Introducing Dialectical Bases in Defeasible Argumentation

Marcela Capobianco* Carlos I. Chesñevar

Grupo de Investigación en Inteligencia Artificial (GIIA)
Departamento de Ciencias de la Computación
Universidad Nacional del Sur
Av. Alem 1253 - (8000) Bahía Blanca - REPÚBLICA ARGENTINA
TEL/FAX: (+54) (291) 459 5135 / 5136
E-mail: {mcapobia,ccchesne}@criba.edu.ar

Abstract

Defeasible argumentation is a form of *defeasible reasoning*, that emphasizes the notion of an *argument*. An argument $\mathcal{A}$ for a conclusion q is a tentative piece of reasoning which supports q. In an argumentative framework, common sense reasoning can be modeled as a process in which we must determine whether an argument *justifies* its conclusion.

The process mentioned above takes considerable computational effort. For this reason it would be convenient to keep a repository of already computed justifications to save work already done with previously solved queries.

In this paper we introduce the concept of *dialectical bases* as a first step in direction to defining a *justification maintenance system* for argumentative frameworks.

KEYWORDS: defeasible reasoning, defeasible argumentation, argumentative frameworks

1 Introduction and motivations

Defeasible argumentation is a form of defeasible reasoning [Pol87], which emphasizes the notion of an *argument* [SL92]. An argument $\mathcal{A}$ for a conclusion q, denoted $\langle \mathcal{A}, q \rangle$, is a tentative piece of reasoning that supports q. In an argumentative setting, commonsense reasoning can be modeled as a process in which an argument $\langle \mathcal{A}, q \rangle$ must *defeat* all of its counterarguments in order to become an *acceptable argument*. During the last decade several alternative frameworks for defeasible argumentation have been developed [Pra93, SL92, Che96, Gar97, Ver96].

In *A Mathematical Treatment of Defeasible Reasoning* [SL92] (MTDR for short), a clear and theoretically sound framework for defeasible argumentation was introduced, which has come to be a fairly standard approach.

In MTDR (as well as in many other argumentative frameworks) the following approach is used to determine acceptability of an argument for a given query q. In order to decide the acceptability of an argument $\langle \mathcal{A}, q \rangle$, possible counterarguments against accepting $\langle \mathcal{A}, q \rangle$ are considered. Since counterarguments are also arguments, they have to be in turn tested for acceptability, resulting in a recursive procedure in which arguments, counter-arguments, counter-counterarguments, and so on, have to be taken into account. Finally those counterarguments against $\langle \mathcal{A}, q \rangle$ which are accepted are compared with A using a

*Supported by a fellowship of the Secretaria General de Ciencia y Tecnología, U.N.S.

preference relation in order to determine if any of them can prevent $\langle \mathcal{A}, q \rangle$ to become acceptable. If $\langle \mathcal{A}, q \rangle$ is finally accepted, then A is said to be a *justification* for q.

Actual implementations of argumentative frameworks involve the features present in most knowledge-based systems, namely:

1. A *knowledge base*, which stores the information an intelligent agent has about the real world.

2. An *inference engine*, which allows the agent to obtain conclusions from the information stored in its knowledge base.

Computing justifications demands considerable computational effort on the inference engine, so it would be convenient to keep a 'repository' of already computed justifications in order to save work already done with previously computed queries. As a result, the agent's ability to solve new queries would be improved. We will call this repository of previously computed justifications *dialectical base*.

Intelligent agents must be able to interact in a dynamic environment since they may learn new facts about the world. Therefore, an argumentative system must be capable of incorporating new information into its knowledge base. Consequently, some of the old justifications maintained by the system in the dialectical base may become invalid. The key problem is to decide which elements of the dialectical base are affected after adding new information and to define how to accommodate those changes.

In this paper we introduce the concept of dialectical bases, defining an extension of the argumentative system MTDR [SL92, SCG94b] which basically involves adding a dialectical base to the traditional inference engine to answer queries. It is important to note that the concept of an *argument maintenance system* (AMS) was the first attempt to solve the problem of storing arguments for speeding up inference [GCS93]. This approach became out of date as the original Simari-Loui framework [SL92] evolved into MTDR [SCG94b, SCG94a] by incorporating dialectical concepts. The motivation of this paper was to capture the basic ideas of an AMS under this new, enhanced formalism.

This paper is structured as follows. In section 2 the formal definition of a dialectical base is introduced, and some of its properties are discussed. In section 3 a *dialectical base revision operation* is analyzed. This operation must keep the dialectical base properly updated when new facts are added to the knowledge base. This definition is followed by presenting some definitions and algorithms which help to address this problem. Finally, in section 4 the most important conclusions that were obtained are presented, discussing also some future research lines to be pursued.

2 Dialectical Bases: fundamentals

Next we will define the concept of dialectical bases, which will be used as a basis for extending the MTDR framework by incorporating an *argument-based justification maintenance system*. From now on, we will assume the reader is familiar with the MTDR argumentative framework (see [SCG94b, Che96] for details).

As suggested in the introduction, a dialectical base will act as a repository of justifications computed in the past, containing dialectical trees obtained as answers to previous queries. This allows the system to save work already done. More formally, a dialectical base can be defined as follows:

Definition 2.1. (*Dialectical Base*) Let $(\mathcal{K}, \Delta)$ be a knowledge base and let $\mathcal{T}_{\langle A_i, h_i \rangle}$ denote an acceptable dialectical tree. We will say that the finite set $\mathcal{B}$ of dialectical trees

$$\mathcal{B} = \{\mathcal{T}_{\langle A_1, h_1 \rangle}, \mathcal{T}_{\langle A_2, h_2 \rangle}, \ldots, \mathcal{T}_{\langle A_n, h_n \rangle}\}$$

is a *dialectical base supported by* $(\mathcal{K}, \Delta)$ (denoted $\mathcal{B}_{\langle K,\Delta\rangle}$) if and only if for every $\mathcal{T}_{\langle A_i,h_i\rangle} \in \mathcal{B}_{\langle K,\Delta\rangle}$, $1 \leq i \leq n$, it holds that $(\mathcal{K}, \Delta) \mid\!\sim_{Arg} \mathcal{T}_{\langle A_i,h_i\rangle}$[1] and $\langle \mathcal{A}_i, h_i\rangle$ is a *justification.* ■

Note that according to the previous definition, a dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$ is always a partial argumentative closure of $(\mathcal{K}, \Delta)$ with respect to $\mid\!\sim_{Arg}$.

2.1 Constructing the dialectical base

Once the notion of a dialectical base is formally defined, we need to specify the way in which new elements are going to be incorporated into that base, as well as the role of the dialectical base into the query answering process. The analysis that follows will be focused on the first of these aspects. For the latter we refer the reader to section 2.2, where the process of answering a query using the dialectical base is discussed.

A reasonable strategy for constructing a dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$ consists in adding justifications to $\mathcal{B}_{\langle K,\Delta\rangle}$ every time the system obtains a new justification to a query q performed by the user. We will assume that before any interaction with the user takes place, the dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$ will be empty. When a query q is successfully solved, obtaining its associated dialectical tree $\mathcal{T}_{\langle A,q\rangle}$, the current dialectical base will be *extended* into $\mathcal{B'}_{\langle K,\Delta\rangle} = \mathcal{B}_{\langle K,\Delta\rangle} \cup \{\mathcal{T}_{\langle A,q\rangle}\}$. This process of adding dialectical trees to the dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$ will continue as long as different successful queries are solved.

As a consequence, the dialectical base will represent a "pool" of dialectical trees associated with queries that were previously justified. Our goal is to allow faster answer to the queries solved in the past. We will discuss this subject in more depth in the next section.

2.2 Using the dialectical base to answer queries

When the argumentative system is given a query q that was already solved in the past, the dialectical base $\mathcal{B}_{\langle \mathcal{K},\Delta\rangle}$ helps to speed up the inference process. First of all, the inference engine will try to solve q according to the information stored in $\mathcal{B}_{\langle \mathcal{K},\Delta\rangle}$. If q cannot be justified from $\mathcal{B}_{\langle K,\Delta\rangle}$, then the usual justification procedure will be started, looking for a dialectical tree from $(\mathcal{K}, \Delta)$ whose root is a justified argument for q

This analysis leads us to defining when a literal q is justified with respect to $\mathcal{B}_{\langle K,\Delta\rangle}$. Formally stated:

Definition 2.2. (*Dialectical base justification*) Let $\mathcal{B}_{\langle K,\Delta\rangle}$ be a dialectical base, and let q be a ground literal. We will say that $\mathcal{B}_{\langle K,\Delta\rangle}$ *justifies* q if and only if there is a dialectical tree $\mathcal{T}_{\langle A,q\rangle}$ in $\mathcal{B}_{\langle K,\Delta\rangle}$. ■

The following algorithm shows how to solve a query q in an argumentative system which maintains a dialectical base:

Algorithm 2.1. Solve Query

input: q (a query)
output: $\langle \mathcal{A}, q\rangle$ (a justified argument for q, if any)

```
if B⟨K,Δ⟩ justifies q
   then returns ⟨A,q⟩
   else
      if there exists a justified argument ⟨A,q⟩ wrt (K,Δ)
         then returns ⟨A,q⟩
         else the query can not be justified
```
■

[1]In what follows, we will write $(\mathcal{K}, \Delta) \mid\!\sim_{Arg} \mathcal{T}_{\langle A,h\rangle}$ to denote that $\mathcal{T}_{\langle A,h\rangle}$ is an acceptable dialectical tree which can be obtained from $(\mathcal{K}, \Delta)$ by applying the MTDR inference procedure [SCG94b].

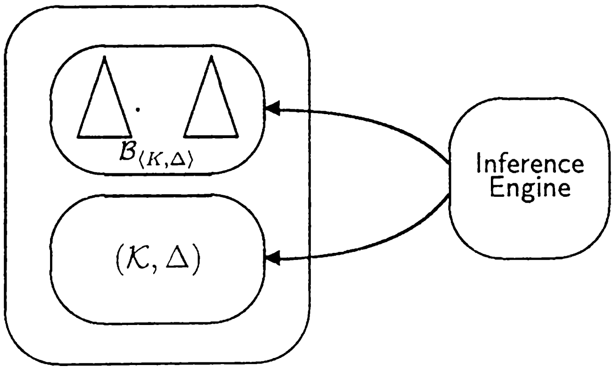

Figure 1: extended MTDR system

Next we will define an *extended* MTDR *framework*, which incorporates a dialectical base.

2.3 Extending the MTDR system with a dialectical base

Now that we have properly defined the concept of a dialectical base, we can state the proposed extension to the original MTDR system. This extension is defined as follows:

Definition 2.3. (*Extended* MTDR *framework*) Let $(\mathcal{K}, \Delta)$ be a MTDR knowledge base. An *extended* MTDR *framework* is composed by:

- An *extended knowledge base* $((\mathcal{K}, \Delta), \mathcal{B}_{\langle \mathcal{K},\Delta\rangle})$ where $\mathcal{B}_{\langle \mathcal{K},\Delta\rangle}$ corresponds to a *dialectical base* supported by $(\mathcal{K}, \Delta)$.
- An *extended inference engine* which behaves according to the algorithm 2.1.

■

The Figure 1 illustrates the structure of our proposed extension. When solving a query q, the system tries to determine in the first place if the dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$ justifies q. If this is the case, q will be justified with respect to $(\mathcal{K}, \Delta)$. In contrast, if $\mathcal{B}_{\langle K,\Delta\rangle}$ does not justify q, then the extended system will try to build an argument for q using the information stored in the knowledge base $(\mathcal{K}, \Delta)$,as the original system does.

3 Revision in Dialectical Bases

It is desirable that a system modeling the behavior of an intelligent agent be capable of acquiring new information in a dynamic way. Therefore, the proposed system provides a mechanism to add a new contingent fact h_c to the non-defeasible knowledge base $\mathcal{K}$, resulting in a new knowledge base $\mathcal{K}' = \mathcal{K} \cup \{h_c\}$ which should be consistent.[2]

Nevertheless, when h_c is added as a new fact to the non-defeasible knowledge $\mathcal{K}$, the justifications obtained in the past stored in $\mathcal{B}_{\langle \mathcal{K},\Delta\rangle}$ may no longer be valid with respect to $(\mathcal{K} \cup \{h_c\}, \Delta)$. To overcome this problem a *revision process* should be performed every time a new fact is added, in order to keep the dialectical base properly updated. We will define an *update* procedure, based on updating each one of the dialectical trees $\mathcal{T}_{\langle A_i,h_i\rangle}$ in a dialectical base $\mathcal{B}_{\langle K,\Delta\rangle} = \{\mathcal{T}_{\langle A_1,h_1\rangle}, \mathcal{T}_{\langle A_2,h_2\rangle}, \ldots, \mathcal{T}_{\langle A_n,h_n\rangle}\}$ with respect to h_c. More formally:

[2]From now on we assume that adding a new fact h to $\mathcal{K}$ results in a new set $\mathcal{K} \cup \{h\}$ such that $\mathcal{K} \cup \{h\} \not\vdash \bot$.

Definition 3.1. (*Update*) Let $\mathcal{B}_{\langle \mathcal{K},\Delta\rangle} = \{ \mathcal{T}_{\langle A_1,h_1\rangle}, \mathcal{T}_{\langle A_2,h_2\rangle}, \ldots, \mathcal{T}_{\langle A_n,h_n\rangle} \}$ be a dialectical base and let h_c be a new contingent fact such that $\mathcal{K} \cup \{h_c\}$ is consistent. An *update of* $\mathcal{B}_{\langle K,\Delta\rangle}$ *with respect to* h_c, denoted $\texttt{Update}(\mathcal{B}_{\langle K,\Delta\rangle}, h_c)$, is defined as follows:

$$\texttt{Update}(\mathcal{B}_{\langle K,\Delta\rangle}, h_c) = \bigcup_{i=1}^{n} \{\texttt{UpdateTree}(\mathcal{T}_{\langle A_i,h_i\rangle}, h_c)\}$$

where $\texttt{UpdateTree}(\mathcal{T}_{\langle A_i,h_i\rangle}, h_c)$ denotes the updating of a dialectical tree in $\mathcal{B}_{\langle K,\Delta\rangle}$. ■

Given a dialectical tree $\mathcal{T}_{\langle A,q\rangle}$ in $\mathcal{B}_{\langle \mathcal{K},\Delta\rangle}$ and a new fact h_c to be introduced into the knowledge base $\mathcal{K}$, two situations may arise:

Argument invalidation: Since $\mathcal{T}_{\langle A,q\rangle}$ is a dialectical tree, all of its nodes must be valid *arguments* [SL92]. The addition of the fact h_c may invalidate some arguments (nodes) in $\mathcal{T}_{\langle A,q\rangle}$, since they must be now consistent and minimal with respect to $(\mathcal{K} \cup \{h_c\}, \Delta)$. As a result, it will be necessary to check every argument associated with the dialectical tree $\mathcal{T}_{\langle A,q\rangle}$ for consistency and minimality using the new, augmented knowledge base $(\mathcal{K} \cup \{h_c\}, \Delta)$.

Dialectical tree invalidation: In addition to being composed by valid arguments, all of the argumentative lines in the dialectical tree $\mathcal{T}_{\langle A,q\rangle}$ must be *acceptable* [SCG94b]. However, after the addition of the fact h_c an acceptable argumentative line may become unacceptable. Thus, in the revision process we must also check for acceptability of the argumentative lines in $\mathcal{T}_{\langle A,q\rangle}$.

It must be also remarked that after adding the fact h_c to the non-defeasible knowledge $\mathcal{K}$ we can obtain new arguments from $(\mathcal{K} \cup \{h_c\}, \Delta)$. These new arguments may be defeaters for some of the stored arguments, which were based on $(\mathcal{K}, \Delta)$, leading to new branches into the existing dialectical tree $\mathcal{T}_{\langle A,q\rangle}$.

The `UpdateTree` procedure must be aware of these issues. In what follows we will discuss each of them in more detail.

3.1 Argument Invalidation

Consistency

As stated before, when we add some new contingent fact h_c to $\mathcal{K}_c$, some of the arguments supported by the old knowledge base may not be consistent with respect to the new knowledge base $(\mathcal{K} \cup \{h_c\}, \Delta)$ and therefore they are no longer considered as arguments.

Thus, after we add a new fact h_c to the knowledge base $\mathcal{K}$ we should verify consistency, with respect to the new knowledge base, for each one of the old arguments stored in the dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$. Checking consistency of existing arguments when adding new contingent information is considered in [GCS93], where an effective procedure for detecting this situation is presented.[3] From now on we will refer to this procedure as a function `Consistent`, defined as follows:

Definition 3.2. Let $(\mathcal{K}, \Delta)$ be a knowledge base, let $\mathcal{A}$ be an argument for h built from $(\mathcal{K}, \Delta)$ and let h_c be a new fact added to $\mathcal{K}$, such that $\mathcal{K} \cup \{h_c\} \nvdash \bot$.

We define the function $\texttt{Consistent}(\mathcal{A}, h_c)$ $\wp(\Delta_{grounded}) \times facts(\mathcal{L}) \mapsto \{\text{TRUE}, \text{FALSE}\}$ as follows:

$$\texttt{Consistent}(\mathcal{A}, h_c) = \begin{cases} \text{TRUE} & \text{if } \mathcal{K} \cup \{h_c\} \cup Co(\mathcal{A}) \nvdash \bot \\ \text{FALSE} & \text{otherwise} \end{cases}$$

[3] We refer the reader to [GCS93] for further details.

where $Co(\mathcal{A})$ denotes the consequents of the rules in $\mathcal{A}$, and $facts(\mathcal{L})$ denotes all the possible facts in the underlying knowledge representation language $\mathcal{L}$. ■

Note that this checking must be performed on every argument $\langle \mathcal{A}_i, h_i \rangle$ belonging to every $\mathcal{T}_{\langle A,h \rangle}$ in $\mathcal{B}_{\langle K,\Delta \rangle}$. This might lead to prune existing dialectical trees in $\mathcal{B}_{\langle K,\Delta \rangle}$. If there is an argument $\langle \mathcal{A}_i, h_i \rangle$ in some dialectical tree $\mathcal{T}_{\langle A,h \rangle}$ which is not consistent with respect to the new knowledge base we must eliminate not only $\langle \mathcal{A}_i, h_i \rangle$ from $\mathcal{T}_{\langle A,h \rangle}$, but also the sub-tree rooted in $\langle \mathcal{A}_i, h_i \rangle$. We must also note that when a sub-tree is eliminated we need to perform some operation in order to ensure the correctness of the resulting dialectical tree. To illustrate this fact let us consider the following example.[4]

Example 3.1. Suppose that $\mathcal{B}_{\langle K,\Delta \rangle}$ is a dialectical base supported by $(\mathcal{K}, \Delta)$, where

$$\mathcal{K} = \{d \leftarrow \texttt{true},\ c \leftarrow h\}$$

and

$$\Delta = \{c \mathrel{-\!\!<} \texttt{true},\ b \mathrel{-\!\!<} \texttt{true},\ a \mathrel{-\!\!<} b,\ \neg b \mathrel{-\!\!<} c,\ \neg c \mathrel{-\!\!<} d\}$$

and suppose that $\mathcal{T}_{\langle A_1,a \rangle}$ is a dialectical tree stored in $\mathcal{B}_{\langle K,\Delta \rangle}$ composed by the arguments:

- $\langle \mathcal{A}_1, a \rangle = \{b \mathrel{-\!\!<} \texttt{true},\ a \mathrel{-\!\!<} b\}$,
- $\langle \mathcal{A}_2, \neg b \rangle = \{c \mathrel{-\!\!<} \texttt{true},\ \neg b \mathrel{-\!\!<} c\}$
- $\langle \mathcal{A}_3, \neg c \rangle = \{\neg c \mathrel{-\!\!<} d\}$.

Where $\langle \mathcal{A}_1, a \rangle$ is the root of the tree, $\langle \mathcal{A}_2, \neg b \rangle$ is a defeater for $\langle \mathcal{A}_1, a \rangle$ and $\langle \mathcal{A}_3, \neg c \rangle$ is a defeater for $\langle \mathcal{A}_2, \neg b \rangle$.

The dialectical tree $\mathcal{T}_{\langle A_1,a \rangle}$ is justified [5] because $\langle \mathcal{A}_2, \neg b \rangle$ (the defeater for $\langle \mathcal{A}_1, a \rangle$), is in turn defeated by $\langle \mathcal{A}_3, \neg c \rangle$ and as a result, $\langle \mathcal{A}_1, a \rangle$ is justified.

However, if we add a new fact h to $\mathcal{K}$ then the argument $\langle \mathcal{A}_3, \neg c \rangle$ is not an argument with respect to $(\mathcal{K} \cup \{h_c\}, \Delta)$, because $\mathcal{K} \cup h \cup \mathcal{A}_3 \mathrel{|\!\!\sim} \bot$. Therefore, $\langle \mathcal{A}_2, \neg b \rangle$ is now undefeated, and hence $\langle \mathcal{A}_1, a \rangle$ is defeated. ■

The previous example shows that the procedure for eliminating the arguments which are not consistent with respect to the new knowledge base, involves also updating the marking of the resulting dialectical tree.

Minimality

An argument must be minimal with respect to the set of rules used in its construction [SL92]. However, when we add a new fact h_c to the knowledge base, the arguments stored in the $\mathcal{B}_{\langle K,\Delta \rangle}$ may no longer satisfy this constrain with respect to $(\mathcal{K} \cup \{h_c\}, \Delta)$. In fact, it may happen that for some argument $\langle \mathcal{A}, q \rangle$ in the $\mathcal{B}_{\langle K,\Delta \rangle}$, we could obtain a new argument $\langle \mathcal{A}', q \rangle$ with respect to $(\mathcal{K} \cup \{h_c\}, \Delta)$, such that $\mathcal{A}' \subset \mathcal{A}$ and $\mathcal{K} \cup \{h_c\} \cup \mathcal{A}' \mathrel{|\!\!\sim} q$. In this case, $\langle \mathcal{A}', q \rangle_{\mathcal{K} \cup \{h_c\}}$ will be called a *reduction* of $\langle \mathcal{A}, q \rangle_{\mathcal{K}}$. Formally:

Definition 3.3. (*Reduction*) An argument $\langle \mathcal{A}_1, h \rangle_{\mathcal{K} \cup \{h_c\}}$ will be called a *reduction* of the argument $\langle \mathcal{A}_2, h \rangle_{\mathcal{K}}$ if $\mathcal{A}_1 \subset \mathcal{A}_2$. ■

Consequently, after we add a new fact to $\mathcal{K}$ we must check the minimality of the previously computed arguments. There exists a procedure for checking the minimality of a set of grounded defeasible rules [GCS93], which can be used in our case. From now on we will refer to this procedure as a function `Minimal` defined as follows:

[4] In our examples, we will use the language of *defeasible logic programming* [Gar97]

[5] We will overload the term "justified" by saying that a dialectical tree is justified iff its root is a justification [SCG94b].

Definition 3.4. Let $(\mathcal{K}, \Delta)$ be a knowledge base, let $\langle \mathcal{A}, h \rangle$ be an argument built from $(\mathcal{K}, \Delta)$ and let h_c be a new fact added to $\mathcal{K}$, such that $\mathcal{K} \cup \{h_c\} \not\vdash \perp$.

The function $\texttt{Minimal}(\langle \mathcal{A}, h \rangle, h_c) \quad \wp\,(Args((\mathcal{K}, \Delta))) \times facts(\mathcal{L}) \mapsto \{\text{TRUE}, \text{FALSE}\}$ is defined as follows:

$$\texttt{Minimal}(\langle \mathcal{A}, h \rangle, h_c) = \begin{cases} \text{TRUE} & \text{if } \langle \mathcal{A}, h \rangle \text{ has no reduction wrt } h_c \\ \text{FALSE} & \text{otherwise} \end{cases}$$

where $Args((\mathcal{K}, \Delta))$ denotes all the arguments which can be built from the knowledge base $\mathcal{B}_{\langle K, \Delta \rangle}$ ■

After adding a new fact to $\mathcal{K}$, we must apply the $\texttt{Minimal}$ function to every argument $\langle \mathcal{A}_i, h_i \rangle$ belonging to every tree $\mathcal{T}_{\langle A,h \rangle}$ in $\mathcal{B}_{\langle K, \Delta \rangle}$. If we find that for some argument $\langle \mathcal{A}_i, h_i \rangle$ in some $\mathcal{T}_{\langle A,h \rangle}$ supported by the old knowledge base $(\mathcal{K}, \Delta)$, $\texttt{Minimal}(\langle \mathcal{A}_i, h_i \rangle, h_c) = \text{FALSE}$, then we must replace $\langle \mathcal{A}_i, h_i \rangle$ for its corresponding reduction with respect to the new added fact h_c. Thus, we need to define a function which maps every argument into its corresponding reduction with respect to a contingent fact h_c. Formally stated:

Definition 3.5. Let $\langle \mathcal{A}, h \rangle$ be an argument supported by $(\mathcal{K}, \Delta)$, and let $(\mathcal{K} \cup \{h_c\}, \Delta)$ be the new knowledge base. We will define the function

$$\texttt{FindReduction}(\langle \mathcal{A}, h \rangle, h_c) \quad \wp\,(Args((\mathcal{K}, \Delta))) \times facts(\mathcal{L}) \mapsto \wp(Args((\mathcal{K}, \Delta)))$$

which given an argument $\langle \mathcal{A}, h \rangle$ and an added fact h_c returns the reduction of $\langle \mathcal{A}, h \rangle$ with respect to h_c. ■

The function $\texttt{FindReduction}$ can be computed in the following way: if $\langle \mathcal{A}, h \rangle$ is not affected by the introduction of h_c, then $\texttt{FindReduction}(\langle \mathcal{A}, h \rangle, h_c) = \langle \mathcal{A}, h \rangle$. In spite of this, if $\langle \mathcal{A}, h \rangle$ is affected by h_c, we can compute $\texttt{FindReduction}(\langle \mathcal{A}, h \rangle, h_c)$ by eliminating the defeasible rules in $\mathcal{A}$ which are no longer needed to conclude h [GCS93]. Note that $\texttt{FindReduction}(\langle \mathcal{A}, h \rangle, h_c)$ always exists, but in some cases (for instance when the new added fact is h) it can result in the empty argument.

Is important to mention that when we replace a stored argument $\langle \mathcal{A}_i, h_i \rangle$ from a dialectical tree $\mathcal{T}_{\langle A,h \rangle}$, for an argument $\langle \mathcal{A}'_i, h_i \rangle$, where $\texttt{FindReduction}(\langle \mathcal{A}_i, h_i \rangle, h_c) = \langle \mathcal{A}'_i, h_i \rangle$, we should apply some procedure on $\mathcal{T}_{\langle A,h \rangle}$ in order to ensure the correctness of the resulting dialectical tree. The following example illustrates this fact.

Example 3.2. For instance, let $\mathcal{T}_{\langle A_1, \neg a \rangle}$ be a dialectical tree stored in $\mathcal{B}_{\langle K, \Delta \rangle}$, where

$$\mathcal{K} = \{d \leftarrow \texttt{true},\ e \leftarrow \texttt{true}\}$$

and

$$\Delta = \{\neg a \prec d,\ a \prec h,\ h \prec c,\ c \prec \texttt{true},\ \neg c \prec e\}$$

and $\mathcal{T}_{\langle A_1, \neg a \rangle}$ is composed by the followings arguments:

- $\langle \mathcal{A}_1, \neg a \rangle = \{\neg a \prec d\}$
- $\langle \mathcal{A}_2, a \rangle = \{a \prec h,\ h \prec c,\ c \prec \texttt{true}\}$
- $\langle \mathcal{A}_3, \neg c \rangle = \{\neg c \prec e\}$

where $\langle \mathcal{A}_1, \neg a \rangle$ is the root of the tree, $\langle \mathcal{A}_2, a \rangle$ is a defeater for $\langle \mathcal{A}_1, \neg a \rangle$ and $\langle \mathcal{A}_3, \neg c \rangle$ is a defeater for $\langle \mathcal{A}_2, a \rangle$. Clearly $\langle \mathcal{A}_1, \neg a \rangle$ is a justification with respect to $(\mathcal{K}, \Delta)$.

If a new fact h is added to $\mathcal{K}$, the argument $\langle \mathcal{A}_2, a \rangle$ is no longer minimal and as a consequence, we must replace it for its corresponding reduction with respect to h, $\langle \mathcal{A}'_2, a \rangle = \{a \prec h\}$. We must note that when we replace $\langle \mathcal{A}_2, a \rangle$ for $\langle \mathcal{A}'_2, a \rangle$ in $\mathcal{T}_{\langle A_1, \neg a \rangle}$, $\langle \mathcal{A}_3, \neg c \rangle$ (which was before a defeater for $\langle \mathcal{A}_2, a \rangle$) is not a defeater for $\langle \mathcal{A}'_2, a \rangle$, and as a result $\langle \mathcal{A}'_2, a \rangle$ is now undefeated with respect to $(\mathcal{K} \cup \{h\}, \Delta)$. Thus $\langle \mathcal{A}_1, \neg a \rangle$ becomes a defeated argument. ■

3.2 Dialectical tree invalidation

After we add a new fact h_c to $\mathcal{K}$, a dialectical tree $\mathcal{T}_{\langle A,h\rangle}$ stored in a dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$ may no longer remain valid, even the arguments in $\mathcal{T}_{\langle A,h\rangle}$ remain consistent and minimal. This situation obeys to one of the following causes: invalidation of some of the tree argumentative lines in $\mathcal{T}_{\langle A,h\rangle}$, or generation of new conflicting arguments using the new fact. In what follows we discuss these two situations, and propose solutions to these problems.

Argumentative line invalidation

As we mentioned above, in addition to being composed by valid arguments, all of the argumentative lines in a dialectical tree must be *acceptable* [SCG94b] in order to make acceptable the dialectical tree.

For an argumentative line to be acceptable, it must (among other things) be composed by concordant arguments. When we add a new fact to $\mathcal{K}$, some of the acceptable argumentative lines may become unacceptable, because some of its arguments may not remain concordant.

Example 3.3. Consider the following situation: let $\mathcal{T}_{\langle A_1,\neg a\rangle}$ be a dialectical tree stored in a dialectical base $\mathcal{B}_{\langle K,\Delta\rangle}$, where

$$\mathcal{K} = \{d \leftarrow \texttt{true},\ a \leftarrow \neg c \wedge h\}$$

and

$$\Delta = \{b \prec \texttt{true},\ c \prec \texttt{true}, \neg a \prec b,\ \neg b \prec c,\ \neg c \prec d\}$$

and suppose that $\mathcal{T}_{\langle A_1,\neg a\rangle}$ is composed by three arguments,

- $\langle A_1, \neg a\rangle = \{b \prec \texttt{true},\ \neg a \prec b\}$,
- $\langle A_2, \neg b\rangle = \{c \prec \texttt{true},\ \neg b \prec c\}$
- $\langle A_3, \neg c\rangle = \{\neg c \prec d\}$.

where the argument $\langle A_3, \neg c\rangle$ defeats $\langle A_2, \neg b\rangle$ which in turn defeats the root, $\langle A_1, \neg a\rangle$. Note that in this situation $\langle A_3, \neg c\rangle$ is undefeated, and thus $\langle A_2, \neg b\rangle$ is defeated and $\langle A_1, \neg a\rangle$ is undefeated (justified).

In this case, if we add a new fact h to $\mathcal{K}$, the argumentative line $\lambda = [\langle A_1, \neg a\rangle, \langle A_2, \neg b\rangle, \langle A_3, \neg c\rangle]$ in $\mathcal{T}_{\langle A_1,\neg a\rangle}$ becomes unacceptable, since the supporting arguments in λ, $\langle A_1, \neg a\rangle$ and $\langle A_3, \neg c\rangle$, are not concordant (*i.e.*, $\mathcal{K} \cup \mathcal{A}_1 \cup \mathcal{A}_3 \mid\!\sim \bot$). As a result $\mathcal{T}_{\langle A_1,\neg a\rangle}$ is no longer an acceptable dialectical tree. ■

To overcome this problem, when a new fact is introduced we will also have to check for acceptability each argumentative line in every dialectical tree stored in the dialectical base. To accomplish this task, each dialectical tree $\mathcal{T}_{\langle A,q\rangle}$ in $\mathcal{B}_{\langle K,\Delta\rangle}$ is traversed using a top-down approach. For each argument $\langle \mathcal{A}_i, q_i\rangle$ in $\mathcal{T}_{\langle A,q\rangle}$, we its consistency is checked with respect to the sequence $\lambda = [\langle \mathcal{A}, q\rangle, \langle \mathcal{A}_1, q_1\rangle \ldots, \langle \mathcal{A}_i, q_i\rangle]$ composed by the arguments that form the path from the root to $\langle \mathcal{A}_i, q_i\rangle$ [6], in the following way:

- if $\langle \mathcal{A}_i, q_i\rangle$ is a supporting argument, then it should hold that $\mathcal{K} \cup \{h_c\} \cup S_\lambda \not\mid\!\sim \bot$, where S_λ denotes the set of supporting argument present in λ.
- if $\langle \mathcal{A}_i, q_i\rangle$ is an interference argument, then it should hold that $\mathcal{K} \cup \{h_c\} \cup I_\lambda \not\mid\!\sim \bot$, where I_λ denotes the set of interference argument present in λ.

[6]Note that λ represents the path from the root to $\langle \mathcal{A}_i, a_i\rangle$ an its present in all the argumentative lines which include $\langle \mathcal{A}_i, a_i\rangle$

If we find that $\langle \mathcal{A}_i, q_i \rangle$ is not consistent with respect to its argumentative line, then we must eliminate the sub-tree rooted with $\langle \mathcal{A}_i, q_i \rangle$ from $\mathcal{T}_{\langle A,q \rangle}$. Note that, as we mentioned before, when a sub-tree is eliminated we need to perform some operation in order to ensure the correctness of the resulting dialectical tree.

New conflicting arguments

When a new contingent fact h_c is introduced into $(\mathcal{K}, \Delta)$ new arguments, which were not possible to obtain from $(\mathcal{K}, \Delta)$, can be built using h_c. These new arguments may be defeaters for the some of the arguments belonging to a dialectical tree $\mathcal{T}_{\langle A,q \rangle}$ stored in the dialectical base $\mathcal{B}_{\langle K,\Delta \rangle}$. Hence, the generation of these new arguments may lead to changes into $\mathcal{T}_{\langle A,q \rangle}$ and as matter of fact, $\mathcal{T}_{\langle A,q \rangle}$ may no longer be a justified dialectical tree.

Example 3.4. Let us consider the following situation. Suppose that $\mathcal{T}_{\langle A_1, \neg a \rangle}$ is a dialectical tree stored in $\mathcal{B}_{\langle K,\Delta \rangle}$, where $\mathcal{K} = \{\ \}$ and

$$\Delta = \{b \prec \texttt{true},\ c \prec \texttt{true},\ d \prec \texttt{true},\ \neg a \prec b,\ \neg b \prec c,\ \neg c \prec d,\ \neg d \prec h\}$$

and $\mathcal{T}_{\langle A_1, \neg a \rangle}$ is composed by the following three arguments:

- $\langle \mathcal{A}_1, \neg a \rangle = \{b \prec \texttt{true},\ \neg a \prec b\}$,
- $\langle \mathcal{A}_2, \neg b \rangle = \{c \prec \texttt{true},\ \neg b \prec c\}$
- $\langle \mathcal{A}_3, \neg c \rangle = \{d \prec \texttt{true},\ \neg c \prec d\}$.

In $\mathcal{T}_{\langle A_1, \neg a \rangle}$ the argument $\langle \mathcal{A}_3, \neg c \rangle$ defeats $\langle \mathcal{A}_2, \neg b \rangle$ which in turn defeats the root, $\langle \mathcal{A}_1, \neg a \rangle$. It goes without saying that $\langle \mathcal{A}_3, \neg c \rangle$ and $\langle \mathcal{A}_1, \neg a \rangle$ are undefeated and $\langle \mathcal{A}_2, \neg b \rangle$ is defeated by $\langle \mathcal{A}_3, \neg c \rangle$.

Suppose that we add a new fact h into $\mathcal{K}$. Thus we can generate a new argument $\langle \mathcal{A}_4, \neg d \rangle = \{\neg d \prec h\}$, such that $\langle \mathcal{A}_4, \neg d \rangle$ defeats $\langle \mathcal{A}_3, \neg c \rangle$. Accordingly, in the dialectical tree constructed with respect to $(\mathcal{K} \cup \{h\}, \Delta)$, $\langle \mathcal{A}_2, \neg b \rangle$ is now undefeated and defeats the root of the tree, $\langle \mathcal{A}_1, \neg a \rangle$. Therefore $\mathcal{T}_{\langle A_1, \neg a \rangle}$ is no longer a justified dialectical tree. ■

This situation enforces the definition of a procedure for updating each one of the dialectical trees stored in $\mathcal{B}_{\langle K,\Delta \rangle}$. That procedure must take into account the generation of the new defeaters, whose introduction may change the marking of some of the stored dialectical trees.

4 Conclusions

We believe that the notion of dialectical bases can improve the performance of existing argumentative frameworks. In this work we have outlined the mains issues to consider in order to incorporate a dialectical base into the MTDR framework.

The complete definition of an argumentative setting extended with a dialectical base demands taking into account several interacting features (argument consistency and minimality, acceptability of argumentation lines, etc), which were discussed in this work.

We contend that formalizing the notion of dialectical bases and studying its related properties will be the first step in direction towards a definition of a justification maintenance system for argumentative frameworks.

References

[Che96] Carlos I. Chesñevar. El problema de la inferencia en sistemas argumentativos: Alternativas para su solución (msc thesis), December 1996.

[Gar97] Alejandro J. García. Programación en lógica rebatible: su definición teórica y computacional (msc thesis), July 1997.

[GCS93] Alejandro J. García, Carlos I. Chesñevar, and Guillermo R. Simari. Making Argument Systems Computationally Attractive. In *Anales de la XIII Conferencia Internacional de la Sociedad Chilena para Ciencias de la Computación*. Universidad de La Serena, La Serena (Chile), October 1993.

[Pol87] John L. Pollock. Defeasible Reasoning. *Cognitive Science*, 11:481–518, 1987.

[Pra93] Henry Prakken. *Logical Tools for Modelling Legal Arguments*. PhD thesis, Vrije University, Amsterdam (Holanda), January 1993.

[SCG94a] Guillermo R. Simari, Carlos I. Chesñevar, and Alejandro J. García. Focusing inference in defeasible argumentation. In *Anales de la Conferencia IBERAMIA'94*. Asociación Venezolana para Inteligencia Artificial, Caracas (Venezuela), October 1994.

[SCG94b] Guillermo R. Simari, Carlos I. Chesñevar, and Alejandro J. García. The role of dialectics in defeasible argumentation. In *Anales de la XIV Conferencia Internacional de la Sociedad Chilena para Ciencias de la Computación*. Universidad de Concepción, Concepción (Chile), November 1994.

[SL92] Guillermo R. Simari and Ronald P. Loui. A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence*, 53:125–157, 1992.

[Ver96] Bart Verheij. *Rules, Reasons, Arguments: formal studies of argumentation and defeat*. PhD thesis, Maastricht University, Holland, December 1996.