Acceptability in timed frameworks with intermittent arguments

Maria Laura Cobo, Diego C. Martinez, and Guillermo R. Simari

Artificial Intelligence Research and Development Laboratory (LIDIA)
Department of Computer Science and Engineering, Universidad Nacional del Sur
Av. Alem 1253 - (8000) Bahía Blanca - Bs. As. - Argentina
{mlc,dcm,grs}@cs.uns.edu.ar
http://www.cs.uns.edu.ar/lidia

Abstract. In this work we formalize a natural expansion of timed argumentation frameworks by considering arguments that are available with (possibly) some repeated interruptions in time, called *intermittent arguments*. This framework is used as a modelization of argumentation dynamics. The notion of acceptability of arguments is analyzed as the framework evolves through time, and an algorithm for computing intervals of argument defense is introduced.

1 Introduction

One of the main concerns in Argumentation Theory is the search for rationally based positions of acceptance in a given scenario of arguments and their relationships. This task requires some level of abstraction in order to study pure semantic notions. Abstract argumentation systems [10, 17, 2] are formalisms for argumentation where some components remain unspecified, being the structure of an argument the main abstraction. In this kind of system, the emphasis is put on the semantic notion of finding the set of accepted arguments. Most of these systems are based on the concept of *attack* represented as an abstract relation, and extensions are defined as sets of possibly accepted arguments. For two arguments $\mathcal{A}$ and $\mathcal{B}$, if $(\mathcal{A}, \mathcal{B})$ is in the attack relation, it is said that argument $\mathcal{A}$ *attacks* $\mathcal{B}$, implying that the acceptance of $\mathcal{B}$ is conditioned by the acceptance of $\mathcal{A}$, but not the other way around.

The simplest abstract framework is defined by Dung in [10]. It only includes a set of abstract arguments and a binary relation of attack between arguments, allowing the definition of several semantic notions. Dung's framework became the foundation of further research, either by extending the formalism, as in [2, 5, 13] or by elaborating new semantics [11, 4].

In recent works [6, 15, 16], the dynamics of an argumentation framework is considered. The main subject of study is how the outcome changes when the set of arguments or the set of attacks are changed. In this paper we are interested in the overall evolution of an argumentation framework through time. The combination of time and argumentation is a novel research line. In [12] a calculus

for representing temporal knowledge is proposed, and defined in terms of propositional logic. This calculus is then considered with respect to argumentation, where an argument is defined in the standard way: an argument is a pair constituted by a minimally consistent subset of a database entailing its conclusion. This work is thus related to [3].

In [7, 8] a novel framework is proposed, called *Timed Abstract Framework* (TAF), combining arguments and temporal notions. In this formalism, arguments are relevant only in a period of time, called its *availability interval*. This framework maintains a high abstract level in an effort to capture intuitions related with the dynamic interplay of arguments as they become available and cease to be so. The notion of *availability interval* refers to an interval of time in which the argument can be legally used for the particular purpose of an argumentation process. In particular, the actions of adding and removing an argument in a framework, as considered in argumentation dynamics, can be naturally interpreted as a temporal availability of that argument. As several arguments are added or removed through time, the classical notion of acceptability of arguments requires a deeper analysis.

Timed abstract frameworks are suitable to model the evolution of argumentation. In this work we formalize a natural expansion of timed argumentation frameworks by considering arguments with more than one availability interval. These are called *intermittent arguments*, and it is a more accurate modelization of argument dynamics. These arguments are available with (possibly) some repeated interruptions in time. Using this extended timed argumentation framework, we analyze the notion of *acceptability* and we introduce a timed structure for arguments, called t-profiles, that allows a refined characterization of classical acceptability when new arguments are available or cease to be so.

This paper is organized as follows. In the next section we recall the notion of time-intervals [7, 8] and the terminology used in this work. In Section 3, our Timed Abstract Argumentation Framework with intermittent arguments is introduced. Acceptability is formalized in Section 4 and a procedural refinement is introduced in Section 5. Finally, conclusions and future work are discussed.

2   Time representation

In order to capture a time-based model of argumentation, we enrich the classical abstract frameworks with temporal information regarding arguments. In this work we use *temporal intervals of discrete time* as primitives for time representation [1, 9, 14], and thus only metric relations for intervals are applied.

Definition 1 *An interval is a pair build from $a, b \in \mathbb{Z} \cup \{-\infty, \infty\}$, in one of the following ways:*

- *$[a, a]$ denotes a set of time moments formed only by moment a.*
- *$[a, \infty)$ denotes a set of moments formed by all the numbers in $\mathbb{Z}$ since a (including a).*
- *$(-\infty, b]$ denotes a set of moments formed by all the numbers in $\mathbb{Z}$ until moment i (including b).*

- $[a, b]$ denotes a set of moments formed by all the numbers in $\mathbb{Z}$ moment i until moment j (including both a and b).
- $(-\infty, \infty)$ a set of moments formed by all the numbers in $\mathbb{Z}$.

The moments a, b are called endpoints. The set of all the intervals defined over $\mathbb{Z} \cup \{-\infty, \infty\}$ is denoted ι.

For example, $[5, 12]$ and $[1, 200]$ are intervals. If X is an interval then X^-, X^+ are the corresponding endpoints (i.e., $X = [X^-, X^+]$). An endpoint may be a point of discrete time, identified by a natural number, or infinite.

We will usually work with sets of intervals (as they will be somehow related to arguments). Thus, we introduce several definitions and properties needed for semantic elaborations.

Definition 2 *Let S be a set of intervals and let i be a moment of time. The exclusion of i from S, denoted $S \ominus i$, is defined as follows:*

$$S \ominus i = \{I : I \in S \land i \notin I\} \qquad \cup$$
$$\{[I^-, i-1] : I \in S \land i \in I, i \neq I^-\} \cup$$
$$\{[i+1, I^+] : I \in S \land i \in I, i \neq I^+\}$$

The exclusion of interval I from set S, denoted as $S \circleddash I$, is recursively defined as follows:

> i. $S \circleddash I = S \ominus I^-$ if $I^- = I^+$
> ii. $S \circleddash I = (S \ominus I^-) \circleddash [I^- + 1, I^+]$ if $I^- \neq I^+$

The difference between sets of intervals, denoted $S_1 \xrightarrow{L} S_2$ is defined as $\Cap(S_1 \circleddash I)$, $\forall I \in S_2$.

Intersection is another relevant operation on intervals. The intersection of two intervals is the interval formed by all the common points in both of them. Its endpoints are the minimal and maximal time points in common.

Definition 3 *Let I_1 and I_2 be two intervals. The intersection is defined as: $I_1 \cap I_2 = [x, y]$ with $x, y \in I_1$ and $x, y \in I_2$ such that there are no $w, z : w, z \in I_1$ and $w, z \in I_2$ with $w < x$ or $y < z$.*

Definition 4 *Let S_1 and S_2 be two sets of intervals. The intersection of these sets, noted as $S_1 \Cap S_2$, is: $S_1 \Cap S_2 = \{I : I = I_1 \cap I_2, I \neq [\ ], \forall I_1 \in S_1, I_2 \in S_2\}$.*

The intersection of two sets of intervals S_1 and S_2 is formed by all the intersections of an interval of S_1 with an interval of S_2. The following definition formalizes a special relation of inclusion for sets of intervals. In order to do that, we define the set of time-points for a set of intervals. If S is a set of arguments, then $tp(S)$ is the set of time-points in S. For instance, if $S = \{[1, 3], [8, 10]\}$ then $tp(S) = \{1, 2, 3, 8, 9, 10\}$.

Definition 5 *Let S_1 and S_2 be two sets of intervals. The timed-inclusion, denoted $\subseteq'$, is defined as $S_1 \subseteq' S_2 = tp(S_1) \subseteq tp(S_2)$.*

The *partition* of a set of arguments *breaks* overlapped intervals into smaller intervals, and then there is no overlapping intervals in a partition. This is formalized as follows.

Definition 6 *Let S be a set of intervals. The partition of S, denoted* $\mathsf{Part}(\mathsf{S})$ *is defined as:*

- $\mathsf{Part}(\mathsf{S}) = S$ *if* $\forall I_1, I_2 \in S, I_1 \cap I_2 = \emptyset$.
- $\mathsf{Part}(\mathsf{S}) = \mathsf{Part}(\mathsf{S} - \{\mathsf{I}_1, \mathsf{I}_2\} \cup \{\mathsf{I}_1 - (\mathsf{I}_1 \cap \mathsf{I}_2), \mathsf{I}_2 - (\mathsf{I}_1 \cap \mathsf{I}_2), \mathsf{I}_1 \cap \mathsf{I}_2\})$, *with* $I_1, I_2 \in S$ *and* $I_1 \cap I_2 \neq \emptyset$

The notion of partition simplifies semantic elaborations, since it discretizes the evolution of the framework according to moments where arguments start or cease to be available.

In the following section we present Timed Abstract Argumentation Frameworks with intermittent arguments.

3   Timed Argumentation Framework

As remarked before, in Timed Argumentation Frameworks the period of time in which an argument is available for consideration is modeled. The formal definition of our timed abstract argumentation framework follows.

Definition 7 *A timed abstract argumentation framework (TAF) is a 3-tuple* $\langle Args, Atts, \mathcal{A}v \rangle$ *where $Args$ is a set of arguments, $Atts$ is a binary relation defined over $Args$ and $\mathcal{A}v$ is the availability function for timed arguments, defined as* $\mathcal{A}v : Args \to \wp(\iota)$.

Example 1 *The triplet* $\langle Args, Atts, \mathcal{A}v \rangle$, *where* $Args = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{E}\}$, *$Atts =$* $\{(\mathcal{B}, \mathcal{A}), (\mathcal{C}, \mathcal{B}), (\mathcal{D}, \mathcal{A}), (\mathcal{E}, \mathcal{D})\}$ *and the availability function is defined as*

$Args$	$\mathcal{A}v$	$Args$	$\mathcal{A}v$
$\mathcal{A}$	$\{[10, 40], [60, 75]\}$	$\mathcal{B}$	$\{[30, 50]\}$
$\mathcal{C}$	$\{[20, 40], [45, 55], [60, 70]\}$	$\mathcal{D}$	$\{[47, 65]\}$
$\mathcal{E}$	$\{(-\infty, 44]\}$		

is a timed abstract argumentation framework.

The framework of Example 1 can be depicted as in Figure 1, using a digraph where nodes are arguments and arcs are attack relations. An arc from argument $\mathcal{X}$ to argument $\mathcal{Y}$ exists if $(\mathcal{X}, \mathcal{Y}) \in Atts$. Figure 1 also shows the time availability of every argument, as a graphical reference of the $\mathcal{A}v$ function. It is basically the framework's evolution in time. Endpoints are marked with a vertical line, except for $-\infty$ and ∞. For space reasons, only some relevant time points are numbered in the figure. The shorthand notation $\mathcal{A}^+$ and $\mathcal{A}^-$ are used to represent, respectively, the set of arguments attacked by $\mathcal{A}$ and the set of arguments attacking $\mathcal{A}$.

An attack to an argument may actually occur only if both the attacker and the attacked argument are available in the framework. An attack in such a condition is said to be *attainable*.
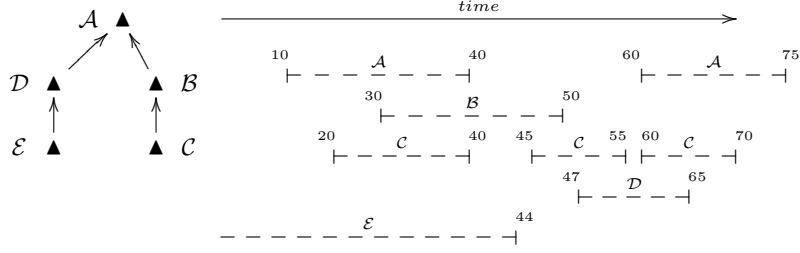
Fig. 1. Framework of Example 1

Definition 8 *Let $\Phi = \langle Args, Atts, Av \rangle$ be a TAF, and let $\{\mathcal{A}, \mathcal{B}\} \subseteq Args$ such that $(\mathcal{B}, \mathcal{A}) \in Atts$. The attack $(\mathcal{B}, \mathcal{A})$ is said to be attainable if $Av(\mathcal{A}) \cap Av(\mathcal{B})$ is not empty. The attack is said to be attainable in $Av(\mathcal{A}) \cap Av(\mathcal{B})$. The set of intervals where an attack $(\mathcal{B}, \mathcal{A})$ is attainable will be noted as $Av(\mathcal{B}, \mathcal{A})$.*

Example 2 *Consider the timed argumentation framework of Example 1. The attacks $(\mathcal{D}, \mathcal{A})$ and $(\mathcal{B}, \mathcal{A})$ are both attainable in the framework. Attack $(\mathcal{D}, \mathcal{A})$ is attainable since $[47, 65]$ overlaps $[60, 75]$ with $[47, 65] \in Av(\mathcal{D})$ and $[60, 75] \in Av(\mathcal{A})$. Attack $(\mathcal{B}, \mathcal{A})$ is attainable since $[30, 50]$ overlaps $[10, 40]$, in $[30, 40]$. Recall that $[30, 50] \in Av(\mathcal{B})$, $[10, 40] \in Av(\mathcal{A})$. The attack $(\mathcal{C}, \mathcal{B})$ is also attainable. Since $Av(\mathcal{C}) = \{[20, 40], [30, 50]\}$ and $Av(\mathcal{B}) = \{[30, 50]\}$ then, we can assure the attainability of the attack by one of the following relations: $[20, 40]$ overlaps $[30, 50]$, $[45, 55]$ overlaps $[30, 50]$. The attack is then attainable at $\{[30, 40], [45, 50]\}$, i.e. $Av(\mathcal{C}) \cap Av(\mathcal{B})$. The attack $(\mathcal{E}, \mathcal{D})$ is not attainable, since $[-\infty, 45]$ does not overlaps $[47, 65]$. The arguments involved in this attack are never available at the same time.*

Since an argument is defended by attacking its attackers, it is important to capture the set of intervals in which an argument is actually defended, as next.

Definition 9 *Let $\Phi = \langle Args, Atts, Av \rangle$ be a TAF, and let $\{\mathcal{A}, \mathcal{B}, \mathcal{C}\} \subseteq Args$ such that $(\mathcal{B}, \mathcal{A}), (\mathcal{C}, \mathcal{B}) \in Atts$. The overlapped availability of the attacks, denoted $Av(\mathcal{C}, \mathcal{B}, \mathcal{A})$, is defined as $Av(\mathcal{C}, \mathcal{B}, \mathcal{A}) = Av(\mathcal{A}) \cap Av(\mathcal{B}) \cap Av(\mathcal{C})$.*

The set $Av(\mathcal{C}, \mathcal{B}, \mathcal{A})$ captures all the intervals in which $\mathcal{C}$ actually defends $\mathcal{A}$ against $\mathcal{B}$. In the following section the notion of defense in timed frameworks is analyzed.

4 Defense through time

An argument may be attacked in several intervals of time. It is defended in these *threat* intervals, only when another argument has an attainable attack to its attacker. Thus, unlike classic frameworks where an argument may be defended or not, here an argument may be defended or not in *some* moments of time. It is not enough to establish if a defense condition is present by looking attacks. It is mandatory to find out *when* these defenses may occur.

Definition 10 *Let $\Phi = \langle Args, Atts, Av \rangle$ be a TAF. The set of defense intervals for $\mathcal{A}$ against $\mathcal{B}$, denoted $\mathcal{D}f(\mathcal{A}, \mathcal{B})$, is defined as: $\bigcup_{\mathcal{X} \in \mathcal{B}^-} Av(\mathcal{A}, \mathcal{B}, \mathcal{X})$.*

For a set of arguments S, the intervals restricted to defenses in S, denoted $\mathcal{D}f(\mathcal{A}, \mathcal{B})_S$, is defined as: $\bigcup_{\mathcal{X} \in \mathcal{B}^- \cap S} Av(\mathcal{A}, \mathcal{B}, \mathcal{X})$.

The set $\mathcal{D}f(\mathcal{A}, \mathcal{B})$ captures all the intervals in which a defense is available, *i.e.* there is an overlapping between $\mathcal{A}$, its attacker $\mathcal{B}$ and every attacker of $\mathcal{B}$.

Now that the set of defense intervals is characterized, the classical notion of acceptability [10] can be addressed. An argument $\mathcal{A}$ is acceptable with respect to a set of arguments S if this set can provide attackers for every attacker of $\mathcal{A}$. In a timed, dynamic context it is natural to consider defenses for an argument $\mathcal{A}$ whenever $\mathcal{A}$ is available, that is, during $Av(\mathcal{A})$.

Definition 11 *Let $\Phi = \langle Args, Atts, Av \rangle$ be a TAF. An argument $\mathcal{A}$ is acceptable with respect to a set of arguments S if $Av(\mathcal{A}, \mathcal{B}) \subseteq' \mathcal{D}f(\mathcal{A}, \mathcal{B})_S$ for every attacker $\mathcal{B}$ of $\mathcal{A}$.*

This definition is close to the classical definition of acceptability. However, in a timed scenario is a very restrictive definition since it is possible for an argument to temporarily be out of defenses.
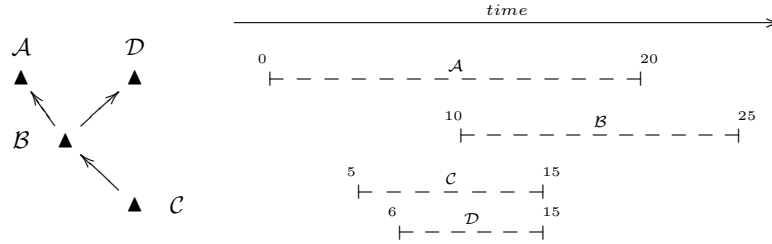

Fig. 2. Argument $\mathcal{D}$ acceptable with respect to $\{\mathcal{C}\}$

Consider the framework of Figure 2. Argument $\{\mathcal{C}\}$ is acceptable with respect to the $\emptyset$. Argument $\mathcal{D}$ is acceptable with respect to $\{\mathcal{C}\}$, since it is defended whenever $\mathcal{B}$ attacks $\mathcal{D}$. However, argument $\mathcal{A}$ is not acceptable with respect to the set $\{\mathcal{C}\}$ since it has no defense against $\mathcal{B}$ in the period $[15, 20]$.

Definition 11 does not take into account the fact that it may be some periods of time in where the argument is actually defended. It would be interesting to have an alternative, more refined definition of acceptability in order to capture proper intervals of defense. Even more, by characterizing specific intervals of provided defense, it is possible to analyze the acceptance of other arguments. Consider the framework depicted on Figure 3. The only argument that is acceptable with respect to $S = \emptyset$ is $\mathcal{C}$. Notice that in the period of time where $\mathcal{E}$ needs defense against $\mathcal{D}$, $\mathcal{A}$ is not attacked so the defense actually takes place. It would be interesting to capture the precise intervals in which an argument is

provided with a defense. For instance, if $S = \emptyset$ the only acceptable argument according to Definition 11 is $\mathcal{C}$. However, the argument $\mathcal{A}$ does not need defenders in $[0, 15]$.
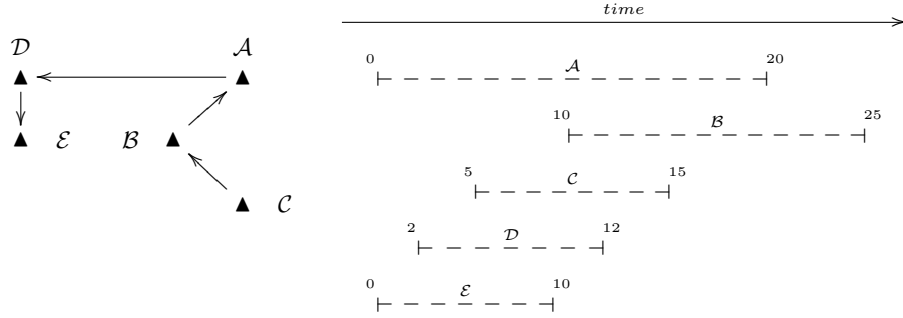

Fig. 3. Framework Simple

A more refined version of acceptability can be provided, but we need a structure to associate arguments and intervals of time. This is called a *t-profile*, and it may have different particular meanings: the availability of the argument, the periods of time in which this argument is attacked, defended, or completely justified under some particular semantics. The definition follows.

Definition 12 *Let $\Phi\langle Args, Atts, \mathcal{A}v\rangle$ be a TAF. A timed argument profile in Φ, or simply* t-profile*, is a pair $[\mathcal{A}, T]$ where $\mathcal{A} \in Args$ and $T \subseteq \mathcal{A}v(\mathcal{A})$.*

The set of all the profiles definable from Φ will be noted as Θ. In the following section we present an algorithm that can be used to compute the set of intervals in which an argument is defended. This is a procedural refinement of acceptability as defined in Definition 11.

5   Computing intervals of defense

Following the general idea of acceptability of arguments, an algorithm is defined for the characterization of the those intervals of time in which a set of t-profiles defend a particular argument.

The algorithm receives an argument X and a set of t-profiles P. There are three different main procedures. The first one captures those moments in time where the argument was previously acceptable or unattacked. The second one determines all the particular defenses that P provides against every attacker in a particular interval. Finally, the intersection of them is calculated to establish *when* the argument is actually defended against all of its attackers in that interval. The last two parts are evaluated for all the intervals that are obtained from the partition of all the availability periods of the attacks against the argument in consideration.

Algorithm 1 DEFENDED PERIODS

Require: $\mathcal{X} \in Args$, P a set of t-profiles, $\Phi = \langle Args, Atts, \mathcal{A}v \rangle$ a TAF
Ensure: D the set of intervals where T defends $\mathcal{X}$.

1: $S = \bigcup_{Y \in X^-} \mathcal{A}v(\mathcal{Y}, \mathcal{X})$
2: **if** $[\mathcal{X}, U] \in P$ **then**
3:      $S = S \stackrel{I}{-} U$
4:      $D = U$
5: **else**
6:      $D = \mathcal{A}v(\mathcal{X}) \stackrel{I}{-} S$
7: **end if**
8: $S = \mathsf{Part}(\mathsf{S})$
9: **for all** $I \in S$ **do**
10:      $\Gamma = \emptyset$
11:      **for all** $\mathcal{Y} \in Args$ such that $\{I\} \subseteq \mathcal{A}v(\mathcal{Y}, \mathcal{X})$ **do**
12:          **for all** $[\mathcal{Z}, U] \in P$ such that $(\mathcal{Z}, \mathcal{Y}) \in Atts$ and $U \cap I \neq \emptyset$ **do**
13:              $\Gamma = \Gamma \cup [U \cap (\mathcal{A}v(\mathcal{Z}, \mathcal{Y}, \mathcal{X})])$
14:          **end for**
15:      **end for**
16:      $\Lambda = \{I\}$
17:      **for all** $X \in \Gamma$ **do**
18:          $\Lambda = \Lambda \cap \{X\}$
19:      **end for**
20:      $D = D \cup \Lambda$
21: **end for**

Assuming that unions and intersections are computed in constant time, this algorithm has a cubic order of execution in the worst case. Particularly the order of execution is mnp being m, n and p the cardinality of the sets S, $Args$ and P respectively. The choice of relevant arguments and t-profiles in the inner loops helps to keep a low execution time.

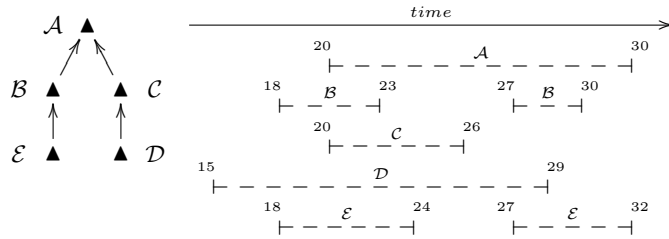**Example 3** *Consider the framework depicted on Figure 4.*

Fig. 4. Framework to ilustrate the use of Algorithm 1

Let us analyze the behavior of Algorithm DEFENDED PERIODS *when it is called with the argument* $\mathcal{A}$ *and the empty set as* P. *The first line of the algorithm builds a set* S *formed by the intervals in which* $\mathcal{A}$ *is attacked. In this*

case the set is $S = \mathcal{A}v(\mathcal{B},\mathcal{A}) \cup \mathcal{A}v(\mathcal{C},\mathcal{A})$, i.e. $\{[20,23],[27,30],[20,26]\}$. The set S is then partitionated in order to properly analyze defenses. The S is then $\{[20,23],[24,26],[27,30]\}$. For each one of these intervals, a set of defenses is calculated. This partition is made since defense requirements are different in each one of them. In $[20,23]$ a defense is needed against $\mathcal{B}$ and $\mathcal{C}$, and in $[24,26]$ against $\mathcal{C}$, while in $[27,30]$ it is needed against $\mathcal{B}$. The set D is initialized with the periods on $\mathcal{A}v(\mathcal{A}) \stackrel{I}{-} S$, i.e. those periods of time in which $\mathcal{A}$ is not attacked. Following the example, the set D is $\{[31,32]\}$, $\{[20,30]\} \stackrel{I}{-} \{[20,23],[24,26],[27,30]\}$. The inner loops are not relevant in this case since the are no defenders for $\mathcal{A}$, since P is the emptyset. So D ends with $\{[31,32]\}$.

If the algorithm is called again with $\mathcal{A}$ and $P = \{ [\mathcal{D},\mathcal{A}v(\mathcal{D})], [\mathcal{E},\mathcal{A}v(\mathcal{E})]\}$, the inner loops become relevant now. Set D starts with $\{[31,32]\}$, i.e. those intervals in which $\mathcal{A}$ is not attacked. Lets see in detail how the algorithm evaluates Γ for $[20,23]$. The for-cycle on line 11 chooses the arguments that attack $\mathcal{A}$ at $[20,23]$. In this case there are two arguments: $\mathcal{B}$ and $\mathcal{C}$. The inner for-cycle (line 12) chooses the t-profiles in P that can provide a defense, i.e. where the time in the profile overlaps $[20,23]$ and they attack the attacker. The set Γ begins as an empty set. Then,

Argument	t-profile	Γ
$\mathcal{B}$	$[\mathcal{E},\mathcal{A}v(\mathcal{E})]$	$\mathcal{A}v(\mathcal{E}) \cap \mathcal{A}v(\mathcal{E},\mathcal{B},\mathcal{A})$
		$\{[18,24],[27,32]\} \cap \{[20,23],[27,30]\}$
		$\{[20,23],[27,30]\}$
$\mathcal{C}$	$[\mathcal{D},\mathcal{A}v(\mathcal{D})]$	$\mathcal{A}v(\mathcal{D}) \cap \mathcal{A}v(\mathcal{D},\mathcal{C},\mathcal{A})$
		$\{[15,29]\} \cap \{[20,26]\}$
		$\{[20,26]\}$

Finally it makes the intersection of these sets with I, given that the defense provided by P in I is I.

6   Conclusions and future work

In this work we presented an extension of previously defined Timed Argumentation Frameworks in which arguments with more than one availability interval are considered. These arguments are called *intermittent arguments*, and are temporaly available with some repeated interruptions in time. Using this extended timed argumentation framework, we studied the notion of acceptability, which requires the consideration of time as a new dimension, leading to the definition of *t-profiles* of timed arguments. This is important in argumentation change, when arguments are available and cease to be so as the framework evolves through time.

Future work has several directions. A declarative formalization of acceptability semantics using t-profiles is being studied. Also, algorithms for other classical semantics such as stable extensions are being considered. We are also interested in the general study of the evolution of the framework through time. For instance, using timed frameworks it is possible to analyze the impact of a new

argument, by examining how much the status of acceptance is affected for other arguments, and for how long they are affected. This may lead to future semantic elaborations that contribute to the research area of argumentation change.

References

1. Allen, J.: Maintaining knowledge about temporal intervals. Communications of the ACM 1(26), 832–843 (1983)
2. Amgoud, L., Cayrol, C.: On the acceptability of arguments in preference-based argumentation. In: 14th Conference on Uncertainty in Artificial Intelligence (UAI'98). pp. 1–7. Morgan Kaufmann (1998)
3. Augusto, J.C., Simari, G.R.: Temporal defeasible reasoning. Knowl. Inf. Syst. 3(3), 287–318 (2001)
4. Baroni, P., Giacomin, M.: Resolution-based argumentation semantics. In: Proc. of 2nd International Conf. on Computational Models of Argument (COMMA 2008). pp. 25–36 (2008)
5. Bench-Capon, T.: Value-based argumentation frameworks. In: Proc. of Nonmonotonic Reasoning. pp. 444–453 (2002)
6. Cayrol, C., de Saint-Cyr, F.D., Lagasquie-Schiex, M.C.: Change in abstract argumentation frameworks: adding an argument. In: Journal of Artificial Intelligence Research. vol. 38, pp. 49–84 (2010)
7. Cobo, M., Martinez, D., Simari, G.: An approach to timed abstract argumentation. In: Proc. of Int. Workshop of Non-monotonic Reasoning 2010 (2010)
8. Cobo, M., Martinez, D., Simari, G.: On admissibility in timed abstract argumentation frameworks. In: Coelho, H., Studer, R., Wooldridge, M. (eds.) ECAI. Frontiers in Artificial Intelligence and Applications, vol. 215, pp. 1007–1008 (2010)
9. Dechter, R., Meiri, I., Pearl, J.: Temporal constaints networks. In: Proceedings KR '89. pp. 83–93 (1989)
10. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence 77(2), 321–358
11. Jakobovits, H.: Robust semantics for argumentation frameworks. Journal of Logic and Computation 9(2), 215–261 (1999)
12. Mann, N., Hunter, A.: Argumentation using temporal knowledge. In: Proc. of 2nd International Conf. on Computational Models of Argument (COMMA 2008). pp. 204–215 (2008)
13. Martínez, D.C., García, A.J., Simari, G.R.: Modelling well-structured argumentation lines. In: Proc. of XX IJCAI-2007. pp. 465–470 (2007)
14. Meiri, I.: Combining qualitative and quantitative contraints in temporal reasoning. In: Proceedings of AAAI '92. pp. 260–267 (1992)
15. Rotstein, N.D., Moguillansky, M.O., Falappa, M.A., Garcia, A.J., Simari, G.R.: Argument theory change: Revision upon warrant. In: Proceeding of the 2008 Conference on Computational Models of Argument- COMMA 2008. pp. 336–347 (2008)
16. Rotstein, N.D., Moguillansky, M.O., Garcia, A.J., Simari, G.R.: An abstract argumentation framework for handling dynamics. In: Proc. of Int. Workshop of Non-monotonic Reasoning 2010. pp. 131–139 (2010)
17. Vreeswijk, G.A.W.: Abstract argumentation systems. Artificial Intelligence 90(1–2), 225–279 (1997)