a new semantics for locution. Based on this semantics, dialogue game protocols can be compared so the protocol that can convey the right meaning can be identified.

Contents

List of Publications **v**

Abstract **vii**

List of Figures **xiii**

1 Introduction **1**

1.1 Background 1

1.1.1 Motivation 3

1.2 Problem statement 4

1.2.1 First problem 4

1.2.2 Second problem 4

1.3 Approach 4

1.3.1 Organization 5

2 Argumentation **7**

2.1 Argumentation framework 7

2.2 Related works in Argumentation 9

2.3 Conclusion 9

3 Defeasible logic and Argumentation **11**

3.1 Basic Defeasible Logic 11

3.2 Argumentation semantics of defeasible logic 14

3.3 Conclusion 18

4 Dialogue game protocol 19

4.1 Introduction 19

4.2 Components of a dialogue game 21

4.3 Logical properties of dialogue game 24

4.4 Desired properties of protocol 26

4.5 Dialogue game in artificial intelligence and law 28

4.6 Conclusion 29

5 Dialogue game in defeasible logic 31

5.1 Introduction 31

5.2 On dialogue games 33

5.3 Modeling dialogue games in defeasible logic 33

5.3.1 A protocol 34

5.3.2 Strengthening of rules 35

5.3.3 Transition rules 38

5.3.4 An example 39

5.4 Conclusion and future work 41

6 An asymmetric protocol for dialogue games 43

6.1 Introduction 43

6.2 Argumentation games 44

6.3 Dialogue games in defeasible logic - a symmetric protocol 45

6.4 An asymmetric protocol 47

6.5 Conclusion 55

7 A new semantics for locutions 57

7.1 Introduction 57

7.2 Dialogue game protocol 58

7.3 Dialogue tree 60

7.4 Agent's knowledge-base 65
7.5 Components of a dialogue game protocol 68
7.5.1 Locutions 71
7.5.2 Combination rules 73
7.6 Conclusion and future work 74

8 Future work **77**

References **79**

List of Figures

5.1 Strategic behaviour 37

7.1 Dialogue tree at step 1 61

7.2 Dialogue tree at step 2 61

7.3 Transition of dialogue trees 65

7.4 Knowledge-base 67

7.5 Game at step1 68

7.6 Game at step 3 69

7.7 Ambiguity in dialogue game 70

7.8 Locutions: Given a dialogue state different locutions can produce different states 74

7.9 Locutions 75

7.10 Combination rules 75

1

Introduction

1.1 Background

Interactions among agents are modeled as auctions, agent communication language, argumentation and dialogue games. Among these mechanisms, the dialogue game protocol is the most expressive and promising. A dialogue game is a process where multiple agents construct proofs for a proposition by taking turn in presenting arguments. A dialogue game protocol can be based on an argumentation framework, which is a declarative model of interaction. An argumentation framework describes the constructs as arguments, attacks on arguments by counter arguments and justification of an argument. Argumentation is more expressive than auction and agent communication language as it allows the challenging of an argument, and the presenting of a supportive argument so that an agent can have better understanding of another agent's internal structure. This improves the possibility of success of an interaction and may speed up the interaction. Dialogue game describes the ‘process’ of argumentation.

The dialogue game protocols are used in many agent interaction scenarios as follows:

1. Information seeking dialogue game: In this type of dialogue game, an agent seeks to answer some question [27].

2. Inquiry dialogue game: In this type of dialogue game, agents collaborate to answer some question [32].

3. Persuasion dialogue game: In this type of dialogue game, an agent tries to persuade another agent to accept an argument [28], [41], [44].

4. Negotiation dialogue game: In this type of dialogue game, agents argue about share and trade-off of some resources [21], [31], [48].

5. Deliberation dialogue game: In this type of dialogue game, agents argue about the course of actions to be undertaken by the agents [18].

6. Coalition formation dialogue game: In this type of dialogue game, agents argue for coalition formation [24].

[33] has structured dialogue game protocols into components as follows:

1. Commencement rules: Rules for deciding the topic of conversation.

2. Locutions: Legal moves available for dialogue game protocols. For example ASSERT{}, Justify{} etc.

3. Combination rules: These rules decide the set of locutions available at a particular step of the dialogue game.

4. Commitment rules: Commitment rules store the previous promises expressed in previous utterance by communicating agents.

5. Termination rules: Rules to indicate when to end a conversation.

[34] argued that, as more and more dialogue game protocols are developed there is a need to compare these protocols so that we can identify the appropriate protocol at a particular scenario. [36] has shown how to compare two protocols on the basis of functionality of the components of dialogue game protocols.

1.1.1 Motivation

First motivation: Lack of strategic behaviour in dialogue game protocols

Previous models of dialogue game protocols provide structures of protocols to be applied for different situations. But an agent's strategic behaviour in a dialogue game protocol is still not modeled. Agents enter the process of a dialogue game with a fixed knowledge base and a goal to win the game. As dialogue game protocols describe the turn-taking behaviour of agent's, winning the game not only depends on agents knowledge bases but also on the 'timing' and proper use of knowledge. Proper timing of an argument means the appropriate time of introduction of an argument into the process of the dialogue game. An agent's knowledge base will contain arguments those will support or oppose goals of the agent with respect to the current state of a dialogue game. The proper usage of knowledge base in a dialogue game protocol means efficient disclosure of knowledge, so that an agent can direct a conversation in a desired way.

Second motivation: Lack of semantics in dialogue game protocols

In previous attempts to compare protocols [36] only functional differences between protocols are discussed. But as dialogue protocols are designed separately from agent design, components of these protocols can impose some restrictions on agents as they have to obey obligation rules given in these protocols. These restrictions cause a difference between what an agent wants to convey and what it can convey using a dialogue game protocol. Thus protocols should be modeled on the basis of how efficient they are in conveying the right message.

1.2 Problem statement

1.2.1 First problem

First, we want to develop dialogue game protocols with certain properties which will extract the strategic behaviour of the agents. A protocol should give the players opportunities to react to opponent's arguments in a structured way, so that the protocol itself can extract strategic behaviour from how the agents play the game. This problem is divided into two subproblems. First, we consider the case in which all communicating agents are homogeneous. This means all agents have the same obligations to obey the rules given in a protocol. Second, we consider the case in which agents are heterogeneous so they have to maintain different levels of obligations.

1.2.2 Second problem

Second, we want to compare protocols in terms of how efficient they are to convey the right message. This problem creates another problem as components of dialogue game protocols particularly locutions are modeled according to speech act semantics given in FIPA ACL specification. This approach is not suitable for our purpose as we want to model how a protocol can convey the right 'meaning' of an utterance. Therefore, we have to develop a semantics for the components of protocols.

1.3 Approach

For the first problem, we develop two protocols of a dialogue game in defeasible logic. We use defeasible logic [22] as an argumentation framework. Argumentation semantics for defeasible logic is already captured in [5]. The linear complexity of defeasible logic will also help us in implementing the model. The first protocol [49] is developed for communication between homogeneous agents. [49] is based on the intuition that agents have the right to present arguments most favourable to their cases even if they contain some unknown misinterpretations. The second protocol [19] is an extension of the first protocol as we consider the case of communication between heterogeneous agents. In [19], we consider asymmetric priorities in a dispute between agents.

For the second problem, we develop a new semantics for locutions in a dialogue game protocol. We use communication theory to develop the new semantics for locutions. Next we compare protocols based on this new semantics.

1.3.1 Organization

This thesis is organized as follows. In Chapter 2 we provide a brief literature review regarding argumentation. In Chapter 3 we discuss about the role of defeasible logic in argumentation. In Chapter 4 we discuss previous works in dialogue game protocols. In Chapter 5 we present a dialogue game protocol for homogeneous agents. In Chapter 6 we present another dialogue game protocol for heterogeneous agents. In Chapter 7 we develop a new semantics for components according to our second problem. In Chapter 8 we conclude the thesis and discuss future works.

2

Argumentation

In this chapter we discuss previous developments in argumentation. In Section 2.1 we summarize the structure of arguments and in Section 2.2 we present related works in artificial intelligence and law on argumentation, where most of the proposals for dialogue games come from.

2.1 Argumentation framework

In this section we will summarize the argumentation framework developed in [11] and [51]. An argumentation framework contains arguments and attack relations between arguments.

Definition 2.1.1 *An argumentation framework is a pair $< Ar, Attack >$, where Ar is a set of arguments and $Attack$ is an attack relation between two arguments so that $Attack \subset A \times B | A, B \in Ar$. $Attack(A, B)$ means A attacks B.*

Definition 2.1.2 *A set of arguments S is said to be conflict free if there are no two arguments A and B in S such that there is a relation $Attack(A, B)$.*

Definition 2.1.3 *An argument $A \in Ar$ is acceptable with respect to a set of arguments S ($S \subset Ar$) if for each argument $B \in Ar$: $Attack(B, A)$ there is a argument $s \in S$: $Attack(s, B)$.*

Definition 2.1.4 *A conflict free set of arguments S ($S \subset Ar$) is said to be an admissible set of arguments if for each argument s ($s \in S$) is acceptable with respect to S.*

Definition 2.1.5 *A preferred extension of an argumentation framework (AF) is the maximal set of admissible arguments.*

Definition 2.1.6 *A conflict free set of arguments S is said to be the stable extension if S attacks any argument that does not belong to the set S.*

It is shown in [11] that, given an argumentation framework :

- Every stable extension is a preferred extension but not every preferred extension is a stable extension.
- Every admissible set is contained in a maximal admissible set of arguments.
- Stable extensions do not always exist.
- Preferred extensions always exist.
- Stable and preferred extensions are generally not unique.

Definition 2.1.7 *Fixed point semantics of an argumentation framework ($AF =< A, Attack >$) is described by a characteristic function as $F_{AF} : 2^A \rightarrow 2^A$*
$F_{AF}(S) = \{A|$ A is acceptable with respect to $S\}$

Grounded (Skeptical) semantics for argumentation framework is defined as :

Definition 2.1.8 *A grounded extension of an argumentation framework $AF =< A, Attack >$ is the least fixed point of F_{AF}.*

[51] has developed credulous and skeptical semantics for argument games. Given an argumentation framework $AF =< A, Attack >$ and an argument $a(a \in A)$ the credulous argument game is about constructing at least one admissible set of arguments which contains a. A skeptical argument is an argument game showing that all admissible sets of arguments in AF contain a.

2.2 Related works in Argumentation

Substantial work has been done on argumentation games in AI and law. [23] presents an early specification and implementation of an argumentation game based on the Toulmin argument-schema without a specified underlying logic. [13] presented The Pleadings Game as a normative formalization and fully implemented computational model, using conditional entailment. The goal of the model was to identify issues in the argumentation rather than elaborating on the status of the main claim. The dialectic proof procedures presented by [12] focus on minimizing the culprit of argumentation. The proof procedures are expressed as metalogic programs. DiaLaw [15] is a two player game, in which both players make argument moves. The model combines the exchange of statements and exchange of arguments, dealing with rhetorical as well as psychological issues of argumentation. However, the main focus for the two players is to convince each other rather than defeat an adversary as in our case. The abstract argumentation systems of [17, 52] study arguments as the object of defeat. The results, however, are more related to stable semantics than skeptical as in the defeasible logic utilized in our framework and the study is devised as meta games for changing the rules of argumentation games.

2.3 Conclusion

In this chapter we have summarized the previous works in argumentation. An argumentation framework is the basis of a dialogue game. A dialogue game describes the process of constructing valid arguments with respect to an argumentation framework. In the next chapter, we discuss defeasible logic semantics of argumentation. We will use defeasible logic to represent the agents and their arguments and this representation will be used to model a dialogue game.

3

Defeasible logic and Argumentation

In this chapter we discuss defeasible logic and argumentation semantics of defeasible logic.

3.1 Basic Defeasible Logic

Over the years defeasible logic [22, 38, 53] proved to be a simple, flexible, rule based non-monotonic formalism able to capture different and sometimes incompatible facets of non-monotonic reasoning [9], and efficient and powerful implementations have been proposed [8, 30].

Knowledge in defeasible logic can be represented in two ways: facts and rules.

Facts are indisputable statements, represented either in form of states of affairs (literals). Facts are represented by predicates. For example, “Tweety is a penguin” is represented by $Penguin(Tweety)$.

A *rule*, on the other hand, describes the relationship between a set of literals (premises) and a literal (conclusion), and we can specify how strong the relationship is. Rules allow us to derive new conclusions given a set of premises. We distinguish between *strict rules*, *defeasible rules* and *defeaters* represented, respectively, by expressions of the form $A_1, \ldots, A_n \rightarrow B$, $A_1, \ldots, A_n \Rightarrow B$ and $A_1, \ldots, A_n \rightsquigarrow B$, where $A_1, \ldots, A_n$ is a possibly empty set of prerequisites (causes) and B is the conclusion (effect) of the rule. We only consider rules that are essentially propositional. Rules containing free variables are interpreted as the set of their ground instances.

Strict rules are rules in the classical sense: whenever the premises are indisputable then so is the conclusion. Thus they can be used for definitional clauses. An example of a strict rule is “Penguins are birds”, formally: $\mathit{Penguin}(X) \rightarrow \mathit{Bird}(X)$.

Defeasible rules are rules that can be defeated by contrary evidence. An example of such a rule is “Birds usually fly”: $\mathit{Bird}(X) \Rightarrow \mathit{Fly}(X)$. The idea is that if we know that X is a bird, then we may conclude that X can fly *unless there is other evidence suggesting that she may not fly.*

Defeaters are a special kind of rules. They are used to prevent conclusions, not to support them. For example: $\mathit{Heavy}(X) \rightsquigarrow \neg \mathit{Fly}(X)$. This rule states that if something is heavy then it might not fly. This rule can prevent the derivation of a “fly” conclusion. On the other hand it cannot be used to support a “not fly” conclusion.

Defeasible logic is a “skeptical” non-monotonic logic, meaning that it does not support contradictory conclusions. Instead, defeasible logic seeks to resolve conflicts. In cases where there is some support for concluding A but also support for concluding $\neg A$, defeasible logic does not conclude either of them (thus the name “skeptical”). If the support for A has priority over the support for $\neg A$ then A is concluded. No conclusion can be drawn from conflicting rules in Defeasible logic unless these rules are prioritized. The *superiority relation* among rules is used to define priorities among rules, that is, where one rule may override the conclusion of another rule. For example, given the defeasible rules

$$r : \mathit{Bird}(X) \Rightarrow \mathit{Fly}(X) \qquad r' : \mathit{Penguin}(X) \Rightarrow \neg \mathit{Fly}(X)$$

which contradict one another, no conclusive decision can be made about whether a Tweety can fly or not. But if we introduce a superiority relation $\succ$ with $r' \succ r$, then we can indeed conclude that Tweety cannot fly since it is a penguin.

We now give a short informal presentation of how conclusions are drawn in defeasible logic. Let D be a theory in defeasible logic (i.e., a collection of facts, rules and a superiority relation). A *conclusion* of D is a tagged literal and can have one of the following four forms:

$+\Delta q$ meaning that q is definitely provable in D (i.e., using only facts and strict rules).

$-\Delta q$ meaning that we have proved that q is not definitely provable in D.

$+\partial q$ meaning that q is defeasibly provable in D.

$-\partial q$ meaning that we have proved that q is not defeasibly provable in D.

Strict derivations are obtained by forward chaining of strict rules, while a defeasible conclusion p can be derived if there is a rule whose conclusion is p, whose prerequisites (antecedent) have either already been proven or given in the case at hand (*i.e.*, facts), and any stronger rule whose conclusion is $\neg p$ has prerequisites that fail to be derived. In other words, a conclusion p is derivable when:

- p is a fact; or
- there is an applicable strict or defeasible rule for p, and either
  - all the rules for $\neg p$ are discarded (i.e., are proved to be not applicable) or
  - every applicable rule for $\neg p$ is weaker than an applicable strict[1] or defeasible rule for p.

Formally a Defeasible Logic (as formalized by [10]) theory is a structure $D = (F, R, \succ)$ where F is a finite set of factual premises, R a finite set of rules, and $\succ$ a superiority relation on R. Given a set R of rules, we denote the set of all strict rules in R by R_s, the set of strict and defeasible rules in R by R_{sd}, the set of defeasible rules in R by R_d, and the set of defeaters in R by R_{dft}. $R[q]$ denotes the set of rules in R with consequent q. In the following $\sim p$ denotes the complement of p, that is, $\sim p$ is $\neg q$ if $p = q$, and $\sim p$ is q if p is $\neg q$. For a rule

[1] Note that a strict rule can be defeated only when its antecedent is defeasibly provable.

r we will use $A(r)$ to indicate the body or antecedent of the rule and $C(r)$ for the head or consequent of the rule. A rule r consists of its antecedent $A(r)$ (written on the left; $A(r)$ may be omitted if it is the empty set) which is a finite set of literals, an arrow, and its consequent $C(r)$ which is a literal.

Provability is based on the concept of a derivation (or proof) in D. A derivation is a finite sequence $P = (P(1), \ldots, P(n))$ of tagged literals. Each tagged literal satisfies some proof conditions. A proof condition corresponds to the inference rules corresponding to one of the four kinds of conclusions we have mentioned above. $P(1..i)$ denotes the initial part of the sequence P of length i. Here we state the conditions for strictly and defeasibly derivable conclusions (see [22] for the full presentation of the logic):

If $P(i+1) = +\Delta q$ then

(1) $q \in F$, or

(2) $r \in R_s[q]$, $\forall a \in A(r) : +\Delta a \in P(1..i)$.

If $P(i+1) = +\partial q$ then

(1) $+\Delta q \in P(1..i)$, or

(2) (2.1) $\exists r \in R_{sd}[q] \forall a \in A(r) : +\partial a \in P(1..i)$ and

(2.2) $-\Delta \sim q \in P(1..i)$ and

(2.3) $\forall s \in R[\sim q]$ either

(2.3.1) $\exists a \in A(s) : -\partial a \in P(1..i)$ or

(2.3.2) $\exists t \in R_{sd}[q]\ \forall a \in A(t) : +\partial a \in P(1..i)$ and $t > s$.

3.2 Argumentation semantics of defeasible logic

An argumentation system consists of a logical language (defeasible logic), definition of argument, conflict between arguments and status of an argument. The following definitions are used for the explanation of argumentation semantics in defeasible logic :

Definition 3.2.1 *An argument for a literal P based on a set of rules is a tree, where the root is P and nodes of the tree are labeled with literals such that for a label H the following are true :*

- *If B is a child node of H then there is a rule in R such that H is the head of the rule and B is the body of the rule.*
- *If the above rule is a defeater then H is the root of the argument.*
- *The arcs in a proof tree are labelled by rules used to obtain the proof.*
- Given a theory in defeasible logic D, the set of arguments that can be generated from D is represented by $Args_D$.
- A supportive argument is a finite argument which has no defeater.
- A strict argument is an argument where no defeasible rules are applied. Otherwise the argument is a defeasible argument.
- An argument A is supported by a set of arguments S if every proper subargument of A is in S.

Definition 3.2.2 *Attack relations among arguments: An argument A attacks a defeasible argument B if the conclusion of A is the opposite of the conclusion of B and conclusion of B is not a part of a strict subtree. A set of arguments S attacks an argument A if there is an argument B in S such that B attacks A.*

Definition 3.2.3 *Undercut attack: A defeasible argument A is undercut by a set of arguments if S supports an argument B which attacks a proper non-strict subargument of A.*

The undercut definition presented here [5] is slightly different from Pollock's theory of defeasible logic. In [40] the undercut attack is defined as 'we can think of it as giving us a reason for believing that (under the present circumstances) the truth of the premises does not guarantee the truth of the conclusion'. So if an argument A (and a subargument B, which acts as the premise of A) is undercut by an argument S it can be interpreted as S preventing the implication of A from B even if B is true. Here the conclusion of S is not necessarily the opposite of the conclusion of B. But according to the defeasible logic semantics we follow here, the conclusion of S is necessarily opposite to that of B (or sub arguments of B). Again, 'the truth of the premises does not guarantee the truth of the conclusion' is an inherent property of the defeasible type [22] of rules ($\Rightarrow$) and '... believing that (under the present circumstances) ... ' suggests that 'the truth of the premises' does not hold but the inference relation 'if B then it may be A' is still valid but not applicable.

Definition 3.2.4 *Status of an argument: The notion of acceptance of an argument A with respect to a set of arguments S is as follows: if we accept S is valid then we have to accept A. Using this intuition justified the set of arguments J_i^D in a theory D defined by the following recursive construction :*

$$J_o^D = \emptyset$$
$$J_{i+1}^D = \{a \in args_d | a \textit{ is acceptable with respect to } J_i^D\}$$

The set of justified arguments in a defeasible theory D is represented as $Jargs^D = \{\cup_{i=0}^{\infty} J_i^D\}$. A literal p is justified if it is the conclusion of a supportive argument in $Jargs^D$.

Defeasible logic can be tuned to describe nonmonotonic phenomenon as ambiguity blocking and ambiguity propagation. Defeasible logic has an ambiguity blocking nature but it can be modified to show ambiguity propagation nature.

Definition 3.2.5 *Grounded semantics and ambiguity propagation: In the case of ambiguity propagation[2], the acceptability is defined as follows. An argument A is acceptable with respect to a set of arguments S if A is finite and*

- *A is strict, or*
- *Every argument attacking A is attacked by S.*

The rejection of an argument for ambiguity propagation is defined as follows: Given S and T as sets of verified and rejected arguments, an argument A is rejected by sets of arguments S and T if A is not strict and either of the following is true :

[2] Ambiguity propagation semantics for defeasible logic has a stronger provability condition for $\pm\partial$ proofs. In this case, to prove a literal p we require that a rule with conclusion $\sim p$ is not applicable because its premises can not be supported. See [5] for details.

- one proper subargument of A is in S
- A is attacked by a finite argument in T.

Provability of defeasible logic is applied for ambiguity propagation as follows: Let D be a defeasible theory, p is a literal and T be a set of arguments

- $D \vdash +\partial_{ap} p$ iff p is justified.
- $D \vdash -\partial_{ap} p$ iff p is rejected with respect to T.

Defeasible semantics and ambiguity blocking: In the case of ambiguity blocking[3] the acceptability is defined as follows. An argument A is acceptable with respect to a set of arguments S if A is finite and :

- A is strict or
- Every argument attacking A is undercut by S.

The rejection of an argument for ambiguity blocking is defined as follows. An argument A is rejected by sets of arguments S and T if a is not strict and either of the following is true :

- One proper subargument of A is in S
- A is attacked by an argument supported by T.

Provability of defeasible logic is applied for ambiguity blocking as follows. Let D be a defeasible theory, p is a literal :

- $D \vdash +\partial_{ap} p$ iff p is justified.
- $D \vdash -\partial_{ap} p$ iff p is rejected $JArgs^D$.

[3] Semantics for ambiguity blocking defeasible logic does not allow contradictory conclusions. The semantics [22] is the same as the defeasible logic described in the previous section

3.3 Conclusion

In this chapter we gave an overview of argumentation semantics for defeasible logic. We will use this semantics to develop dialogue games and the tree representation of arguments is further extended in Chapter 7, where we will sketch the semantics for the 'meaning' of the speech acts. In the next chapter we discuss current developments in dialogue games.

4

Dialogue game protocol

4.1 Introduction

Dialogue game protocols are developed to model agent interaction. In a dialogue game there is a specific topic and the agents take turns to present their arguments. A dialogue game protocol imposes some rules upon the communicating agents so that the presented arguments are valid and the communicating agents have a fair chance to win the game. There are different types of dialogue game as shown in Chapter 1:

1. Information seeking dialogue game [27].

2. Inquiry dialogue game: In this type of dialogue game, agents collaborate to answer some question [32].

3. Persuasion dialogue game: In this type of dialogue game, an agent tries to persuade another agent to accept an argument [28], [41], [44].

4. Negotiation dialogue game: In this type of dialogue game, agents argue about share and trade-off some resources [21], [31], [48].

5. Deliberation dialogue game: In this type of dialogue game, agents argue about the course of actions to be undertaken by the agents [18].

6. Coalition formation dialogue game: In this type of dialogue game, agents argue for coalition formation [24].

A general semantics of a dialogue game protocol is as follows:

1. There is a specific topic in a dialogue game.

2. Agents takes turns to present an argument.

3. Arguments must support or oppose the topic of dialogue.

4. Arguments presented at any stage of the game (except the first step) must oppose arguments of previous step(s).

5. An agent can not repeat an argument[1].

6. An agent must not contradict its own argument (what it has said before).

[1] In most of the previous works in dialogue game protocols repetition of a dialogue is not allowed. This is because of the following reason :

- Once a dialogue is presented it will be either accepted or rejected by the other player. In the case where the dialogue is accepted, there is no need to repeat it as the rule representing the dialogue remains and in a case where the dialogue is rejected, if it is again repeated then it will be again defeated by the counter argument of the other player, who has to repeat its counter argument again. It will definitely slow the process to reach a result through argumentation and even prevent any conclusion being reached at all.

7. An agent can retract what it has said before (a persuasion dialogue game may not allow this).

8. If an agent runs out of moves then the other player wins the game.

In this chapter we discuss (1) classification of a dialogue game into components, (2) logical properties of a dialogue game, (3) desired properties of a dialogue game and (4) related works in AI and law about argumentation.

4.2 Components of a dialogue game

Different dialogue protocols have been proposed for the different types of dialogues mentioned in the previous section. As classified by [33], all these protocols share some basic components of dialogues such as

1. Commencement rules: Commencement rules initiate a dialogue game. These rules decide the topic of a dialogue game. Agents can be involved in some interactions to decide the topic of a conversation.

2. Locutions: Locutions are move labels in a dialogue game. A dialogue game protocol can have a finite number of locutions. Examples of locutions are Assert, Justify and Propose. Different dialogue game protocols can have the locutions with the same functionality but with different names.

3. Combination rules: Combination rules indicate which locutions are permitted at a particular stage of the protocol.

4. Commitments: Rules that define circumstances under which participants express commitments to propositions.

5. Termination rules: Circumstances under which the dialogue game ends.

Among these components, locutions are the most important. Other components are the same for different protocols. [35] provides a semantics of locution, which is similar to

FIPA [4] speech act semantics. Semantics of a speech act is described by (1) feasibility preconditions and (2) rational effect. Feasibility preconditions are conditions which must be satisfied before an agent can use a speech act. Rational effects are conditions that will arise once an agent utters a speech act. For example: $Inform_{a,b}\{M\}$ (Agent a wants to INFORM agent b that it is M) has the following semantics :

- Feasibility preconditions: $B_a(M) \wedge B_a(B_b(M)) \vee B_a(B_b(\neg M))$. This means agent a believes that it is M and it believes that agent b does not have any opinion about M.
- Rational effects: The rational effect of $Inform_{a,b}\{M\}$ is $B_b(M)$ which reads as B believes that it is M after a informs b that it is M.

[35] has followed the following syntax for a locution :

$illocution(P_i, \chi)$ or $illocution(P_i, P_j, \chi)$

where P_i and P_j are identifiers or names of the agent who is uttering the locution and χ is the content of the locution. [35] has developed syntax for 5 principal locutions as follows :

F1: $Assert(P_i, \chi)$: Speaker P_i asserts that it is χ. By uttering this locution P_i has an obligation to justify if asked by an opponent.

F2: $Question(P_i, P_j, \chi)$: Speaker P_i questions a previous utterance $Assert(P_j, \chi)$ of P_j, so P_j has to justify its previous utterance. P_i does not have any obligation to use the *Question* locution.

F3: $Challenge(P_i, P_j, \chi)$: Speaker P_i challenges P_j's previous utterance as $Assert(P_j, \chi)$, so P_i has to justify its previous argument. Locution *Challenge* imposes an obligation on P_i that it has to justify $\neg\chi$ when asked by an opponent.

F4: $Justify(P_j, Th \vdash^{+} \chi)$: By uttering $Justify$ locution, agent P_j provides supportive argument Th to its previous argument $Assert(P_j, \chi)$. $Justify(P_j, Th \vdash^{-} \chi)$ shows p_j's justification Th against χ (which is asserted by another agent).

F5: $Retract(P_i, \chi)$: Agent P_i withdraws its previous argument $Assert(P_i, \chi)$. By withdrawing the previously asserted statement P_i removes all obligations regarding χ.

The combination rules for the above locutions are as follows :

[35] has given semantics of locution in accord with FIPA ACL. This semantics has three components: (1) preconditions; (2) post conditions; (3) dialectical obligations. The semantics of the locutions are as follows:

$Assert(P_i, \chi)$

- Precondition: Speaker P_i desires another agent P_j to believe that P_i believes χ.
- Post Condition: Opponent agent believes that P_i desires that the opponent to believe that P_i believes χ.
- Dialectical obligations: P_i may have to justify χ.

$Question(P_j, P_i, \chi)$

- Precondition: P_i has an obligation to justify χ and P_i desires that P_i believes that P_j desires that P_i utters $Justify$ locution in support of χ.
- Post condition: P_i must utter a justify locution.
- Dialectical obligations: No effect.

$Challenge(P_j, P_i, \chi)$

- Precondition: P_i has the obligation to justify χ and P_j desires that P_i believes that P_j desires P_j to utter $Justify$ locution and P_j does not believe χ.
- Post condition: P_i must utter $Justify$ in support of χ.
- Dialectical obligations: P_j has obligations to justify $\neg\chi$.

$Justify(P_j, Th \vdash^{+} \chi)$

- Precondition: P_j has an obligation to justify χ and P_i has uttered either $Challenge(P_i, P_j, \chi)$ or $Question(P_i, P_j, \chi)$ and P_j desires that P_i believes that Th is an argument for χ.
- Post condition: Opponent(P_i) believe that P_j desires that P_i believes that Th is an argument for χ.

- Dialectical obligations: P_j has the obligation to justify Th.

Retract: $Retract(P_i, \chi)$

- Precondition: P_i desires that opponent believes that P_i no longer beliefs χ
- Post Condition: P_j beliefs that P_i desires that P_j believes that P_i no longer believes χ
- Dialectical obligations: Successive obligations for *Assert* or *Challenge* made on χ are no longer valid.

4.3 Logical properties of dialogue game

Every dialogue game protocol has some logical properties such as :

- Termination rule
- Computational complexity
- Automatability
- Soundness
- Fairness

Termination rules: 'Termination' of a dialogue game is the circumstances where the game ends.

[39] provides termination conditions for information seeking dialogues, enquiry dialogues and persuasion dialogues. In an information seeking protocol given a '$question(P)$', an agent can '$assert(p)$' or '$assert(\neg p)$' '$assert(\emptyset)$' (this means the agent has no opinion about p). A confident agent presents an *assert* proposition if it has an argument for it and a thoughtful agent asserts only if it has an acceptable argument. A credulous agent accepts a proposition p if there is an argument for p. A cautious agent accepts p if it is unable to construct an argument for $\neg p$. A skeptical agent accepts p if there is an acceptable argument for p. [39] assures that

> An information seeking dialogue under protocol IS between a credulous, cautious or skeptical agent G and a confident or thoughtful agent H will always terminate.

The explanation for a 'definite' termination is as follows:

1. If the response of H is '$assert(\emptyset)$' then the game terminates.

2. If G is credulous, then it will accept p (or $\neg p$).

3. If G is cautious, then G will either accept p or has arguments for $\neg p$. In the latter case it will challenge p and will receive a response (an argument S for p). If G does not has an argument against $s \in S$ then s will be accepted but still p will not be accepted. As G can utter only '$challenge(p)$' the dialogue will terminate. If G has an argument against s, then it will challenge those arguments and will generate a response as $\{\{s\},s\}$ from H, which G can not accept and also can not challenge as it will repeat $challenge(s)$. So the game will terminate.

4. If G is skeptical, then it will accept only 'acceptable' arguments from H, which includes further arguments on subarguments for '$assert(p)$'.

[39] follows a protocol for enquiry dialogue as if agent B enquiries about some proposition p then agent A will answer as $q \rightarrow p$. Agent B can accept it of challenge it. In later case, A will provide argument for its assertion, which can also be challenged by B. [39] asserts that

> An enquiry dialogue I between agents G and H with any acceptance or assertion will terminate.

The explanation for this termination is that enquiry dialogues can be converted into information seeking dialogues (with roles of the agents changing) and as information seeking dialogues always terminate, the enquiry dialogue will always terminate. [39] presents a persuasion dialogue game protocol which allows an agent to '$assert(p)$' and in response, the opposition agent can challenge p as $assert(\neg p)$ or $challenge(p)$. In the first case, the roles of the agents are reversed. In the second case, the first agent provides an argument for '$assert(p)$', which can be accepted or challenged by the opposition agent. [39] assures that this persuasion protocol always terminates. The explanation for termination is :

1. Persuasion dialogue protocol (P) is the same as the information seeking protocol (IS) and as IS terminates, P will also terminate.

2. The only exception is that $Assert(\neg p)$ is allowed in P. This could lead to a non-termination situation if another agent presents $Asset(p)$. But as the agents can not repeat the same dialogue the game will terminate.

[50] describes termination circumstances for negotiation dialogues in abductive logic. In persuasion dialogues, termination is a circumstance where an agent runs out of moves as described in [49], [19].

Complexity: [25] discusses computational complexity of dialogue games. [25] describes the complexity problem of dialogue as :

1. when the given argument can be defended, how many rounds could it take to prove to a challenging party that the argument may be defended against any attack?

2. when the given argument cannot be defended against all possible attacks, how many rounds must it take to convince putative defenders that their position is untenable?

Soundness: [45] describes soundness for persuasion dialogues as:

> ... if the proponent wins a dispute, its initial argument is defeasibly provable on the basis of what has been said in the dispute?

Fairness or Completeness: [45] defines fairness for persuasion dialogue as:

> Is a given protocol for dispute fair (or complete), in the sense that if a certain argument becomes defeasibly provable on the basis of what has been said in the dispute, the proponent (opponent) can win any continuation of the dispute in which no new information is introduced?

4.4 Desired properties of protocol

[37] made an attempt to identify the desired properties of a dialogue game protocol to be efficient as an agent interaction protocol. [37] has identified 13 properties as follows:

1. Stated dialogue purpose: A dialectical system must have a stated purpose which is publicly announced. For example, in negotiation dialogues the share of resources must be publicly announced. This will help agents to participate in a dialogue game.

2. Diversity of individual purpose: A dialectical system must allow agents to follow their own purpose consistent with the overall purpose of the game.

3. Inclusiveness: A dialectical system must not preclude participation of agents.

4. Transparency: Participants should know the rules of the game in advance.

5. Fairness: All participants should be treated equally otherwise any asymmetries should be made explicit.

6. Clarity of argumentation theory: A dialectical system should conform to an argumentation theory. This will ensure arguments and counter arguments are syntactically correct.

7. Separation between syntax and semantics: This separation will allow the usage of the same protocol syntax for different semantics.

8. Rule consistency: The locutions and game rules should be consistent.

9. Encouragement of resolution: Resolution to each dialogues should be facilitated.

10. Discouragement of disruption: Disruptive behaviour of participants should be discarded.

11. Enablement of self-transformation: The dialectical system should allow agents to change their preferences during the course of the game.

12. System simplicity.

13. Computational simplicity.

4.5 Dialogue game in artificial intelligence and law

Inference System (IS) was proposed in [41] to capture dialogue games. A theory in IS is represented by $T_{IS} = (R, \leq)$ where R a is set of strict and defeasible rules and $\leq$ is a partial preorder which resolves any conflicts on precedence between rules. Arguments are *justified*, *overruled* and *defensible* depending on the outcome of the dialogue game. [42] describes the burdens associated with IS. According to [42], there are three kinds of burdens, namely, (1) *burden of persuasion* (2) *burden of production* and (3) *tactical burden of proof.* The burden of persuasion determines which party will be losing if the evidence is balanced. Burden of production concerns on whom, at any time of the game, the burden of presenting evidence is placed. Tactical burden of proof concerns a party's assessment of the risk of losing a game. In [41], players have fixed roles as the burden of prosecution lies on the proponent, leaving the opponent with the burden to *interfere*. [44] modified IS as it proposes switching of roles in a *Litigation Inference System* (LIS). A theory in LIS is represented as $T_{LIS} = (R, \leq, b_\pi, b_\delta)$ where $(R, \leq)$ is an IS theory. b_π, b_δ are burdens of prosecution for proponent and opponent respectively. [46] modifies LIS and proposes an *Augmented Litigation Inference System* (ALIS) which generates the content of b_π, b_δ as a result of an argument-based reasoning. A theory in ALIS is represented as $T_{ALIS} = (R, \leq)$, where $(R, \leq)$ is the IS theory described by a language which has a predicate *burden*. $burden(p, l)$ means that on the player p is placed the burden of prosecution for the literal l. A dialogue move m has three components: (1) *pl(m)* the player who made the move; (2) *r(m)* the role of the player; and (3) *a(m)* the argument put forward in the move. ALIS imposes a protocol to be followed by the players. The protocol in ALIS differs from the protocol proposed in LIS in the sense that (1) if in the adjacent previous step the opponent *weakly defeated* an argument proposed by the proponent, then in the current step the proponent can argue that the opponent now has a burden on that literal; and (2) if the players weakly defeat each other while in their opponent role, then the *plaintiff* can argue that the *defendant* has the burden of proof.

Modelling dialogue games in defeasible logic has been addressed by [14, 26, 29]. [14] focuses on persuasion dialogues and it includes the cognitive mental states of agents such as knowledge and belief. In addition it presents some protocols for some types of dialogues (e.g. information seeking, explanation, persuasion). The main reasoning mechanism is based on basic defeasible logic (see Section 3.1) and it ignores recent developments in extensions of defeasible logic with modal and epistemic operators for representing the cognitive states of agents [6, 7], and it does not cover adversarial dialogues. [26] provides an extension of defeasible logic to include the step of the dialogue. A main difference is that the resulting mechanism just defines a metaprogram for an alternative computation algorithm for ambiguity propagating defeasible logic while the logic presented here is ambiguity blocking. In [29], the authors focus on rule scepticism and propose the use of a sequences of defeasible (meta) theories, and use meta-reasoning (meta-rules or high level rules) to assess the strength of rules for the theories at lower levels.

4.6 Conclusion

We want to model a dialogue game protocol to extract the strategic behaviour of agents. We are inspired by [26] as we have separated the knowledge of the players into (1) private knowledge and (2) public knowledge. The common public knowledge forms the common set of arguments, which is a theory in defeasible logic. In the next chapter we describe our dialogue game protocol for homogeneous agents.

5

Dialogue game in defeasible logic

5.1 Introduction

Agents interact with other agents. Among two agents, the nature of their interactions can be of various kinds. Here we consider two types of interaction: cooperative and adversarial. In a cooperative situation the agents exchange information with the aim of reaching a common goal, while in an adversarial scenario the goals of the parties are conflicting. However, this does not imply a clear-cut dissimilarity between the two types of interaction. Conflicting sub-goals often are found amongst agents in a cooperative setting, while in an adversarial discussion one agent may partially accept a proposal of her adversary as it provides a stronger justification of her case.

These kinds of interactions are part of the broader field of argumentation, and formal argumentation is the branch using logic and (formal methods in general) to model it. Over the past few years a line of research emerged for the representation of these type of arguments: dialogue games. Dialogue games have proven to be extremely useful for modelling some forms of legal reasoning. In this chapter we focus on one form of dialogue games, the adversarial, where the two parties debate one topic.

Most formal models of dialogues provide computational and procedural representations of some real-life domain (e.g., legal reasoning). Dialogue games are by their own nature defeasible, meaning that arguments put forward by one of the agents in support of a conclusion can be defeated by contrary evidence put forward by the other agent. Accordingly, standard model-theoretic semantics is not appropriate for this kind of reasoning. Dung [11] proposed argumentation semantics to obviate this issue. The main idea of argumentation semantics is that the main objects we evaluate are "arguments"[1]. Various relationships (e.g. attack, rebut and defeat) between arguments are defined by the semantics, and the relationships are extended to sets of arguments. The key notion for a set of arguments is the notion of *support*, that is whether a set of arguments is self-consistent and provides the base to derive a conclusion; in other words, if it is possible to prove the conclusion from the rules, facts and assumptions in the set of supporting arguments. A conclusion is justified, and thus provable, if there is a set of supporting arguments and all counterarguments are deficient when we consider the arguments in the set of supporting arguments. Various argumentation semantics have been proposed to capture different relationships between supporting and opposing set of arguments. However, in general some forms of argumentation semantics are able to characterize dialogue games [45].

Defeasible logic [22, 38] is an efficient non-monotonic formalism that encompasses many logics proposed for legal reasoning. Defeasible Logic can be characterized in terms of argumentation semantics [5], thus the correspondence between defeasible logic on one side and dialogue games on the other follows implicitly from their common semantics. The aim of this chapter is to propose a direct mapping between dialogue games and defeasible logic, and to show that defeasible logic offers a general, powerful and computationally efficient framework

[1] In the abstract formulation of the argumentation semantics 'arguments' are left unspecified, however, in the majority of concrete instances of the argumentation framework, arguments are defined as a chain of reasoning based on facts or assumptions and rules captured in some formal language or logic.

to model and to extend dialogue games.

The chapter is organized as follows: in Section 5.2 we outline the basic ideas of dialogue games and in Section 5.3 we show how to adapt defeasible logic to model dialogue games.

5.2 On dialogue games

We consider dialogue games as a game where we have two players called the *Proponent* and the *Opponent*. Each player is equipped with a set of arguments, a subset of which the players *move*, i.e. take turns in putting forward. The aim of the game is to justify a particular conclusion while adhering to the particular protocol scheme governing the game. A basic protocol for the admissible moves by the players would be, for the proponent, that the current move attacks the previous move of the opponent, and that the main claim (the content of the dispute) follows from the arguments assessed as currently valid. For the opponent, we have that the arguments of the move attack the previous move, and the main claim is not derivable. Even though more complex winning conditions are possible, by a basic protocol a player wins the dialogue game when the other party is out of admissible moves.

5.3 Modeling dialogue games in defeasible logic

During the dialogue game the agents take turns in presenting their arguments (rules). The literal the agents are trying to prove or disprove will be called the critical literal. The intuition behind the notion of critical literal (cl) is as follows: the cl is the central issue of the confrontation between two agents. Agents may have confrontation about other literals and arguments from which cl can be inferred. Arguments presented by an agent may not include cl (except the first step of a dialogue game) but it must alter the proof of the cl with respect to its proof in the previous step of the dialogue game. Each agent has knowledge that initially is known only to the agent (private knowledge). Initially all the arguments are private. In addition, both agents have access to a set of common knowledge. By putting forward arguments from the private knowledge of an agent, these arguments become part of the set of common knowledge. We assume that all arguments (rules) are defeasible, acknowledging an agent's right to put forward interpretations of assumptions, fact and evidence in the way most favorable for his case (cf. [29]). The set of common arguments is continuously updated at each step and defeated defeasible rules are removed at each step. At any time the set of common arguments contains defeasible rules only from the current step t_i and adjacent previous step t_{i-1} and facts, strict rules from previous steps. The theory of common set of arguments is $T_{common} = (F, R, >)$, where F is the set of facts, R is the set of rules, $>$ is the superiority relationship among the rules. At each step the proof procedure is applied on the critical literal. The nature of the game determines the burden as well as the winner of the game. A party wins the game if the proof is $+\Delta A$ (A is the critical literal) at any stage of the game. If a party at any stage of the game proves $+\partial A$ (A is the critical literal) the other party has the burden to produce proof of $-\partial A$, or $-\Delta A$ or $+\Delta\neg A$ or $+\partial\neg A$. Our notion of burden is restricted only to the critical literal unlike the notion of burden in [41]. The condition that an agent cannot repeat its rule is unnecessary in our model. If a rule $C(r)$ has been put forward and successfully defeated any counter arguments supporting the opposite conclusion $\neg C(s)$, the rule $C(r)$ is added to the common set of knowledge. In accordance to our protocol, the rule $C(r)$ will effectively prevent the opponent from putting forward the defeated arguments (rules) $\neg C(r)$ into the dialogue as we require all admissible arguments to be defeating opposing arguments presented at a previous step. In addition this criterion guarantees that the dialogue game terminates, since we assume the private and public set of arguments are finite.

5.3.1 A protocol

Mainly we adhere to the protocol of a dialogue game captured in [41, 42, 44, 46]. Thus, the rules for our dialogue games are as follows:

1. The parties cannot present arguments in parallel. Thus, the parties take turns in presenting their arguments.

2. As we allow for each agent to put forward as argument the interpretations of rules and evidence in the way most favorable for his case, all arguments presented by an

agent are initially treated as defeasible rules. If in the next step the other party fails to provide valid counter arguments, these defeasible rules will be upgraded to strict rules.

3. The arguments in support of a critical literal $\neg A$ presented by a party 1 at any step must attack (at least) the conclusion in support of the critical literal A put forward by the other party 2 in the previous step. Moreover, in order to prevent a strengthening of the defeasible rules in support of A in the set of common arguments and to remove from the set of common arguments all defeasible rules which have as conclusion A, party 1 must present at least one new argument with its own critical literal $\neg A$ as its conclusion.

4. An agent cannot attack its own arguments. In our dialogue game framework it is not admissible for an agent to contradict itself by putting forward rules with a conclusion that contradicts a rule previously presented by the agent.

5. A particular dialogue game is won by an agent when the other party at its turn cannot make an admissible move.

6. A argument r is stronger than an argument s, conflicting with r and played in the previous time-step, if r is not attacked in successive steps.

7. A valid argument will remove all contradictory arguments from the common set of knowledge.

5.3.2 Strengthening of rules

Dialogues are parsed into a defeasible theory. The time of a dialogue is translated as the time of a rule. All the rules presented at the current step t_i and at the adjacent previous step t_{i-1} are defeasible rules. Here we consider time as a set of finite numbers and each number is one unit more or less than its previous or next number. If not immediately rebutted by the other party, we allow for the rule strength of a rule to be strengthened from defeasible into strict. Strengthening defeasible rules to strict rules is the central theme of this thesis.

The rationale behind this idea is as follows: arguments presented by an agent are defeasible in the sense that they are available for attack by the other agent in the next step. But if the other agent does not attack in the next step then it can not attack in future (as assumptions will be changed to facts, which can not be rebutted and defeasible rules will be changed to strict rules which can be only undercut). This gives the agents a specific opportunity to attack the defeasible rules and extracts the strategic behaviour of the agents. Consider an example as follows:

Let the set of arguments for the critical literal cl, presented by an agent A at the step i be $Ar_i = \{\Rightarrow d, \Rightarrow b, d, b \Rightarrow a, a \Rightarrow cl, \}$. Ar_i can be represented as a tree [5] shown in Figure 5.1, where each node represents a literal with the root as the critical literal, a directed path from a node (say x or child node) to another node (y or parent node) represents a rule as $x \Rightarrow b$ and the leaf nodes are either assumptions or facts. Here child and parent relations consider only two nodes with a directed path from child to parent and do not include transitive relations. Given a node n_1 with child nodes $n_2, \ldots, n_k$ represents a rule as $n_2, \ldots, n_k \Rightarrow n_1$.

Now agent B presents an argument Ar_{i+1} in the next step. In Ar_{i+1}, B can attack any set of nodes in the tree representing Ar_i. Say B attacks nodes $z = \{z_1, \ldots, z_h\}$. By doing so all parent nodes of z (and z itself) are defeated by B and B has chosen not to defy the child nodes of z and all other nodes which are not in z and not among the parents of z. For example Ar_{i+1} can be $\Rightarrow x, x \Rightarrow \neg b$. So that B chooses not to defy $\Rightarrow d$. This choice of attack shows the strategic behaviour of the agents as follows:

- B can use $\Rightarrow d$ in its future arguments. Given a choice of counter arguments B have to choose which arguments has strategic importance in the sense that B can use arguments of A in its favour as it may be the case that B had no information about the assertion d (or had sufficient doubt about the assertion d) but has information about arguments which involve d as premise and that defy Ar_i. A's assertion of d gives B the chance to proceed with an argument that involves an assertion of d. So the agents choose their counter arguments wisely, which reflects their strategic behavior.

- A should be careful about what information it should reveal in order to prove cl as we have seen that B can use some information from A to defy A's arguments which is

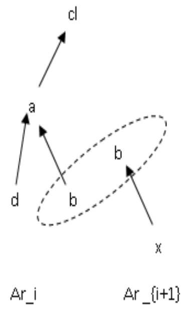

Figure 5.1: Strategic behaviour

previously unknown to B. So A's choice of Ar_i reflects the strategic behaviour of A.

In this thesis we present the mechanism to guide the dialogue game that supports strategic behaviour as we develop dialogue game protocols. But the mechanism for choosing arguments those are strategically important is not in the scope of this thesis.

A rule is represented as $R^t_x|x \in (d, s, sd)$ where t is the time (or the move when the rule has been played), d means the rule is defeasible, s means the rule is strict, sd means the rule is either strict or defeasible. We write $a@t$ to denote the literal a being put forward or upgraded at time t. The condition for upgrading a defeasible rule to a strict rule is described below.

If P is the conclusion of a defeasible rule of the adjacent previous step t_{i-1}, we upgrade the strength of the rule to strict in next step t_{i+1}if

$$\exists r \in R^{t'}_d[P]\ t' < t\ \forall t'' : t' < t'' < t\ R^{t''}_{sd}[\neg P] = \emptyset \text{ and } \forall a \in A(r) : +\Delta a@t.$$

We consider that the strength of the strict rule is greater than the defeasible rule.

If $\forall r \in R^{t_1}_s[q]$ and $\forall s \in R^{t_1}_d[\neg q]$, then $r >^{t_1} s$.

At each step of the game, if an argument (rule) has precedence over any contradictory defeasible rule of the previous steps, an agent is allowed to put forward that argument. The strength is determined either by previously known superiority relationships or the validity of the rule. We assume that if at time t_2 we have a *valid* rule $R^{t_2}2$ which contradicts a defeasible rule $R^{t_1}1$ of time t_1 and $t_2 > t_1$, the strength of $R^{t_2}2$ is greater than $R^{t_1}1$. We will use defeasible logic to determine the strength of a new rule.

$w > s$ if $+\partial(w > s)$

$+\partial(w > s)@t$ iff $w > s \in (>)$ or $w \in R^{t'}[P]$, $s \in R^{t}[\neg P]$ where $t' < t$.

5.3.3 Transition rules

The sets of common arguments construct the theories $T_1, T_2, T_3, \ldots$ respectively. Here the subscripts indicate the time at which the sets of common arguments are constructed. If at time 1, the game begins and arguments in support of a critical literal A are put forward by the proponent, then T_1 contains only defeasible rules. At time 2, the opponent proposes new defeasible rules which by the above presented precedence rules are stronger than some rules in theory T_1. The set of common arguments of the first two theories T_1 and T_2 consists only of defeasible rules. (Theory T_2 consists of defeasible rules from both time 1 and time 2.) Let the first theory $T_1 = (\{\}, R_d^1, >)$ be created from arguments (ARG_1) of the proponent, and the second theory $T_2 = (\{\}, R_d^2, >)$ be created through modifications of T_1 by arguments (ARG_2) from the opponent. Now the transition rules from the first theory to the second theory are:

1. If $r \in R_d^1$ and $\forall s \in ARG_2.\ \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_d^2$.

2. All rules of ARG_2 are added to T_2 as defeasible rule. Here we assume that ARG_2 is valid and that a valid argument, by the above defined precedence relations, is stronger than any contradictory argument of the previous step.

At time 3, theory T_3 is created through modification of T_2 by arguments (ARG_3) of the proponent. The rules for transition from T_2 to T_3 are :

1. If $r \in R_d^1$ and $\forall s \in ARG_2, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_s^3$. Here we should note that the proponent will not oppose its previous argument. Thus, all unchallenged rules of time 1 are upgraded as strict rules at time 3.

2. If $r \in R^2_d$ and $\forall s \in ARG_2, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R^3_d$. All unchallenged defeasible rules of time 2 are added to T_3 as defeasible rules at time 3.

3. All rules of ARG_2 are added to T_3 as defeasible rules. Here we assume that ARG_2 is valid and that a valid argument, by the above defined precedence relations, is stronger than any contradictory argument of a previous step.

Subsequent transitions are conducted in the same way as presented in the transition rules from T_2 to T_3. To be noted is that in the first two theories only proof of $+\partial A$ or $-\triangle A$ of a literal A could result. Thus, this framework needs at least three steps in order to determine a winner of a particular game. When a party cannot produce defeasible rules that will defeat contradictory defeasible rules presented by the other party at the previous step, the undefeated defeasible rules of the latter party will be strengthened into strict rules. This allows this party to support its argument and hence prove the critical literal definitely. Let theory T_i be created through modification of theory T_{i-1} by argument (ARG_i) of player 1 and the critical literal A is defeasibly proven $+\partial A$ in this theory. Let it be that at time $i+1$, player 2 cannot produce any arguments defeating the arguments of player 1, then player 1 wins at time $i+2$ as the proof as $+\partial A$ by the strengthening of the rules from defeasible into strict will result in the proof $+\triangle A$ at time $i+2$.

5.3.4 An example

Now we will present the model of dialogue game defeasible logic using an example. Consider an argumentation game with two players, Alice and Bob. Agent Alice is trying to prove A and agent Bob is trying to prove $\neg A$ using defeasible logic. At each step of the dialogue game they maintain a current set of rules (CR_t, where t is the time) in which a rule consists of its name $R'i$ (where $R'i$ indicates that the rule belongs to the current set CR_t as opposed to the rules present in the private knowledge bases of the parties denoted by Ri), its antecedent $A(r)$, which is a finite set of literals, an arrow, its consequent $C(r)$ which is a literal and $@t_x x \in \{1, 2, 3, \ldots, n\}$ denotes the time of the rule), which is updated at each step. Let it be that at time t_1 the game starts and Alice makes the first move as :

$R1 : \emptyset \Rightarrow B$, $R2 : B \Rightarrow A$.

This will generate two defeasible rules as $R'1(\emptyset \Rightarrow B)@t_1$, $R'2(B \Rightarrow A)@t_1$. Thus at time t_1 CR_{t_1}=[$R'1$, $R'2$] and we have proof $+\partial A@t_1$. But at time t_2, Bob presents new evidence in order to disprove A. At time $t_2(t_2 > t_1)$, Bob presents the following argument:

$R3 : \emptyset \Rightarrow D$, $R4 : D \Rightarrow \neg A$

This will generate two new defeasible rules as $R'3(\emptyset \Rightarrow D)@t_2$, $R'4(D \Rightarrow \neg A)@t_2$. Now, Bob only attacks $R'2$ presented by Alice at the previous step by $R'4$ and $R'2$ is removed from CR. Note that as $t_2 > t_1$, the strength of $R'4$ is greater than $R'2$ the strength determination rule. At time t_2, $R'1$ remains unchallenged and it is changed to a strict rule as $R'1(\emptyset \rightarrow B)@t_2$(which is a fact). Note that we change the time stamp of the rule from $t - 1$ to t_2 to indicate that it is a member of CR at time t_2. At each step the time stamp of all facts and strict rules are also changed to current time to match with the CR time stamp. So CR_{t_2} becomes [$R'1$, $R'3$, $R'4$]. The proof at time t_2 is $+\partial\neg A$. To be noted is that at this time the common set of knowledge also consists of the knowledge presented by Bob at t_2. Thus, in order to succeed the argument presented by Alice needs to defeat these (new) arguments as well. Therefore, at time t_3, Alice presents the following arguments:

$R5 : B \Rightarrow \neg D$, $R6 : \emptyset \Rightarrow E$ and
$R7 : E \Rightarrow A$.

So the translated defeasible rules are $R'5(B \Rightarrow \neg D)@t_3$, $R'6(\emptyset \Rightarrow E)@t_3$, $R'7(E \Rightarrow A)@t_3$. Now the CR_{t_3} is [$R'1$, $R'5$, $R'6$, $R'7$] as $R'3$ and $R'4$ are defeated by $R'5$ and removed. So the proof at time $t3$ is $+\partial A$. If Bob does not present a valid argument in the next step, Alice wins the game as we allow an agent to upgrade the strength of unchallenged rules in the next time $t4$. Thus, in $t4$ Alice can upgrade the defeasible rules supporting the proof of $+\partial A$ into strict rules and subsequently prove $+\Delta A$.

Reconsideration

By reconsideration we mean an agent can change its argument put forward at a previous step. In our argumentation framework, as in dialogue games in general, reconsideration is not possible as rules are either carried to the next step as strict rules or facts, or they are defeated and removed. If the rule is removed, the agent can no longer argue based on this previous decision. Also, if the status of the rule is strengthened it cannot be defeated and removed. But in a step an agent can have more than one set of suitable arguments. Here we present some intuitions on how to efficiently distinguish between these choices.

An agent can argue with additional information even if it is not related to the current argument in order to block an opponent's future arguments at an early stage. For example, if at t_1, Alice presents two arguments as $R1 : A \Rightarrow B$ and $R2 : \Rightarrow \neg D$, which is defended by Bob at t_2 by $R3 : C \Rightarrow \neg B$ R2 is strengthened into a fact at t_3. Now at t_3, Alice passes argument $R4 : E \Rightarrow B$. Thus, at t_4 Bob has only one argument to defend as $R5 : D \Rightarrow \neg B$. Bob cannot put forward argument R5 as R2 is a fact and stronger than R5. Hence Alice wins. This will save one step as if Alice had not passed R2 at t_1, Bob will present R4 at t_4 and it has to play R2 at t_5.

The goal of an agent is to win the game with minimum cost. The agent can always make some additional moves if we assume that cost is associated with the number of steps in a game rather than the number of arguments put forward in the game.

> Cost=$\sum_{i=T_s}^{T_f} k$, where T_f and T_s are the ending time point and starting time point respectively, and k is the cost of each step. Here we assumed that each step takes the same duration irrespective of how many arguments are played.

5.4 Conclusion and future work

We have presented a dialogue game framework in defeasible logic. We have shown that reconsideration of arguments are unnecessary and can be avoided.

A dialogue game does not allow for backtracking or reconsideration. At a given time, however, a player may have more than one suitable argument to choose from. Although in this chapter we have not presented any algorithm for strategies on how to win a particular dialogue game, we have provided a foundation for development of such strategies. We have introduced the cost function in a dialogue game and discussed how a strategy can be developed with the aim of maximizing the payoff of the game [47]. In addition, this framework could be extended to model the behavior of an agent φ in a dynamic environment ϵ. By representing the environment as one of the parties, the agent φ is enabled, by putting forward arguments, to reason on its environment in both reactive and proactive way. This would allow for a natural characterization of the environment as the uncertainty in the environment would be modeled as the private knowledge of ϵ. This protocol is extended to an asymmetric protocol in the next chapter.

6

An asymmetric protocol for dialogue games

6.1 Introduction

Adversarial situations arise as agents in pursuit of their goals interact with other agents pursuing goals of a conflicting nature. In a setting where issues need to be resolved, the agent interaction could be modeled as an adversarial argumentation game. Argumentation games are defeasible, meaning that an argument put forward by one of the agents in support of a conclusion could be defeated by contrary evidence and arguments put forward by the other agent. Thus, the agents resolve the dispute by putting forward the arguments that will enable the best outcome for their case. Using a symmetrical protocol an argumentation game between homogeneous (equally strong) parties could be modeled. However, as in many real-life settings, disputes also arise in agent system where the claims of the agents involved in the interaction (*e.g.* regarding distribution of a scarce resource) are of unequal importance to the overall goal of the agent system, and thus need to be handled accordingly. In addition, as in many real-life situations the evidence presented by the parties of a dispute are inconclusive and the accompanying arguments incoherent. Thus, a majority of the disputes has to be resolved by higher-level principles guiding the interaction. One important principle is referred to as the burden of proof, *cf. e.g.* [46].

To accommodate a correct outcome for argumentation games in heterogeneous agent systems, we present an asymmetric protocol for adversarial argumentation games.

The chapter is organized as follows. In Section 6.2 we present argumentation games and their setup. We discuss the formalization of argumentation games using defeasible logic as presented in [49] in Section 6.3. Section 6.4 presents the asymmetrical protocol for argumentation games. We use a criminal litigation setting to illustrate and discuss some of the benefits of the model.

6.2 Argumentation games

Consider an adversarial argumentation (dialogue) game as an interaction between two parties, the *proponent* and the *opponent*. The two parties debate a topic. Each equipped with a set of arguments, the parties take turns in putting forward a subset of these arguments, *i.e. move*, with the sole purpose of justifying their claim. The game is governed by a protocol for admissible moves and the winning conditions. For the proponent, a basic protocol for an argumentation game would be that the arguments of the move attack the previous move of the adversary, and that the main claim follows from the arguments assessed as currently valid. For the opponent, an admissible move has to attack the previous move of the adversary, and the main claim is not derivable. Even though more complex winning criteria could be devised, by a basic protocol, a player wins the argumentation game when the other party is out of admissible moves.

6.3 Dialogue games in defeasible logic - a symmetric protocol

In Thakur et al. [49] we presented a model for an argumentation game in Defeasible Logic. The model provided is of a basic symmetric protocol for an adversarial dispute. We parse the dialogue into defeasible rules utilizing the time of the dialogue as the time of the rule. In order to resolve the dispute, the agents take turns in putting forward arguments from a private knowledge base, *i.e.* a finite set of (defeasible) arguments in support of their claim. At each time step, an agent is allowed to put forward any of its arguments (rules) that has precedence over any contradictory defeasible rule of the previous steps.

In this symmetric protocol we assume that if at time t_2 we have a *valid* rule $w \in (R_{sd}^{t_2})$ which contradicts a defeasible rule $s \in (R_d^{t_1})$ of time t_1 and $t_2 > t_1$ then the strength of w is greater than s. $a@t$ denotes the literal a being put forward or upgraded at time t:

$$w > s \text{ if } +\partial(w > s)$$

$$+\partial(w > s)@t \text{ iff } (w, s) \in (>) \text{ or}$$
$$w \in R^{t'}[P],\ s \in R^{t}[\neg P] \text{ where } t' < t\ .$$

A common public knowledge base holds the common knowledge, which is a theory in defeasible logic. The sets of agreed common knowledge construct the theories $T_1, T_2, \ldots, T_n$ respectively as the undefeated defeasible rules from the previously adjacent step are strengthened into strict rules and the defeated defeasible rules are removed. Thus, if P is the conclusion of a defeasible rule of the adjacent previous step t_{i-1}, regardless of its origin, the agreed common knowledge is created by strengthening the status of rules from defeasible to strict in the adjacent next step t_{i+1} if:

$$\exists r \in R_d^{t'}[P]\ t' < t\ \forall t'' : t' < t'' < t\ R_{sd}^{t''}[\neg P] = \emptyset \text{ and}$$
$$\forall a \in A(r) : +\Delta a@t.$$

The proof procedures of the defeasible logic are applied to the critical literal at each time step, thus determining the burden of proof and the outcome of the argumentation game. The first theory $T_1 = (\{\}, R_d^1, >)$ is created from the arguments (ARG_1) presented by the first player, and the second theory $T_2 = (\{\}, R_d^2, >)$ is created through modifications of T_1 by the arguments (ARG_2) presented by player 2. The transition rules from the first theory to the second theory were devised as follows:

1. If $r \in R_d^1$ and $\forall s \in ARG_2$, $\neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_d^2$.

2. By the assumption that all rules of (ARG_2) are valid, and by the above defined precedence relations all valid arguments are stronger than any contradictory rules of the previous step, we add all rules of (ARG_2) to T_2 as defeasible rules.

At time 3, theory $T_3 = (R_s^3, R_d^3, >)$ is created through modification of T_2 by arguments (ARG_3) put forward by the first player. The rules for transition from T_2 to T_3 are:

1. If $r \in R_d^1$ and $\forall s \in ARG_2, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_s^3$. All unchallenged rules presented at time 1 are upgraded to strict rules at time 3.

2. If $r \in R_d^2$ and $\forall s \in ARG_2, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_d^3$. All unchallenged defeasible rules of time 2 are added to T_3 as defeasible rules at time 3.

3. By the assumption that all rules of (ARG_3) are valid, and by the above defined precedence relations all valid arguments are stronger than any contradictory rules of the previous step, we add all rules of (ARG_3) to T_3 as defeasible rules.

The winning criteria for a basic game are devised as an agent to be winning if the claim q is definitely proven $+\Delta q$ at any time step. If an agent at any step of the game proves $+\partial A$, the burden of production as well as persuasion of $-\partial A$, or $-\Delta A$ or $+\Delta\neg A$ or $+\partial\neg A$ are placed on the other party.

Using a symmetric protocol, a dispute between equally strong parties could be modeled as an argumentation game and resolved accordingly. However, in many situations ethical, moral or other reasons (*cf.* *e.g.* criminal litigation [1]) advocate for special concerns to be taken on behalf of one of the parties. To accommodate such settings, asymmetric protocols are required.

[1] "Homo praesumitur bonus donec probetur malus"lat: Innocent until proven guilty. The adoption of this presumption of innocence in many national statutes results in that the defendant of a criminal litigation only is required to, at most, produce an exception to the accusation.

6.4 An asymmetric protocol

Here we present an asymmetric model for adversarial argumentation games in defeasible logic between two parties, the *prosecutor* and the *defendant*. As in the symmetric protocol, we parse the dialogue into defeasible rules utilizing the time of the dialogue as the time of the rule. Each agent has at its disposition a private knowledge base consisting of a finite set of defeasible arguments in support of their claim (the critical literal). Initiated by the prosecutor, the parties take turns in presenting their arguments. At each time step the proof procedures are applied to the critical literal. The outcome of an argumentation game is determined by the final stage of the game. For common sense reasons, as an argument put forward cannot be revoked from impacting the argumentation, we do not allow for backtracking.

The winning criteria for a basic game are devised as an agent to be winning if the claim q is definitely proven $+\Delta q$ at any time step. However, analogously to the burden of persuasion, which imposes a requirement to provide a justified (*i.e.* strongly defeating) argument for the issue on which the burden rests (based on rebutting defeat)[46], we require of the prosecutor a strong defeat of any argument (including the critical literal) presented by the defendant. Thus, if the prosecutor at any step of the game proves $+\partial A$, the prosecutor still holds the burden to produce proof of $+\Delta A$ in order to win. In contrast, for the defendant only a burden of production of an exception $+\partial\neg A$, (being subsumed by $-\partial A$, $-\Delta A$ or $+\Delta\neg A$) is imposed. If the defendant at any step of the game proves the exception $+\partial\neg A$, the burden of persuasion placed on the prosecutor necessitates the proof of $+\Delta A$ (including $-\partial\neg A$) in order for the prosecutor to win.

In the symmetric protocol, regardless of its origin, time brings strengthening of undefeated rules from defeasible to strict. Here we require that the strengthening of rules originating from the prosecutor only occurs when the rule could be derived from arguments already put forward by the defendant. In other cases, undefeated rules from the previously adjacent step presented by the prosecutor remain as defeasible rules in the common knowledge base. Defeated rules are removed at each step. As we do not allow the prosecutor to repeat arguments and the arguments put forward have to strongly defeat any arguments put forward by the defendant, the game will terminate.

In the symmetric protocol at each step any agent whose turn it is to move can present an argument if its strength is stronger than contradictory defeasible rules of the previous steps. In our asymmetrical distribution of the burden of proof, the defendant is allowed to present an argument that is merely weakly defeating the argument of the prosecutor of the previous steps. As a consequence, the defendant could remain with the same argument to fulfill his burden of production of an assumption $+\partial\neg A$ as response to $+\partial A$.

The strength of an argument is determined by either previously known superiority relationships or validity of that rule. Adhering to the above presented syntax, we write $y_i \in R_x^{t_j}|x \in (d,s,sd)$. Here y is a rule identifier with the subscripts $i \in \{p,d\}$ where p means that the origin of the rule is the prosecutor and d means that the origin of the rule is the defendant. In the following, unless needed, the indexing is left out for readability reasons. Moreover, we write $a@t^j$ to denote the literal a being put forward or upgraded at time t, where $t = (t^{'}, t^{''}, \ldots, t^n)$:

$$\text{If } \forall r \in R_s^{t_i}[q] \text{ and } \forall s \in R_d^{t_i}[\neg q], \text{ then } r >^{t_i} s.$$

We consider that the rule strength of a strict rule is greater than the rule strength of a defeasible rule:

$$w_d > s_p \text{ if } +\partial(w_d > s_p)$$

$$+\partial(w_d > s_p)@t \text{ iff } (w_d, s_p) \in (>) \text{ or } s_p \in R^{t'}[P] \text{ and } w_d \in R^{t''}[\neg P], \text{ where } t' < t'' < t.$$

For defeasible rules presented by the defendant we simply assume that if at time t_2 we have a *valid* rule $w_d \in R^{t_2}$ which contradicts a defeasible rule $s_p \in R^{t_1}$ of time t_1 and $t_2 > t_1$ then the strength of w_d is greater than s_p. This fits well with the burden of persuasion being placed on the prosecutor. We utilize defeasible logic to determine the strength of a new rule presented by the players.

$$s_p > w_d \text{ if } +\partial(s_p > w_d)$$

$+\partial(s_p > w_d)@t$ iff $s_p > w_d \in (>)$ or $w_d \in R^{t'}[\neg P]$ and $s_p \in R^{t''}[P]$ and $\forall a \in A(s) : +\Delta a@t$, where $t' < t'' < t$

else

$w_d > s_p$ if $+\partial(w_d > s_p)$

$+\partial(w_d > s_p)@t$ iff $w_d > s_p \in (>)$ or $w_d \in R^{t'}[\neg P]$ and $s_p \in R^{t''}[P]$ and $\neg\forall a \in A(s) : +\Delta a@t$, where $t' < t'' < t$.

As the prosecutor holds the burden of persuasion, we assume that unless the rule priority is set, that only if at time t_2 we have a *valid* rule $s_p \in R^{t_2}$ which contradicts a defeasible rule $w_d \in R^{t_1}$ of time t_1 and $t_2 > t_1$ and the rule presented by the prosecutor strongly defeats the rule of the defendant then the strength of the argument s_p of the prosecutor is *greater* than the argument of the defendant w_d. In all other situations the opposite goes, thus rendering the strength of the argument s_p of the prosecutor *weaker* than the argument of the defendant w_d.

In this asymmetric protocol the criteria for strengthening the rule strength of a defeasible rule to a strict rule are devised as follows:

$$\exists r \in R_d^{t'}[P]\ t' < t\ \forall t'' : t' < t'' < t\ R_{sd}^{t''}[\neg P] = \emptyset \text{ and } \forall a \in A(r) : +\Delta a@t.$$

If P is the conclusion of a defeasible rule $r \in R_d^{t'}$ of the adjacent previous step t' and the rule was presented by the defendant then we can upgrade the rule status from defeasible to strict in the next time step t if no counterarguments are presented by the prosecutor at time t''. Here, we write $a@t^j$ to denote the literal a being put forward or upgraded at time t, where $t = (t^{Odd'}, t^{Even'}, \ldots, t^{Odd^n}, t^{Even^n})$:

$\exists r \in R_d^{t^{Odd''}}[P]\ t^{Odd''} < t\ \forall t^{Even''} : t^{Even''} < t^{Odd''} < t\ R_{sd}^{t^{Even''}}[\neg P] = \emptyset$ and $\forall a \in A(r) : +\Delta a@t$ and

1) $\exists r \in R_{ds}^{t^{Even''}}[P]\ t^{Even''} < t\ \forall t^{Even''} : t^{Odd''} < t^{Even''} < t$ and $\forall a \in A(r) : +\Delta a@t$

or

2) $+\partial[P]@tfromR_{ds}^{t^{Even''}}$.

However, if P is the conclusion of a defeasible rule $r \in R_d^{t^{Odd''}}$ of the adjacent previous step $t^{Odd''}$ and the rule was presented by the prosecutor then we can upgrade the rule status from defeasible to strict in the next step only in the case of no counter arguments being presented by the defendant at the adjacently following time $t^{Even''}$ and the defeasible (or strict) rule r has been put forward by the defendant or the conclusion P follows defeasibly from the defeasible or strict rules $R_{ds}^{t^{Even''}}$ presented by the defendant at the adjacently following time $t^{Even''}$.

An argumentation game is initiated at time 1 by the prosecutor agent putting forward arguments (ARG_1) from its private knowledge into the common knowledge base to prove its claim (critical literal) A. As the parties take turns in presenting their arguments, at time 2 the defendant agent responds to the accusations. We allow arguments in the form of valid defeasible rules being as strong or stronger than at least some of the rules of theory T_1. The common sets of argument construct the theories $T_1, T_2, T_3, \ldots, T_n$ respectively where the subscripts indicate the time at which the common sets of argument are constructed. As all arguments in the private knowledge base of the agents are defeasible, in the first two theories the common set of arguments consists only of defeasible rules from both time 1 and time 2, according to the following transition rules:
Let the first theory $T_1 = (\{\}, R_d^1, >)$ be created from arguments (ARG_1), the operative plea of prosecutor, and the second theory $T_2 = (\{\}, R_d^2, >)$ be created through modifications of T_1 by arguments (ARG_2) from the defendant. Now the transition rules from the first theory T_1 to the second theory T_2 are as follows:

1. If $r \in R_d^1$ and $\forall s \in ARG_2, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_d^2$.

2. All rules of (ARG_2) are added to T_2 as defeasible rules, under the assumption of (ARG_2) being valid and that, by the above defined precedence relations, any valid argument from the defendant is stronger than its contradictory argument (from the prosecutor) of the adjacent previous step. As all unchallenged rules of the prosecutor are added to T_2 as defeasible rules T_2 now consists of all unchallenged rules of the prosecutor and all arguments (ARG_2) of the defendant.

At time 3, theory T_3 is created through modification of T_2 by arguments (ARG_3) of the prosecutor. Accounting for the heterogeneity of the parties we capture the asymmetrical burden of proof by the following rules for transition from theories T_2 to T_3:

1. If $r \in R^1_d$ and $\forall s \in ARG_2, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, and 1) $r \in R^2_d$ or 2) $R^2_d \vdash C(r)$, then $r \in R^3_s$.

2. If $r \in R^1_d$ and $\forall s \in ARG_2$, $\neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R^3_d$. Here we should note that, in contrast to the symmetric protocol, even though the defendant has not actively challenged these arguments and the prosecutor will not oppose its previous argument by the rules of the game, we find it to be a too strong presumption to strengthen the rule status of these rules to strict rules. Thus, unless the argument is acknowledged by the defendant (see transition rule 1.), all unchallenged rules of time 1 of the prosecutor remain as defeasible rules at time 3.

3. If $r \in R^2_d$ and $\forall s \in ARG_3, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R^3_d$. All unchallenged defeasible rules of time 2 (originating from the defendant) are added as defeasible rules at time 3.

4. If $r \in R^2_d$ and $\forall s \in ARG_3, \neg C(s) = C(r) \wedge \neg C(s) \notin A(r)$, AND $r \geq s$ then $r \in R^3_d$. For removal it is required that all rules of the defendant have to be strongly defeated by the prosecutor. Thus, the defeasible rules of time 2 (originating from the defendant) of equal or stronger strength are added as defeasible rules at time 3.

5. If $r \in ARG_3$ and $\forall s \in R^2_d$, $\neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R^3_d$. All unchallenged rules of (ARG_3) are added to T_3 as defeasible rules.

6. If $r \in ARG_3$ and $\forall s \in R^2_d$, $\neg C(s) = C(r) \wedge \neg C(s) \notin A(r)$ AND $r > s$, then $r \in R^3_d$. All rules of (ARG_3) that are of higher priority, *i.e.* strongly defeat the arguments of the defendant, are added to T_3 as defeasible rules. Here due to the burden of production on the prosecutor, all arguments added are required to either be unchallenged or to strongly defeat all previous arguments of the defendant. This way, by putting forward new arguments, the prosecutor could strengthen its claim.

As a result, T_3 consists of the unchallenged defeasible rules of T_1 of the prosecutor, the unchallenged defeasible rules and the rules T_2 of the defendant that are challenged by (ARG_3) but found equally strong or stronger, and the unchallenged defeasible rules of the prosecutor from (ARG_3).

At time 4, theory T_4 is created through modification of T_3 by arguments (ARG_4) of the defendant. The transitions from T_3 to T_4 are devised as follows:

1. If $r \in R_d^2$ and $\forall s \in ARG_3, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_s^4$. The defendant will not oppose its previous argument by the rules of the game. Thus, all the unchallenged defeasible rules R_d^2 are upgraded as strict rules, *i.e.* facts, at time 4.

2. If $r \in R_d^3$ and $\forall s \in ARG_4, \neg C(s) \neq C(r) \wedge \neg C(s) \notin A(r)$, then $r \in R_d^4$. All unchallenged defeasible rules R_d^3 are added as defeasible rules at time 4. As already stated, the defendant cannot challenge her own rules presented in R_d^3.

3. All rules of (ARG_4) are added to T_4 as defeasible rules. In contrast to arguments *e.g.* (ARG_3) originating from the prosecutor, as (ARG_4) originates from the defendant we merely require that all rules of (ARG_4) are valid and at least as strong (or stronger) than any of its contradictory arguments presented by the prosecutor.

The following theories $T_5, T_6, T_7, \ldots, T_n$ are constructed by the transition rules for the theories T_3, and T_4 while acknowledging the alternating moves of the agents.

An example – Presumption of Innocence

The asymmetric model of argumentation game defeasible logic is illustrated by elaboration on the example of [49]. Consider a particular argumentation game between the prosecutor Alice and the defendant Bob. Alice is trying to convict Bob by proving A and Bob is claiming $\neg A$. At each step they maintain a current set of rules (CR_t, where t is the time). Here a rule consists of its name $R'i$ (where $R'i$ indicates that the rule belongs to the current set CR_t as opposed to the rules present in the private knowledge bases of the parties denoted by R_i), its antecedent $A(r)$, which is a finite set of literals, an arrow, its consequent $C(r)$, which is a literal and $@t_x | x \in \{1, 2, 3, \ldots, n\}$ denotes the time of the rule, which is updated at each step. Alice initiates the game at time t_1 by presenting her first move as :

$$R1 : \emptyset \Rightarrow B \qquad R2 : B \Rightarrow A.$$

This will generate two defeasible rules as $R'1(\emptyset \Rightarrow B)@t_1$, $R'2(B \Rightarrow A)@t_1$. Thus at time t_1 CR_{t_1}=[$R'1$, $R'2$] and we have proof $+\partial A@t_1$. Now at the next time point, Bob gets his chance to disprove A. At time t_2, Bob presents the following argument:

$$R3 : \emptyset \Rightarrow D \qquad R4 : D \Rightarrow \neg A.$$

This will generate two new defeasible rules as $R'3(\emptyset \Rightarrow D)@t_2$ $R'4(D \Rightarrow \neg A)@t_2$. Now, Bob only attacks $R'2$ presented by Alice at the previous step by $R'4$ and $R'2$ is removed from CR_{t_2}. Note that as $t_2 > t_1$, the strength of $R'4$ is greater than $R'2$ according to the strength determination rule for the defendant. At time t_2, $R'1$ remains unchallenged but as this rule is not utilized even as a premise in the reasoning of Bob it does not commit Bob to this rule, (leaving Bob the possibility to dismiss this rule by contesting it at a later time or Alice to present evidence to strengthen this rule by the strength determination rule). Thus, the rule remains in CR_{t_2} as $R'1(\emptyset \Rightarrow B)@t_2$. This is in contrast to the symmetric protocol where $R'1$ when unchallenged is changed to a strict rule $R'1(\emptyset \rightarrow B)@t_2$ (a fact) regardless of its origin. Note that we change the time stamp of the rule from t_1 to t_2 to indicate that it is a member of CR_{t_2} at time t_2. Thus, CR_{t_2} =[$R'1$, $R'3$, $R'4$]. The proof at time t_2 is $+\partial\neg A$ (which implies that we also have $-\Delta$ A as the latter rule $R'4$ is stronger than $R'2$ in accordance to the first strength determination rule.

Next at time t_3, in order to defeat the arguments presented by Bob, Alice presents the following arguments:

$$R5 : B \Rightarrow \neg D \qquad R6 : \emptyset \Rightarrow E \qquad R7 : E \Rightarrow A.$$

So the translated defeasible rules are $R'5(B \Rightarrow \neg D)@t_3$, $R'6(\emptyset \Rightarrow E)@t_3$, $R'7(E \Rightarrow A)@t_3$. Now the CR_{t_3} is [$R'1$, $R'3$, $R'4$, $R'6$] as the rule $R'4$ is stronger than $R'7$ and the rule $R'3$ is stronger than $R'5$ according to the transition rule as neither the argument $R'7$ nor $R'5$ can strongly defeat $R'4$ or $R'3$ respectively and thus they are removed. So the proof at this time point remains $+\partial\neg A$. If Alice cannot present any additional arguments strongly defeating $R'4$ in the next step the rule $R'4$ is strengthened into a strict rule resulting in the proof of $+\Delta\neg A$. Thus, in contrast to the symmetric protocol example where Alice wins the game as she was able to upgrade the defeasible rules supporting the proof of $+\partial A$ by use of the rule priority assumption in 6.4, she would need the rule to be strongly defeated as, for example, by addition of a rule $R'i : E \Rightarrow \neg D$ to prevent the rule $R'4$ from being strengthened into a strict rule at time t_4. As this is not the case, Bob wins the game at time t_4 and is acquitted from the criminal charge of A.

Another Example - Beyond Reasonable Doubt

Consider a second argumentation game between the prosecutor Alice and the defendant Bob. Alice is still trying to convict Bob by proving A and Bob is claiming $\neg A$. Again Alice initiates the game at time t_1 by presenting her first move as:

$$R1 : \emptyset \Rightarrow B \qquad R2 : B \Rightarrow A.$$

This will generate two defeasible rules as $R'1(\emptyset \Rightarrow B)@t_1$, $R'2(B \Rightarrow A)@t_1$. Thus at time t_1 CR_{t_1}=[$R'1$, $R'2$] and we have proof $+\partial A@t_1$. Now at the next time point, Bob gets his chance to disprove A. At time $t_2(t_2 > t_1)$, Bob presents the following argument:

$$R3 : E \Rightarrow D \qquad R4 : (B \wedge D) \Rightarrow \neg A.$$

This will generate two new defeasible rules as $R'3(E \Rightarrow D)@t_2$, $R'4(B \wedge D \Rightarrow \neg A)@t_2$. Now, Bob only attacks $R'2$ presented by Alice at the previous step by $R'4$ and $R'2$ is removed from CR. As $t_2 > t_1$, the strength of $R'4$ is greater than $R'2$ according to the strength determination rule for the defendant. Thus, CR_{t_2} =[$R'1$, $R'3$, $R'4$]. The proof at time t_2 is $-\partial$ A, which includes that we also have $-\Delta$ A as the latter rule $R'4$ is stronger than $R'2$.

Next at time t_3, Alice presents the following arguments :

$$R5 : \neg E \Rightarrow \neg D \qquad R6 : \emptyset \Rightarrow \neg E \qquad R7 : B \Rightarrow A.$$

The translated defeasible rules are $R'5(\neg E \Rightarrow \neg D)@t_3$, $R'6(\emptyset \Rightarrow \neg E)@t_3$, $R'7(B \Rightarrow A)@t_3$. Now CR_{t_3} =[$R'1$, $R'4$, $R'5$, $R'6$, $R'7$] as $R'3$ is strongly defeated by $R'5$ and $R'6$ and thus, it is removed. At time t_3, $R'1$ remains unchallenged and as it is utilized as a premise in the reasoning of Bob and thus commits Bob to this rule (which is justified as Bob could not be allowed to rely on not actively presented inconsistencies), when presented by the prosecutor Alice it is strengthened into a strict rule (*i.e.* a fact) as $R'1(\emptyset \rightarrow B)@t_3$. So the proof at this time point is $+\partial A$ as the rule is stronger according to the transition rules. If Bob does not present valid arguments in the next step Alice wins the game as from Bob's argumentation her claim is corroborated.

6.5 Conclusion

In this chapter defeasible logic is used to capture an asymmetric protocol for argumentation games. We have shown that our model provides for a closer approximation of argumentation games for heterogeneous agent settings. The agent characteristics or the agents' relative importance in fulfilling the overall goal of the system could be captured, while the agent is allowed to argue its case in the best way it knows, for example choosing at what time any subset of its arguments (*i.e.* private knowledge) be disclosed to its adversary.

7

A new semantics for locutions

In this chapter we discuss our second research problem. We want to model semantics of location in terms of how efficient locutions are in conveying the right ‘meaning’ [3].

7.1 Introduction

Dialogue game protocols are developed as agent interaction protocols in many domains of application as persuasion dialogue [43], information seeking dialogue [27], coalition formation dialogue [24] and negotiation dialogue [48]. [33] has structured dialogue game protocols into components as locutions, combination rules and termination rules. [34] has argued that as different dialogue game protocols are developed there is a need to compare these protocols in order to choose an appropriate protocol for a particular agent interaction scenario. [36] provides a functional comparison of dialogue game protocols based on the components of protocols.

Dialogue game protocols are developed in isolation from agent design. So a functional comparison between protocols only provides functional differences between components of protocols but not with respect to agent's architecture. While components of a dialogue game protocol provide a structured approach to develop a protocol, semantics of these components imposes some restriction on agents to communicate with other agents. Agent's interactions are more fruitful if the agents have more information about each other. So instead of just reacting to a set of rules, if an agent knows another agent's mental state that produces the set of rule, it can produce a better response. As agents will be interacting using a dialogue game protocol, to convey a state of mind a speaker agent has to choose from a given set of components that best describes the agent's mental state. Similarly a receiver agent has to relay on the semantics of a component of a protocol to interpret the speaker's mental state. In this chapter we will model the difference between what an agent wants to convey to the other agent and what it can communicate using a dialogue game protocol.

The chapter is organized as follows: in section 7.2 we summarize the protocol given in [49] and [19]. In Section 7.3 we present a tree representation of a dialogue game which will be used to describe components of the protocol. In Section 7.4 we model agent's knowledgebase. Components of the protocols are developed according to component classification of [33]. In Section 7.5 we present our intuition for components of a dialogue game protocol. In Sections 7.5.1 and 7.5.2 we describe semantics of locutions and combination rules.

7.2 Dialogue game protocol

In this section we summarize the dialogue game protocol described in [49] [1]. Let the participating agents be $Agent_1$ and $Agent_2$. Each agent is equipped with a knowledge base. A knowledge base is a theory in defeasible logic (for details see [22]). Let the knowledge base for each agent be KB_1 and KB_2 where, $KB_i = \{F, R^s, R^d, >\}|i \in \{1, 2\}$. F is a set of facts, R^s is a set of strict rules, R^d is a set of defeasible rules with superiority relation $>$. For

[1] In this section we repeat the dialogue game protocol presented in Section 5. The representations used in this section in describing the protocol will be used in the rest of the chapter.

simplicity, we assume that the agents are arguing about a single literal l^C(will be referred to as the critical literal). The semantics of the protocol[2] is as follows:

1. The agents present their arguments and they build a common theory, represented as Th_i^{com} (this theory holds all rules presented by both agents in their turns except the rejected rules). Here i represents the step of the game when Th_i^{com} is created.

2. All arguments presented by an agent are initially treated as defeasible rules. If in the next step the other party fails to provide the valid counter arguments, these defeasible rules will be upgraded to strict rules.

3. The arguments in support of a critical literal $\neg A$ presented by a party at any step must attack (at least) the conclusion in support of the critical literal A put forward by the other party in the previous step. Moreover, in order to prevent a strengthening of the defeasible rules in support of A in the set of common arguments and to remove from the set of common arguments all defeasible rules which have as conclusion A, party 1 must present at least one new argument with its own critical literal $\neg A$ as its conclusion.

4. An agent cannot attack its own arguments. In our dialogue game framework it is not admissible for an agent to contradict itself by putting forward rules with a conclusion that contradicts a rule previously presented by the agent.

5. A particular dialogue game is won by an agent when the other party at his turn cannot make an admissible move.

6. A argument r is stronger than an argument s, conflicting with r and played in the previous time-step, if r is not attacked in successive steps.

7. A valid argument will remove all contradictory arguments from the common set of knowledge.

8. At any step of the game, we use defeasible rules to process the proof for l^C.

[2] The full semantics in Chapter 5.

9. The game ends when a party runs out of moves.

7.3 Dialogue tree

As mentioned in the previous protocol, a common theory is maintained in the dialogue game and defeasible logic is applied to check the proof of the critical literal with respect to the common theory. In this section we represent the process of construction of the common theory as a tree.

> Let the dialogue tree be represented as $< R, \rightarrow^{S|O} >$ where R is a set of rules (R contains both strict and defeasible rules) and each rule represents a node. $\rightarrow^{S|O} \subset R \times R$ is a set of relations among the nodes. The direction of the arrow shows the dependency between nodes. For example $R1 \rightarrow^{S} R2$ means $R1$ supports $R2$. $R1 \rightarrow^{O} R2$ means $R1$ opposes $R2$, where either $R1$ and $R2$ are defeasible rules and $R1$ is stronger than $R2$ or if $R2$ is a strict rule then $R1$ undercuts $R2$. The root of the tree is a node which has a rule with conclusion l^C or $\neg l^C$ or it opposes a node which contains a rule whose conclusion is l^C.

Consider an example: At step 1 (start of a dialogue game), $Agent_1$ presents the following rules :

$R1 : A \Rightarrow l^C$

$R2 : B \Rightarrow A$

$R3 :\Rightarrow B.$

The corresponding dialogue tree is shown in Figure 7.1.

At step 2, $Agent_2$ proposes the following rules

$R4 : C \Rightarrow \neg l^C$

$R5 :\Rightarrow C$

If $R4$ is stronger than $R1$ then the tree will be changed as shown in Figure 7.2. A relation will be called a defeated relation if it is opposed by another relation. As shown in Figure 7.2 $R2 \rightarrow^{S} R1$ is defeated by $R4 \rightarrow^{O} R1$. A defeated relation will be drawn as a dashed arrow.

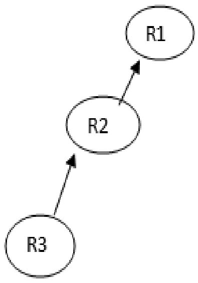

Figure 7.1: Dialogue tree at step 1

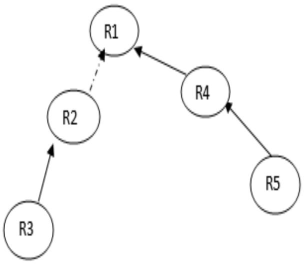

Figure 7.2: Dialogue tree at step 2

Next we introduce the parent-child relationship between nodes. We define a parent relation between two nodes as:

$R2$ is parent of $R1$ or Parent$\{R1$, $R2\}$, if

- there is relation from $R1$ to $R2$.
- there is a node with rule $R3$ and there are relations as $Parent\{R1, R3\}$ and $Parent\{R3, R2\}$.

We define a child relation as :

$R1$ is child of $R2$ or $Child\{R1, R2\}$, if

- there is relation from $R1$ to $R2$.

- there is a node with rule $R3$ and there are relations as $Child\{R1, R3\}$ and $Child\{R3, R2\}$.

Now we introduce the concepts of *active*, *dead* and *sleepy* parts of a tree.

An *active* tree is one whose entire nodes are either not challenged or undefeated. The maximal subset M of T such that all nodes are connected by a normal link where :

1. There is node (N) whose conclusion is l^C or there is a node (N) who has relation $N \rightarrow^O N^{'}$ where conclusion of $N^{'}$ is l^C.
2. For all other nodes ($N_i | i \in \{1 \dots k\}$) in M, there is a parent relation as $Parent\{N, N_i\} | \forall i \in \{1 \dots k\}$.
3. There is no node N_i in M, such that there is an oppose relation as $n \rightarrow^O N_i$ where n is a node outside M. This suggests that all rules in M are stronger than contradictory rules in T.

A *sleepy* tree is a fragment of T which is not active. A node becomes a sleepy tree when its parent node is defeated by the counter argument but the node itself and its children remain undefeated. A subset S of T is a *sleepy* tree if :

1. There is no node in S which contains a rule with conclusion l^C.
2. $\exists s \in S$ such that $\forall s^{'} \in S/s$ we have a parent relation as $Parent\{s, s^{'}\}$. This means s is a root node in S.
3. If n is a parent of s then there is a relation as $n^{'} \rightarrow^O n$. If there is node $n^{''}$ such that there are two relations $Parent\{n, n^{''}\}$ and $Parent\{n^{''}, s\}$ then $n^{''}$ is also defeated. All immediate parent nodes are defeated.
4. $\nexists n \in T$ such that $n \rightarrow^O s$. The root of S remains undefeated or unchallenged.
5. $\nexists n^{''} \in T$ such that $n^{''} \rightarrow^O s''$. Where $s^{''} \in S$ and there is a relation $child\{s^{''}, s\}$. Every child node of s (also belonging to S) remains undefeated or unchallenged.

A part of a dialogue tree becomes dead when its root is defeated or some child node is defeated. The maximal subset is D of T such that :

1. There is a node n in D, such that conclusion of n is l^C or there is relation $n \rightarrow^O n^{'}$ where $n^{'}$ is a rule with conclusion l^C. This means the root is defeated and it either contains the critical literal or directly opposes the critical literal. OR
2. $\forall n_i \in \{D/n\}$, such that there is a relation as $Parent\{n, n_i\}$. n is the root node. AND
3. There is a relation $m \rightarrow^O n_j$ for all rules $n_j \in D/n$ and $m \in T$. This means a child node of n is defeated. AND
4. $\nexists k \in \{D/n_j\}$ such that there is a relation as $Parent\{n_j, k\}$. This means n_j is the lowest node in the tree D.

A dialogue game tree can be divided into active, dead and sleepy parts. As the game progresses arguments and counter arguments are presented. This will expand and collapse these partitions of a dialogue game tree. Next we will consider the transition between these partitions.

Active to dead and sleepy tree: An *active* tree becomes a *dead* tree if it is attacked by counter arguments and it may produce *sleepy* trees. Given an *active* tree M, if the current player plays a set of rules R as $R_1 \ldots R_m$ and D denotes the dead tree (initially empty set) then :

1. $\forall m \in M, \exists r \in R$ such that there is a relation $r \rightarrow^O m$ then $\forall m^{'} \in M$: $Parent\{m^{'}, m\}$, $m^{'}$ and m is added to D. A defeated node and its parent nodes becomes a dead tree.
2. $\forall r^{'} \in M$ where $\exists Child\{r^{'}, r\}$ becomes a sleepy tree if there is no rule $m^{'} \in M$ such that there is a relation $m^{'} \rightarrow Or^{'}$.
3. R becomes the current active tree.

Dead to active tree: Given a dead tree D and the current active tree M if the current player (who owns the dead tree) presents a set of rules R as $R_1 \dots R_m$ then the transition is as follows :

1. If $\exists m \in M$ such that $m \rightarrow^{O} d$ where $d \in D$ and there is a current rule $r \in R$ such that $r \rightarrow^{O} m$.
2. $\forall d_i \in D$ such that $Parent\{d_i, d\}$,
   (a) There is no rule m_i in M such that $m_i \rightarrow^{O} D_i$ or
   (b) If there is a rule m_i in M such that $m_i \rightarrow^{O} D_i$ then there is a rule $r_i \in R$ such that $r_i \rightarrow^{O} m_i$.
3. Upon satisfaction of the above conditions a dead node becomes an active tree and a current active tree will have rules as
   (a) All recovered rules and undefeated or recovered parent rules along with new rule set R.

Sleepy tree to active tree: Given a dead tree D, current active tree M and current set of proposed rules R, a sleepy tree S becomes part of an active tree if :

1. for a rule $s_i \in S$ and $d_i \in D$ if $Parent\{d_i, s_i\}$ and d_i is recovered by R (as in the previous transition)
2. for a rule $s_i \in S$ and $r_i \in R$ if there is relation $s_i \rightarrow^{S} r_i$.

We can represent the above transitions in Figure 7.3.

An instance of a game G is represented by $I_i =< AT, DT, ST >$, where i is the current step, AT is the current active tree, DT is the dead tree(s), and ST are the sleepy trees.

A history of a game G at step i is represented by a set of instances of the game from the start of the game. History is represented as $H_i =< I_1 \dots I_{i-1} >$.

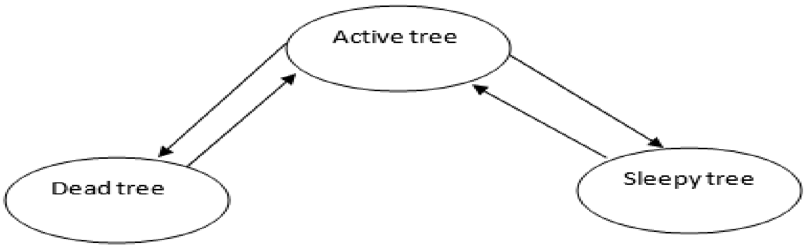

FIGURE 7.3: Transition of dialogue trees

7.4 Agent's knowledge-base

An agent's knowledge base is represented as a graph $G =< N, E >$ with N nodes and E edges where $E \subset N \times N$.

Each node represents a rule. An edge represents a 'support' or 'oppose' relation between two nodes. A node can participate in more than one edge. We represent an attack relation as $E^O\{n_1, n_2\}$ and support relation as $E^S\{n_1, n_2\}$. The agent's knowledge base will have both support and oppose relations for a node. A path in a graph representing the knowledge base of an agent is defined as follows :

A path in a graph G will be represented as $P\{n_1, n_2\} = \{E_1(n_1, n_1^{'}), E_1^{'}(n_1^{'}, n_2^{'}), \ldots E_2(n_k^{'}, n_2)\}$.

Let $V(P)$ represent a set of nodes in the path P and $E(P)$ represent a set of edges in the path P. Given a path $P\{n_1, n_2\}$, n_1 will be called the start node and n_2 will be called the end node. A set of paths originating from a node n_1 is represented as $P\{n_1, *\}$. Given two paths p and $p^{'}$, $p \cap p^{'}$ will represent the common path $p^{''}$.

A path p is a subpath of a path P if (1) $V(p) \subset V(P)$ (2) $E(p) \subset E(P)$.

Given a graph $G =< N, E >$ of a knowledge-base, a line of argument (LA) for an node n:

- P_n is a path P which has start end as n.
- There must not be any oppose edges or even number of oppose edges.

Given a line of argument $LA(n)$, if n_i and n_{i+1} share the first occurance of an oppose edge then :

- All nodes n_j with $Parent\{n_j, n_i\}$ relation with n_i and n_i will be the rules presented by the agent and
- Rules from n_{i+1} to next oppose the relation will be presented by the opponent.

Note that a node can have multiple lines of arguments.

> Given a graph $G =< N, E >$ for a knowledge base, a line of attack (LATTK) for a node n is a path, with the start node as n and the immediate next node of n shares an oppose relation with n and all other nodes share a support relation.

Note that a node can have multiple attack nodes. A line of argument may be protected by a number of lines of attacks. A line of argument becomes dead when there is no available line of attack to protect it. An agent starts a dialogue game (with topic l^C) with a finite set of lines of arguments for l^C and a finite number of lines of attack.

A defendant who is trying to prove l^C can start a new active tree if the current dead tree belongs to it and the defendant can not recover it. A challenger, who is trying to prove that it is not l^C can have new attacks on the same active tree (belonging to the defendant). A similar approach is followed in [25], [51]. [19] also proposes a similar approach where the responsibility of players is different. So the semantics of the dialogue game protocol will impose some rules such as: (1) agents can not repeat an argument (2) an agent can not contradict what it has said before (3) the defendant can use multiple lines of argument (4) the challenger can use multiple lines of attack on the same line of argument. These rules will cause the removal of some nodes from the agent's knowledge-base as they are already played or can not be played. To capture the current available rules to an agent for a dialogue game we introduce mental state of an agent as follows :

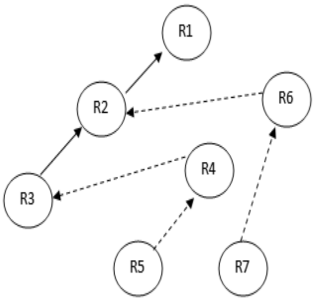

Figure 7.4: Knowledge-base

Given a dialogue game step i, an agent's (Ag_j) mental state is represented by $MS^i_{Ag_j} =< \sum LA^i, \sum LATTK^i >$.

$\sum LA^i$ represents the current line of argument (if not dead) and available line of argument and $\sum LATTK^i$ represents available lines of attack. Consider an example. Let an agent's knowledge-base have the following rules.

$R1 : B \Rightarrow A$; $R2 : C \Rightarrow B$; $R3 : D \Rightarrow C$
$R4 : E \Rightarrow \neg D$; $R5 : F \Rightarrow \neg E$; $R6 : G \Rightarrow \neg C$

Agent's knowledge-base is illustrated in Figure 7.4. Here the line of argument is the path $LA_1 =< R1, R2, R3 >$ where conclusion of $R1$ is l^C. LA_1 is protected by two lines of attack as $LATTK_1 =< R2, R6, R7 >$ and $LATTK_2 =< R3, R4, R5 >$. Note that lines of attack $R2, R6$ and $R3, R4$ are supposed to be played by the challenger.

The dynamics of mental states of agents and game instance is illustrated in figure 7.5 and 7.6. In Figure 7.5, the game instance is $< AT^1 = \{R1, R2, R3\}, DT = \emptyset, ST = \emptyset >$. Figure 7.6 shows the game instance at step 3 after the challenger argues with R4 and the defendant presents counter argument R7. As agents use line of attacks, corresponding rules are removed from the agent's knowledge base.

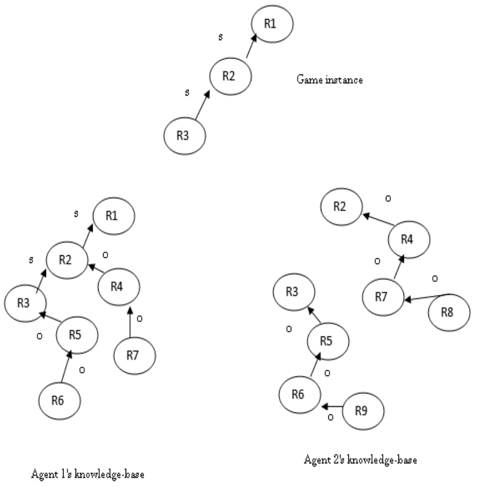

FIGURE 7.5: Game at step1

7.5 Components of a dialogue game protocol

One of our goals in this chapter is to show the dialogue game protocol's limitation in terms of what an agent wants to convey and what it can communicate using a semantics of a protocol. We will use communication theory to model this limitation of dialogue game protocol.

[3] had formalized intentionality in communication. If a speaker S utters χ to 'convey' p to receiver R, then S must intend the following:

1. S's action χ will produce a certain response a in a certain receiver.

2. R to recognize S's intention.

3. R's recognition of S's intention affects R's reasoning to produce a.

In [3] the objective was to model unconventional methods of communication, so intention 2 and 3 were necessary. [2] has shown that A's intention 2 (and hence 3) is not necessary as this intention can be just a logical consequence if the agents communicate using a language (which may be ambiguous). We will follow the communication theory developed in [2] to model this ambiguity. Let A_1 and A_2 be two agents and they share a meaning function $M(\chi)$, where χ is the content of the utterance. Let A want to convey p to B by uttering χ. The communication is as follows :

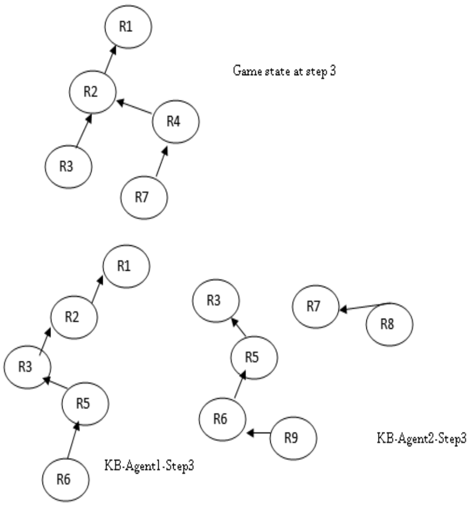

Figure 7.6: Game at step 3

1. A_1 intends to convey p

2. A_1 utters χ to A_2.

3. 2 becomes the common knowledge.

4. A_2 intends to interpret χ.

5. 4 becomes common knowledge.

6. B interprets χ.

7. $P \in M(\chi)$.

8. 7 is common knowledge of both agents.

Success of communication is modeled using game theory. If there is a Nash-Pareto equilibrium (unique solution) then A successfully communicates with B. In this chapter we are only interested in showing ambiguity in a dialogue game. A language becomes ambiguous when it contains a meaning function which can produce several meaning for a certain utterance (including the correct one). Dialogue game protocols will also have the same ambiguity. As an agent has a choice of locutions to utter in response to a particular locution it creates

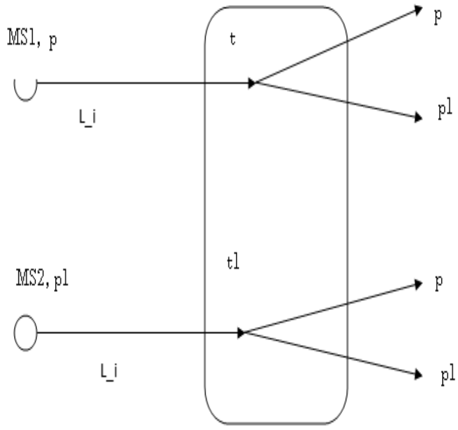

Figure 7.7: Ambiguity in dialogue game

ambiguity. A 'meaning' function in a dialogue game protocol will correspond to a function that maps a locution and content of that locution to a set of agent's (speaker) 'intentions'. A speaker's intention is a certain mental state the speaker wants to create in hearer's mind so that the hearer can produce a certain response (another locution and content of the locution). But as the meaning function can produce several speaker intentions, the receiver can interpret a locution in different ways which will create the ambiguity. As we will see, semantics of a dialogue game protocol will be responsible for the meaning function and hence for this ambiguity.

We can visualize this ambiguity in Figure 7.7. Let there be two agents A (speaker) and B (hearer). Two initial nodes MS1 and MS2 represent two mental states of A. If A is in mental state MS1 then it wants to convey p to B and if it is in MS2 then it wants to convey $p1$ to agent. Here p and $p1$ correspond to mental states MS_p and MS_{p1} in B's mind respectively. A utters the same locution L_i for both situations. A knows that it is in mental state MS1 and want to convey p. But B does not. Upon uttering L_i, A reaches either of two states t or t1. t and t1 are situations when there is a new state in the game and a new mental state of A because of uttering L_i. Now B's interpretation can produce p or p1 for both t and t1.

We can interpret the above situation in Figure 7.5 and 7.6 as follows: $Agent_1$'s mental states are $MS1 = \{R1, R2, R3, R5, R6\}$ and $MS2 = \{R1, R2, R3, R4, R7\}$. If $Agent_1$ is in MS1 then it wants to bring a mental state $P1 = MS3 = \{R7, R8\}\{R3, R5, R6, R9\}$(after step 3) in $Agent_2$ and if it is in MS2 then it wants to bring a mental state $P2 = MS4 = \{R6, R9\}\{R2, R4, R7, R8\}$ in $Agent_2$. Assume that $Agent_1$ is in MS1 and wants to convey P1. $Agent_1$ utters $L_i\{R1, R2, R3\}$, and as $Agent_2$ does not know $Agent_1$'s initial mental state it can not distinguish between t and t1. Here t={R2, R4, R7} and t1={R3, R5, R6}. So in both situations $Agent_2$ can choose any response between $MS3$ and $MS4$.

Next we will model the semantics of locutions and combination rules. Semantics of locution is developed in [35], which uses semantics of FIPA ACL ([4]). It uses feasibility precondition (precondition) and rational effect (post condition) to model semantics of location. For our purpose in this chapter we want to develop semantics on the basis of how a protocol can convey the mental state of agents. We will follow the component classification of dialogue game by [33]. We will only discuss the semantics of locution as the design of the other components will not be changed.

7.5.1 Locutions

Locutions are legal dialogue moves such as Assert, Propose, Reject and Justify. Semantics of locution is as follows:

> Let the current step of a dialogue game be $i + 1$. The mental states of two agents are $MS^i_{Ag_1}$ and $MS^i_{Ag_2}$, the current game instance G^i and it is Ag_1's turn to produce an argument. Ag_1 utters $L_k(R_1 \ldots R_m)$(a locution and content of the locution) to change the current game instance to $G^{i+1}_{L_k}$ so that it can change Ag_2's certain mental state to $MS^{i+1}_{Ag_2}$ so that Ag_2 will make a certain move as $L_{k^1}(R_1 \ldots R_n)$ and the game instance will be changed to G^{i+2}.

The above definition of locution can be understood using Figure 7.9. Agent A is the speaker (first) and B is the hearer. A can be in two mental states namely MS1 and MS2. In MS1, A wants to convey p1(corresponds to MS3 in B) and in MS2 A wants to convey p2 (corresponds to MS4 in B). A knows that it is in MS1 and wants to convey p1. But B does not know that, so A's utterance L1 can correspond to both MS1 and MS2. As A utters L1 it creates two states T1 and T2, corresponding to A's mental state after uttering L1. B can not differentiate between these two state and chooses any one state. So B reaches either state MS3 or MS4. This new mental state corresponds to B's interpretation of L1 to p1 or p2. Now at two mental states $T1$ and T2, B can have a choice of locutions (this choice will be restricted by combination rules). Let B have a choice namely L2 or L3. Let L2 be the locution that follows A's intention. By uttering L2 or L3, B changes the game state to GS1 and GS2. So GS1 is the game state when the locution uttered by A is correctly interpreted by B. Here we have the following ambiguities which will prevent a locution from functioning properly :

- The first ambiguity can arise from the choice between T1 and T2. This means the hearer is uncertain about the speaker's mental state as same the locution can correspond to multiple mental states.

- Due to the first ambiguity, B can not decide whether to choose P1 or P2, so this is the second ambiguity and creates two mental states namely MS3 and MS4. So, unable to distinguish between the speaker's mental states, the hearer will also be unable to distinguish the mental state the speaker wants to bring in the hearer's mind.

- The third ambiguity is due to availability of locutions. There might be several choices of location to an agent at a step and each locution will create a new game state (which may not be the game state the speaker intends).

Now we can use the structure of the dialogue game tree and agent's knowledge base to describe a locution. Let, at step i+1 the game instance be $G^i =< AT^i, DT^i, ST^i >$. Ag_1's mental state is $MS^i_{Ag_1} =< LA^i, LATTK >$ and Ag_2's mental state is $MS^i_{Ag_2} =< LA^i, LATTK >$. Let it be Ag_1's turn to produce an argument. So the current active tree will not belong to Ag_1 and Ag_1 will try to revive a dead tree (here we assume that Ag_1 is the defendant so it owns a line of argument, if it is the challenger then it will try to attack a line of argument hence it will try to convert an active tree to a dead tree). Now, Ag_1 can only revive a tree if it has any available attack or it will start a new line of argument (if any left). This process is described as follows:

1. Let $V\{AT^i\}$ be nodes of a current active tree. If there is a node v in $V\{AT^i\}$ such that it is a root node of Ag_1's line of attack (let it be $LATTK^k$), then Ag_1 will utter $L_j\{LATTK^k\}$.

2. By uttering $L_j\{LATTK_1^k\}$ Ag_1 will change the game state to G^{i+1} and it will try to convey to Ag_2 about Ag_1's intended mental state $MS_{Ag_2}^{i+1}$.

3. $MS_{Ag_2}^{i+1}$ will produce $L_{j1}\{LATTK_2^k\}$ and the game state will change to G^{i+2} and Ag_1's mental state will change to $MS_{Ag_2}^{i+2}$.

4. If the attempt to convey Ag_2 about $MS_{Ag_2}^{i+1}$ is successful then following will be true :

   - The reduced path in Ag_1's knowledge base will be added as expansion of game state tree.
   - The reduced path in transformation from $MS_{Ag_2}^{i+1}$ to $MS_{Ag_2}^{i+2}$ will be the extension of the game state tree.

So we can say that if communication was successful then the game state tree will take the shape of Ag_1's initial mental tree. Again we can go back to figure 7.5 and 7.6 for illustration. Let $Agent_1$ be in mental state $MS1 =< R1, R2, R3, R4, R7 >$ and by uttering $L\{R1, R2, R3\}$ it means to bring a mental state $MS3 =< R7, R8 >$. As we can see from these figures $Agent_2$ correctly interprets $Agent_1$'s locution as the final game state is the same as the initial mental state of $Agent_1$.

7.5.2 Combination rules

Combination rules decide which locutions are applicable in a particular stage of the game. Applicable locutions are determined by the history of the game. Let the history of the game at step i be a collection of states as $H^i = S^1 \dots S^{i-1}$ and the set of locutions in a dialogue game protocol is L. So the combination rules can be represented as :

$$H^i \times L \to L^{'} | L^{'} \subset L$$

The set of locution $L^{'}$ is applicable locutions at stage $i + 1$ as illustrated in Figure 7.10.

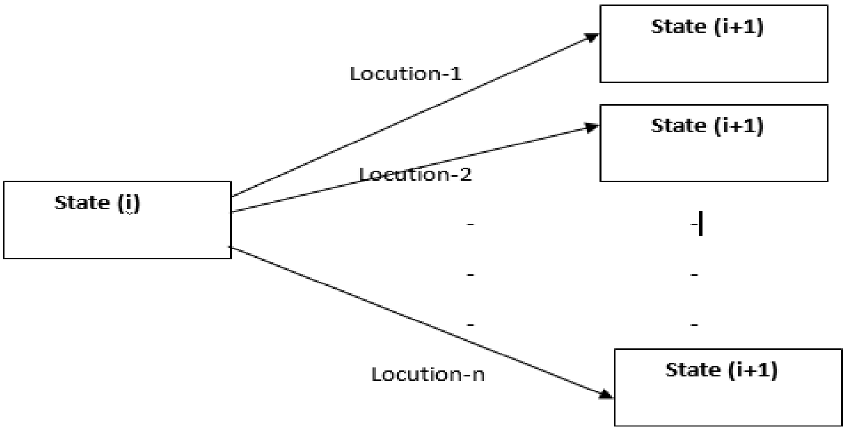

Figure 7.8: Locutions: Given a dialogue state different locutions can produce different states

7.6 Conclusion and future work

In this chapter we provide a new semantics for locutions of a dialogue game. We have shown how misinterpretation can happen in dialogues. Our next research goal is to find an optimum combination of locutions that will reduce the chance of misinterpretation. In a similar problem, [2] applied game theory and an unique solution to the game means an optimum solution. So if there is one locution can represent the best response to a locution then the dialogue game will be optimal.

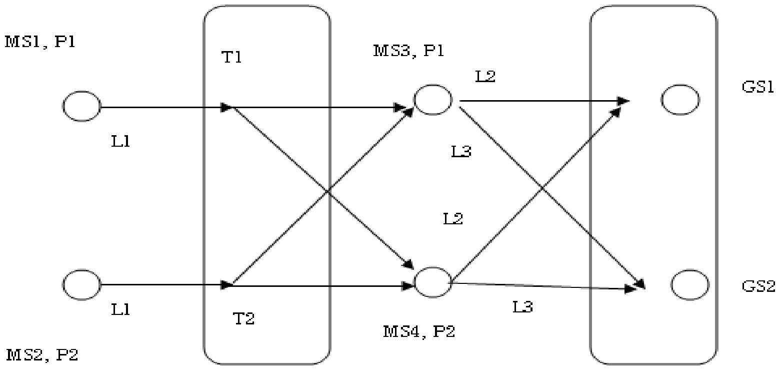

Figure 7.9: Locutions

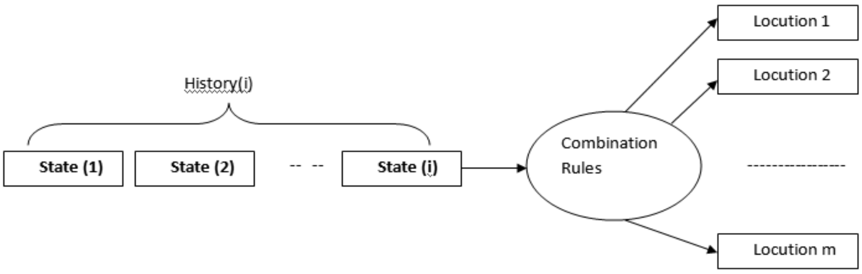

Figure 7.10: Combination rules

8
Future work

We have proposed two protocols for dialogue games. These protocols project the strategic behaviour of agents as agents have to decide which arguments to attack in a given time frame. Possible future works base a on these protocols are as follows :

1. Modeling logical properties as termination rules, complexity, fairness and completeness of dialogue game protocols.

2. Comparing protocols on logical properties.

3. Open knowledge system: In our protocols agents enter the dialogue game with fixed knowledge. But it is unrealistic to assume that an agent's knowledge will not change during the dialogue game. So dialogue game protocols can be modified to support an open knowledge base.

We have proposed a new semantics for locutions. Possible future works based on this new semantics of locution are as follows :

1. Given a set of locutions for a protocol we can identify unambiguous locutions. We can use game theory to find out if there is a unique solution to the game.
2. Given a set of locutions for a protocol a particular set of locutions can be evolutionary stable. We can use evolutionary game theory to find out the sets of locutions which are evolutionary stable.
3. Compare protocols based on the new semantics: Comparison of protocols on this semantics will reveal which protocol can convey the right meaning more efficiently.

References

[1] Guido Boella, Guido Governatori, Joris Hulstijn, Rgis Riveret, Antonino Rotolo and Leemndert van der Torre. FIPA Communicative Acts in Defeasible Logic. *Alankar Karol, Pavlos Peppas and Mary-Anne Williams, editor, Seventh IJCAI International Workshop on Nonmonotonic Reasoning, Action and Change (NRAC'07)*,2007.

[2] Prashant Parikh . The Use of Language. *CSLI publication*,2001.

[3] Paul Grice. Meaning. *The Philosophical Review*,66: 377-88, 1957.

[4] FIPA ACL communicative act library specification. *Technical Report SC00037J*, 2002.

[5] G. Governatori, M. J. Maher, D. Billington, and G. Antoniou. Argumentation semantics for defeasible logics. *Journal of Logic and Computation*, 14(5):675–702, 2004.

[6] G. Governatori and A. Rotolo. Defeasible logic: Agency, intention and obligation. *Deontic Logic in Computer Science*, LNAI 3065: 114–128. Springer, 2004.

[7] G. Governatori, A. Rotolo, and V. Padmanabhan. The cost of social agents. *Proceeding AAMAS 2006*: 513–520. ACM Press, 2006.

[8] B. N. Grosof. Representing e-commerce rules via situated courteous logic programs in RuleML. *Electronic Commerce Research and Applications*, 3(1):2–20, 2004.

[9] G. Antoniou, D. Billington, G. Governatori, M. J. Maher, and A. Rock. A family of defeasible reasoning logics and its implementation. *Proceedings. ECAI 2000*: 459–463. 2000.

[10] D. Billington. Defeasible logic is stable. *Journal of Logic and Computation*, 3:370–400, 1993.

[11] P. M. Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming, and n–person games. *Artificial Intelligence*, 77:321–357, 1995.

[12] P. M. Dung, R. A. Kowalski and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation, *Artificial Intelligence*, 170: 114-159, 2006.

[13] T. Gordon. The Pleadings Game: An artificial Intelligence Model of Procedural Justice *Journal of Artificial Intelligence and Law*, 2(4), Springer Verlag, the Netherlands, 1993.

[14] I. A. Letia and R. Varic. Defeasible protocols in persuasion dialogues. *Proceedings. WE-IAT'06*, IEEE, 2006.

[15] A. R. Lodder. DiaLaw: On Legal Justification and Dialogical Models of Argumentation. *Artificial Intelligence and Law*, 8(2-3): 265-276(12), 2000.

[16] H. Prakken and G. Sartor. Presumptions and burdens of proof. *Proceedings. Jurix 2006*: 21–30, IOS Press, 2006.

[17] G. Vreeswijk. Representation of formal dispute with a standing order. *Artificial Intelligence Law* 8(2/3): 205-231, 2000.

[18] D. Hitchcock, P. McBurney, and S. Parsons. A framework for deliberation dialogues. *H. V. Hansen, C. W. Tindale, J. A. Blair, and R. H. Johnson, editors, Proceedings of the Fourth Biennial Conference of the Ontario Society for the Study of Argumentation (OSSA 2001)*, 2001.

[19] Jenny Eriksson Lundström, Guido Governatori, Subhasis Thakur and Vineet Padmanabhan. An Asymmetric Protocol for Argumentation Games in Defeasible Logic. *PRIMA 07*, 2007.

[20] P. McBurney and S. Parsons. Games that agents play: A formal framework for dialogues between autonomous agents. *Journal of Logic, Language and Information*, 2002.

[21] Leila Amgoud, Simon Parsons, and Nicolas Maudet. Arguments, dialogue, and negotiation. *ECAI*, pages 338–342, 2000.

[22] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher. Representation results for defeasible logic. *ACM Trans. Comput. Logic*, 2(2):255–287, 2001.

[23] Trevor J. M. Bench-capon. Specification and implementation of toulmin dialogue game. *Proceedings of JURIX 98*, pages 5–20. GNI, 1998.

[24] Frank Dignum, Barbara Dunin-Keplicz, and Rineke Verbrugge. Agent theory for team formation by dialogue. *ATAL '00: Proceedings of the 7th International Workshop on Intelligent Agents VII. Agent Theories Architectures and Languages*, pages 150–166, London, UK, 2001. Springer-Verlag.

[25] Paul E. Dunne and T. J. M. Bench-Capon. Two party immediate response disputes: properties and efficiency. *Artif. Intell.*, 149(2):221–250, 2003.

[26] Andreas Hamfelt, Jenny Eriksson Lundström, and Jørgen Fischer Nilsson. A metalogical formalisation of legal argumentation as game trees with defeasible reasoning. *ICAIL '05: Proceedings of the 10th international conference on Artificial intelligence and law*, pages 250–251, 2005.

[27] J. Hulstijn. Dialogue models for inquiry and transaction. *Phd Thesis.* http://doc.utwente.nl/17913/, 2000.

[28] Nicolas Maudet Leila Amgoud and Simon Parsons. Modeling dialogues using argumentation. *ICMAS '00: Proceedings of the Fourth International Conference on MultiAgent Systems (ICMAS-2000)*, page 31, Washington, DC, USA, 2000. IEEE Computer Society.

[29] Jenny Eriksson Lundström, Andreas Hamfelt, and Jørgen Fischer Nilsson. A rule-sceptic characterization of acceptable legal arguments. *ICAIL '07: Proceedings of the 11th international conference on Artificial intelligence and law*, pages 283–284, 2007.

[30] M.J. Maher, A. Rock, G. Antoniou, D. Billington, and T. Miller. Efficient defeasible reasoning systems. *Tools with Artificial Intelligence, IEEE International Conference on*, 0:0384, 2000.

[31] Peter Mcburney, Rogier M. Van Eijk, Simon Parsons, and Leila Amgoud. A dialogue game protocol for agent purchase negotiations. *Autonomous Agents and Multi-Agent Systems*, 7(3):235–273, 2003.

[32] Peter McBurney and Simon Parsons. Representing epistemic uncertainty by means of dialectical argumentation. *Annals of Mathematics and Artificial Intelligence*, 32(1-4):125–169, 2001.

[33] Peter McBurney and Simon Parsons. Games that agents play: A formal framework for dialogues between autonomous agents. *J. of Logic, Lang. and Inf.*, 11(3):315–334, 2002.

[34] Peter Mcburney and Simon Parsons. Dialogue game protocols. *M-P. Huget (Editor): Communication in Multiagent Systems: Agent Communication Languages and Conversation Policies, LNAI 2650.* Springer, 2003.

[35] Peter McBurney and Simon Parsons. Locutions for argumentation in agent interaction protocols. *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1240–1241, Washington, DC, USA, 2004. IEEE Computer Society.

[36] Peter Mcburney, Simon Parsons, and Mark W. Johnson. When are two protocols the same. *Communication in Multi-Agent Systems, LNAI 2650*, pages 253–268. Springer, 2003.

[37] Peter McBurney, Simon Parsons, and Michael Wooldridge. Desiderata for agent argumentation protocols. *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 402–409, New York, NY, USA, 2002. ACM.

[38] D. Nute. Defeasible logic. *Handbook of Logic in Artificial Intelligence and Logic Programming*, 3:1–42, 1994.

[39] Simon Parsons, Michael Wooldridge, and Leila Amgoud. An analysis of formal interagent dialogues. *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 394–401, New York, NY, USA, 2002. ACM.

[40] John L. Pollock. How to reason defeasibly. *Artif. Intell.*, 57(1):1–42, 1992.

[41] H. Prakken and G. Sartor. Rules about rules: Assessing conflicting arguments in legal reasoning. *Artificial Intelligence and Law.*, pages 331–368, 1996.

[42] H. Prakken and G. Sartor. Presumaptions and burdens of proof. *Jurix 2006.*, pages 21–30, 2006.

[43] Henry Prakken. On dialogue systems with speech acts, arguments, and counterarguments. *JELIA '00: Proceedings of the European Workshop on Logics in Artificial Intelligence*, pages 224–238, London, UK, 2000. Springer-Verlag.

[44] Henry Prakken. Modelling defeasibility in law: Logic or procedure? *Fundam. Inf.*, 48(2-3):253–271, 2001.

[45] Henry Prakken. Relating protocols for dynamic dispute with logics for defeasible argumentation. *Synthese*, 127:187–219, 2001.

[46] Henry Prakken and Giovanni Sartor. Formalising arguments about the burden of persuasion. *ICAIL '07: Proceedings of the 11th international conference on Artificial intelligence and law*, pages 97–106, New York, NY, USA, 2007. ACM.

[47] Bram Roth, Régis Riveret, Antonino Rotolo, and Guido Governatori. Strategic argumentation: a game theoretical investigation. *ICAIL '07: Proceedings of the 11th international conference on Artificial intelligence and law*, pages 81–90, New York, NY, USA, 2007. ACM.

[48] F. Sadri, F. Toni, and P. Torroni. Logic agents, dialogues and negotiation: An abductive approach. *In Proceedings AISB'01 Convention.* AISB, 2001.

[49] Subhasis Thakur, Guido Governatori, Vineet Padmanabhan, and Jenny Eriksson Lundström. Dialogue games in defeasible logic. *Australian Conference on Artificial Intelligence*, pages 497–506, 2007.

[50] Paolo Torroni. A study on the termination of negotiation dialogues. *AAMAS '02: Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pages 1223–1230, New York, NY, USA, 2002. ACM.

[51] Gerard Vreeswijk and Henry Prakken. Credulous and sceptical argument games for preferred semantics. *JELIA '00: Proceedings of the European Workshop on Logics in Artificial Intelligence*, pages 239–253, London, UK, 2000. Springer-Verlag.

[52] Gerard A. W. Vreeswijk. Abstract argumentation systems. *Artif. Intell.*, 90(1-2):225–279, 1997.

[53] Grigoris Antoniou, David Billington, Guido Governatori and Micheal J. Maher. Embedding Defeasible Logic into Logic Programming. *Theory and Practice of Logic Programming*, 6, 6, pp 803-835. Copyright Cambridge University Press..