An Approach to Timed Abstract Argumentation

Ma. Laura Cobo and **Diego C. Martínez** 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

Abstract

Argumentation is a reasoning formalism where arguments and counterarguments are evaluated in order to select the ones that can be accepted together. In classical abstract argumentation, arguments and attacks are static in the sense that the same set of arguments is always available to be considered. In this work, we propose a novel abstract argumentation framework where arguments are only valid for consideration in a given period of time, which is defined for every individual argument. Thus, the existence of attacks and defenses is related to the moment in time where the assessment is done. We define several semantic notions to capture the acceptance of arguments on this framework.

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 (Dung 1993; Vreeswijk 1997; Amgoud and Cayrol 2002) 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 single abstract 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, then the acceptance of $\mathcal{B}$ is conditioned by the acceptance of $\mathcal{A}$, but not the other way around. It is said that argument $\mathcal{A}$ *attacks* $\mathcal{B}$, and it implies a priority between conflicting arguments.

The simplest abstract framework is defined by Dung in (Dung 1993). It only includes a set of abstract arguments and a binary relation of attack between arguments. Several semantic notions are defined and the Dung's argument extensions became the foundation of further research. Other proposals extend Dung's framework by the addition of new elements, such as preferences between arguments (Amgoud and Cayrol 2002; Bench-Capon 2002) or subarguments (Martínez, García, and Simari 2007). Other authors use the original framework to elaborate new extensions (Caminada 2006; Jakobovits 1999; Baroni and Giacomin 2008). All of these proposals are based on varied abstract formalizations of arguments and attacks.

In this paper we investigate the modeling of arguments that are relevant only during a certain period of time, called here *timed arguments*, and their semantical consequences. A timed argument refers to information that is dependant on time. This kind of argument has a limited influence in the system. For example, consider the following arguments

A1: *We can go out tonight, since the house alarm can be repaired by the handyman.*
A2: *The alarm cannot be repaired, as the handyman must keep to his bed.*

Argument A2 attacks argument A1. However, A2 is only relevant in the interval of time in which the handyman is ill. Outside this interval, argument A1 cannot be considered attacked by A2. There are no previous abstract proposals allowing the modeling of this kind of arguments.

As related work in combining time and argumentation, in (Mann and Hunter 2008) 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 (Augusto and Simari 2001). In contrast, here we maintain our development at the abstract level in an effort to capture intuitions related with the dynamic interplay of arguments as they become available and cease to be so.

In this work we propose an abstract argumentation framework equipped with a special kind of timed arguments, which are relevant only during a time interval. We formalize the notion of defense between timed arguments and we construct an argument extension based on acceptability (Dung 1993).

In the following section we recall classic argumentation semantic notions. Thereafter, time-intervals and the terminology used in this work are defined, towards the presentation of our Timed Abstract Argumentation Framework.

Classic abstract argumentation

Dung defines several argument extensions that are used as a reference for many authors. The formal definition of the classic argumentation framework follows.

Definition 1 (Dung 1995) *An argumentation framework is a pair $AF = \langle AR, attacks \rangle$ where AR is a set of arguments, and attacks is a binary relation on AR, i.e. $attacks \subseteq AR \times AR$.*

Arguments are denoted by labels starting with an uppercase letter, leaving the underlying logic unspecified. A set of accepted arguments is characterized in (Dung 1995) using the concept of *acceptability*, which is a central notion in argumentation, formalized by Dung in the following definition.

Definition 2 (Dung 1995) *An argument $\mathcal{A}$ is acceptable w.r.t a set of arguments S if and only if every argument $\mathcal{B}$ attacking $\mathcal{A}$ is attacked by an argument in S.*

If an argument $\mathcal{A}$ is acceptable with respect to a set of arguments S then it is also said that S *defends* $\mathcal{A}$. Also, the attackers of the attackers of $\mathcal{A}$ are called *defenders* of $\mathcal{A}$. We will use these terms throughout this paper.

Acceptability is the main property of Dung's semantic notions, which are summarized in the following definition.

Definition 3 *A set of arguments S is said to be*
– conflict-free if there are no arguments $\mathcal{A}, \mathcal{B}$ in S such that $\mathcal{A}$ attacks $\mathcal{B}$.
– admissible if it is conflict-free and defends all its elements.
– a complete extension if S is admissible and it includes every acceptable argument w.r.t. S.
– a grounded extension if and only if it is the least (for set inclusion) complete extension.

The grounded extension is also the least fixpoint of a simple monotonic *characteristic* function:

$F_{AF}(S) = \{\mathcal{A} : \mathcal{A} \text{ is acceptable with respect to } S\}$.

In (Dung 1995), theorems stating conditions of existence and equivalence between these extensions are also introduced.

Example 1 *Consider the argumentation framework $AF_1 = \langle AR, attacks \rangle$, where $AR = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{F}, \mathcal{G}, \mathcal{H}\}$ and $attacks = \{(\mathcal{B}, \mathcal{A}), (\mathcal{C}, \mathcal{B}), (\mathcal{D}, \mathcal{A}), (\mathcal{E}, \mathcal{D}), (\mathcal{G}, \mathcal{H}), (\mathcal{H}, \mathcal{G})\}$. Then:*
– $\{\mathcal{A}, \mathcal{C}, \mathcal{E}\}$ is an admissible set of arguments.
– $\{\mathcal{A}, \mathcal{C}, \mathcal{E}, \mathcal{F}, \mathcal{G}\}$ is a preferred extension.
– $\{\mathcal{A}, \mathcal{C}, \mathcal{E}, \mathcal{F}\}$ is the grounded extension.

This abstract formalism is a sufficient condition to define some basic extensions on arguments. In this work we study the formalization of timed-arguments in an abstract framework, and we present an argument extension inspired by grounded semantics. Some additional remarks about time-related notions are needed.

Timed Arguments

In order to capture a time-based model of argumentation, we enrich the classical abstract frameworks with temporal information regarding arguments. The problem of representing temporal knowledge and temporal reasoning arises in a lot of disciplines, including Artificial Intelligence. There are many ways of representing temporal knowledge. A usual way to do this is to determine a *primitive* to represent time. This primitive can be *time points*, *temporal intervals* or both of them. Once the primitive for time representation is settled, relationships between primitives must be defined, as *metric relations*. For instance,

- point-point metric relations, (Meiri 1992; Dechter, Meiri, and Pearl 1989).
- point-interval relations (Meiri 1992).
- interval-interval relations, called *interval algebra* (Allen 1983).

In this work we will only use *temporal intervals of discrete time* as primitives for time representation, and thus only metric relations for intervals are applied. Intervals are represented as a pair of elements between brackets. In this work we will use *temporal intervals of discrete time* as primitives for time representation, and thus only metric relations for intervals are applied.

There are thirteen possible relations between intervals. In Table 1 we present seven of them (the remaining six are defined as the inverse). The 'x's and 'y's represents the interval X and Y respectively. There are six more possible relations that can be seen as the inverse of the ones presented (except, of course, *Equal*). The table shows the relation between *endpoints*. 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. It is important to remark that $-\infty < i$ and $i < \infty$ for any value i. Also that $\infty = \infty$ and $-\infty = -\infty$.

Argumentation Framework

The consideration of time restrictions for arguments is formalized through an *availability function*, which defines a temporal interval for each argument in the framework. This interval states the period of time in which an argument is available for consideration in the argumentation scenario.

Definition 4 *[Availability function] The availability function $\mathcal{A}v$ is defined as $\mathcal{A}v : Args \rightarrow [a, b]$ with $a, b \in \mathbb{Z} \cup \{-\infty, \infty\}$, such as*

- *$\mathcal{A}v(\mathcal{A}) = [i, i]$ denotes that $\mathcal{A}$ is only available at moment i.*
- *$\mathcal{A}v(\mathcal{A}) = [i, \infty)$ denotes that $\mathcal{A}$ is available since moment i (including i).*

Relation	*Symb*	*e.g.*	*Relation on Endpoints*
X Before Y	ⓑ	xx yy	$X^+ < Y^-$
X Meets Y	ⓜ	xxyy	$X^+ = Y^-$
X OverlapsY	ⓞ	xxx yyy	$X^- < Y^-$, $X^+ > Y^-$
X Starts Y	ⓢ	xxx yyyyyy	$X^- = Y^-$, $X^+ < Y^+$
X During Y	ⓓ	xxx yyyyyyy	$X^- > Y^-$, $X^+ < Y^+$
X Finishes Y	ⓕ	xxx yyyyyy	$X^+ = Y^+$, $X^- > Y^-$
X Equal Y	ⓔ	xxx yyy	$X^- = Y^-$, $X^+ = Y^+$

Table 1: Seven possible qualitative relations among arguments (Allen 1983).

- $\mathcal{A}v(\mathcal{A}) = (-\infty, i]$ *denotes that* $\mathcal{A}$ *is available until moment* i *(including* i*).*
- $\mathcal{A}v(\mathcal{A}) = [i, j]$ *denotes that* $\mathcal{A}$ *is available since moment* i *until moment* j *(including both* i *and* j*).*
- $\mathcal{A}v(\mathcal{A}) = (-\infty, \infty)$ *denotes that* $\mathcal{A}$ *is always available.*

$\mathcal{A}v(\mathcal{A})$ *is called the time restriction, or availability, of argument* $\mathcal{A}$.

The formal definition of our timed abstract argumentation framework follows.

Definition 5 *[Timed framework] A timed abstract argumentation framework (TAF) is a 3-uple* $\langle Args, Atts, \mathcal{A}v\rangle$ *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.*

Example 2 *The triplet* $\langle Args, Atts, \mathcal{A}v\rangle$*, where* $Args = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{E}, \mathcal{F}, \mathcal{G}, \mathcal{H}\}$*,* $Atts = \{(\mathcal{B}, \mathcal{A}), (\mathcal{C}, \mathcal{A}), (\mathcal{C}, \mathcal{B}), (\mathcal{D}, \mathcal{C}), (\mathcal{E}, \mathcal{B}), (\mathcal{F}, \mathcal{E}), (\mathcal{G}, \mathcal{F}), (\mathcal{H}, \mathcal{F})\}$ *and the availability function is defined as*

Args	$\mathcal{A}v$	*Args*	$\mathcal{A}v$
$\mathcal{A}$	$[5, 20]$	$\mathcal{B}$	$[10, 20]$
$\mathcal{C}$	$(-\infty, 7]$	$\mathcal{D}$	$(-\infty, \infty)$
$\mathcal{E}$	$[5, 50]$	$\mathcal{F}$	$(-\infty, 15]$
$\mathcal{G}$	$[0, 12]$	$\mathcal{H}$	$[13, \infty)$

is a timed abstract argumentation framework.

The framework of Example 2 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. For space reasons, only some relevant time points are shown.

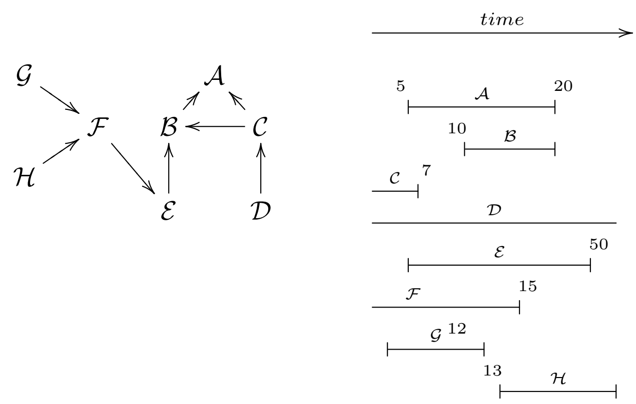

Figure 1: Framework of Example 2

As stated before, the availability of arguments is tied to a temporal restriction. Thus, an attack to an argument may actually occur only if both the attacker and the attacked argument are available. In other words, an attack between two arguments may be *attainable*, under certain conditions. *Attainable attacks* are attacks that will eventually occur in some period of time. In order to formalize this, we need to compare time intervals, using the previously defined metric relations.

Definition 6 *[Attainable attack] Let* $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ *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 one of the following conditions holds:*

- $\mathcal{A}v(\mathcal{A})\ R\ \mathcal{A}v(\mathcal{B})$*, where* $R \in \{$ⓢ, ⓓ, ⓕ, ⓜ, ⓞ, ⓔ$\}$
- $\mathcal{A}v(\mathcal{B})\ R\ \mathcal{A}v(\mathcal{A})$*, where* $R \in \{$ⓢ, ⓓ, ⓕ, ⓜ, ⓞ$\}$

Note that an attack is attainable if the availability of both the attacker and the attacked argument eventually overlaps.

Example 3 *Consider the timed argumentation framework of Example 2. The attacks* $(\mathcal{E}, \mathcal{B})$ *and* $(\mathcal{H}, \mathcal{F})$ *are both attainable in the framework. Attack* $(\mathcal{E}, \mathcal{B})$ *is attainable in the interval* $\mathcal{A}v(\mathcal{B})$ *since* $\mathcal{A}v(\mathcal{B})$ⓓ $\mathcal{A}v(\mathcal{E})$*. Attack* $(\mathcal{H}, \mathcal{F})$ *is attainable at* $[13, 15]$*, since* $\mathcal{A}v(\mathcal{F})$ⓞ $\mathcal{A}v(\mathcal{H})$*, only in that interval of time. For the same reason, attack* $(\mathcal{B}, \mathcal{A})$ *is attainable at* $[10, 20]$*,* $(\mathcal{C}, \mathcal{A})$ *at* $[5, 7]$*,* $(\mathcal{D}, \mathcal{C})$ *at* $\mathcal{A}v(\mathcal{C})$*,* $(\mathcal{F}, \mathcal{E})$ *at* $[5, 15]$*,* $(\mathcal{G}, \mathcal{F})$ *at* $\mathcal{A}v(\mathcal{G})$*.*

The attack $(\mathcal{C}, \mathcal{B}) \in Atts$ *is not attainable, since* $\mathcal{A}v(\mathcal{C})$ⓑ $\mathcal{A}v(\mathcal{B})$*. The arguments involved in this attack relation are never available at the same time.*

The set of all attainable attacks in the framework Φ is denoted $AttainableAtts_{\Phi}$. It is also possible to define the attainability of attacks at a particular point of time, as shown next.

Definition 7 *[Attainable attack at i] Let* $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ *be a TAF, and let* $\mathcal{A}, \mathcal{B} \in Args$ *such* *that $(\mathcal{A},\mathcal{B}) \in Atts$. The attack $(\mathcal{A},\mathcal{B})$ is said to be attainable at moment i if $i \in \mathcal{A}v(\mathcal{A})$ and $i \in \mathcal{A}v(\mathcal{B})$.*

The set of attainable attacks of Φ at moment i is denoted $AttainableAtts_{\Phi}(i)$.

Example 4 *Consider the framework of Example 2. The set $AttainableAtts_{\Phi}$ is: $\{(\mathcal{B},\mathcal{A}),(\mathcal{C},\mathcal{A}),(\mathcal{D},\mathcal{C}),(\mathcal{E},\mathcal{B}),(\mathcal{F},\mathcal{E}),(\mathcal{G},\mathcal{F}),(\mathcal{H},\mathcal{F})\}$. The attainable attacks at time point $i = 13$ is $AttainableAtts_{\Phi}(13) = \{(\mathcal{B},\mathcal{A}),(\mathcal{E},\mathcal{B}),(\mathcal{F},\mathcal{E}),(\mathcal{H},\mathcal{F})\}$*

Acceptability on Timed Arguments

As attacks may occur only on a period of time (that in which the participants are available), argument defense is also occasional. The classical definition of acceptability requires an adaptation to a timed context. It is easy to analyze a defense over a single moment or time point, since attacks are either available or not, and thus an argument may be defended or not. However, when we expand the analysis over a wider temporal domain the situation is more complex, since defenses may occur sporadically. An argument should be considered defended if it has defenders during its availability period. For example, an argument $\mathcal{A}$ may be defended by $\mathcal{X}$ in the first half of its time interval, and later by an argument $\mathcal{Y}$ in the second half. Although $\mathcal{X}$ is not capable of providing a full defense, argument $\mathcal{A}$ is defended while $\mathcal{A}$ is available.

In order to define argument defense in timed abstract argumentation, we need to define *when* an argument is *threatened* by other arguments. Later on, all the possible defenses in that interval must be gathered.

Definition 8 *[Threatened Arg] Let $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, and let $\mathcal{A} \in Args$. Argument $\mathcal{A}$ is a threatened argument if there is at least one argument $\mathcal{B}$, such that $(\mathcal{B},\mathcal{A}) \in AttainableAtts_{\Phi}$.*

Definition 9 *[Threat Interval] Let $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, and let $\mathcal{A},\mathcal{B} \in Args$ such that $(\mathcal{B},\mathcal{A}) \in AttainableAtts_{\Phi}$. The threat interval of $\mathcal{B}$ to $\mathcal{A}$, noted as $\tau_{\mathcal{B}}^{\mathcal{A}}$, is defined as:*

- *$\mathcal{A}v(\mathcal{A})$: If $\mathcal{A}v(\mathcal{A})\ R\ \mathcal{A}v(\mathcal{B})$ and $R \in \{e, d, s, f\}$*
- *$\mathcal{A}v(\mathcal{B})$: If $\mathcal{A}v(\mathcal{B})\ R\ \mathcal{A}v(\mathcal{A})$ and $R \in \{d, s, f\}$*
- *$[\mathcal{A}v(\mathcal{B})^-, \mathcal{A}v(\mathcal{A})^+]$: If $\mathcal{A}v(\mathcal{A}) \circledcirc \mathcal{A}v(\mathcal{B})$*
- *$[\mathcal{A}v(\mathcal{A})^-, \mathcal{A}v(\mathcal{B})^+]$: If $\mathcal{A}v(\mathcal{B}) \circledcirc \mathcal{A}v(\mathcal{A})$*

The threat interval is the period of time in which an argument attacks another. Consider again the framework of Example 2. Argument $\mathcal{A}$ has two threat intervals since there are two attainable attacks, $(\mathcal{B},\mathcal{A})$ and $(\mathcal{C},\mathcal{A})$. The $\tau_{\mathcal{B}}^{\mathcal{A}}$ is $[10, 20]$ while the $\tau_{\mathcal{C}}^{\mathcal{A}}$ is $[5, 7]$.

The notion of defense is the basis of argument acceptability. In our framework, a defense occurs when an argument $\mathcal{X}$ is threatened by another argument $\mathcal{Y}$, which in turn is threatened by a third argument $\mathcal{Z}$. The first one is defended by the last one. However, this requirement is not a sufficient condition. Time restrictions must be taken into account: the defender $\mathcal{Z}$ must attack $\mathcal{Y}$ *while* this argument attacks $\mathcal{X}$.

Definition 10 *Let $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, and let $\mathcal{A},\mathcal{B},\mathcal{C} \in Args$ such that $(\mathcal{B},\mathcal{A}) \in AttainableAtts_{\Phi}$ and $(\mathcal{C},\mathcal{B}) \in AttainableAtts_{\Phi}$. The interval in which $\mathcal{C}$ defends $\mathcal{A}$ against $\mathcal{B}$, denoted $\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}$, is defined as: $\mathcal{A}v(\mathcal{C}) \cap \tau_{\mathcal{B}}^{\mathcal{A}}$*

The intersection of two intervals is the interval formed by time-points in common.

Argument $\mathcal{C}$ is actually a defender of $\mathcal{A}$ if the provided defense interval is non empty. If the defense does not cover the $\tau_{\mathcal{B}}^{\mathcal{A}}$ then the defense task is not fully achieved, since there are moments of time where the threat of $\mathcal{B}$ still prevails.

Definition 11 *[Defender] Let $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, and let $\mathcal{A},\mathcal{B},\mathcal{C} \in Args$ such that $(\mathcal{B},\mathcal{A}) \in AttainableAtts_{\Phi}$ and $(\mathcal{C},\mathcal{B}) \in AttainableAtts_{\Phi}$. We will say that:*

- *$\mathcal{C}$ is a full-defender of $\mathcal{A}$ (in $\tau_{\mathcal{B}}^{\mathcal{A}}$): if $\tau_{\mathcal{B}}^{\mathcal{A}}\ e\ \delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}$*
- *$\mathcal{C}$ is a partial-defender of $\mathcal{A}$ (in $\tau_{\mathcal{B}}^{\mathcal{A}}$): if $\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}} \neq [\ ]$*

Consider the framework of Example 2. Argument $\mathcal{D}$ is a full-defender of $\mathcal{A}$ (in $\tau_{\mathcal{C}}^{\mathcal{A}}$) since $\tau_{\mathcal{C}}^{\mathcal{A}}$ is $[5, 7]$ and $\delta_{\mathcal{D}\mathcal{C}}^{\mathcal{A}}$ is $[5, 7]$ (because $\tau_{\mathcal{C}}^{\mathcal{A}}\ d\ \mathcal{A}v(\mathcal{D})$, *i.e*, the intersection in non empty). Argument $\mathcal{G}$ is a partial-defender of $\mathcal{E}$ (in $\tau_{\mathcal{F}}^{\mathcal{E}}$) since the $\tau_{\mathcal{F}}^{\mathcal{E}}$ is $[5, 15]$ and $\delta_{\mathcal{G}\mathcal{F}}^{\mathcal{E}}$ is $[5, 12]$ (because $\mathcal{A}v(\mathcal{G}) \circledcirc \tau_{\mathcal{F}}^{\mathcal{E}}$).

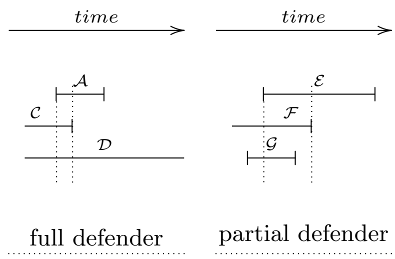

Figure 2: Full and partial defenders in Example 2

If a partial defender is found then there are uncovered sub-intervals that must still be defended.

Definition 12 *[Remaining threat interval] Let $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, and let $\mathcal{A},\mathcal{B},\mathcal{C} \in Args$ such that $\mathcal{C}$ is a partial-defender of $\mathcal{A}$. Argument $\mathcal{A}$ is (still) threatened by $\mathcal{B}$ in the following interval:*

- *$[(\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}{}^{+} + 1), \tau_{\mathcal{B}}^{\mathcal{A}+}]$: If $\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}\ s\ \tau_{\mathcal{B}}^{\mathcal{A}}$*
- *$[\tau_{\mathcal{B}}^{\mathcal{A}-}, (\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}{}^{-} - 1)]$: If $\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}\ f\ \tau_{\mathcal{B}}^{\mathcal{A}}$*
- *$[\tau_{\mathcal{B}}^{\mathcal{A}-}, (\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}{}^{-} - 1)]$ and $[(\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}{}^{+} + 1), \tau_{\mathcal{B}}^{\mathcal{A}+}]$: If $\delta_{\mathcal{C}\mathcal{B}}^{\mathcal{A}}\ d\ \tau_{\mathcal{B}}^{\mathcal{A}}$*

In the example above the remaining threat interval is $[13, 15]$, recall that $\tau_{\mathcal{F}}^{\mathcal{E}}$ is $[5, 15]$ and $\delta_{\mathcal{G}\mathcal{F}}^{\mathcal{E}}$ is [5,12] (see Figure 2). We will say that an argument $\mathcal{A}$ is *completely defended* if it is defended in every moment of $\mathcal{A}v(\mathcal{A})$. Clearly, this may require more than one defender.

The union of several intervals do not necessarily leads to a single interval. Algorithm 1 builds the small set of intervals that may result from a set of intervals. This operation simplifies the task of deternining if some argument is defended from an attack.

Algorithm 1 Union of intervals

Require: a set of intervals σ
Ensure: γ, the minimal set of intervals obtained as union of σ's interval.

$\gamma \leftarrow \emptyset$
Choose X from σ such as X^- is the lower endpoint in σ
Remove X from σ
while $\sigma \neq \emptyset$ **do**
    Choose Y from σ such as Y^- is the lower endpoint in σ
    Remove Y from σ
    if $X \,\mathrm{b}\, Y$ **then**
        $\gamma \leftarrow \gamma \cup X$
        $X \leftarrow Y$
    else if $X \,\mathrm{m}\, Y$ **then**
        $X \leftarrow [X^-, Y^+]$
    else if $X \,\mathrm{o}\, Y$ **then**
        $X \leftarrow [X^-, Y^+]$
    else if $X \,\mathrm{s}\, Y$ **then**
        $X \leftarrow Y$
    end if
end while
$\gamma \leftarrow \gamma \cup X$
return γ

Algorithm 2 Defended from an attack

Require: $\Phi = \langle Args, Atts, \mathcal{A}v \rangle$ be a TAF, and $\mathcal{A}, \mathcal{B} \in Args$ such that $(\mathcal{B}, \mathcal{A}) \in AttainableAtts_\Phi$
Require: S, $S \subseteq Args$ (possible defenders of interest)
Ensure: Whether $\mathcal{A}$ is defended from $\mathcal{B}$ in S or not.

1: $\delta \leftarrow \tau_{\mathcal{B}}^{\mathcal{A}}$
2: $set\_of\_intervals = \emptyset$
3: **for all** $\mathcal{C} \in S$ such that $(\mathcal{C}, \mathcal{B}) \in AttainableAtts_\Phi$ and the $\mathcal{C}$-defense interval $\neq \emptyset$ **do**
4:     $set\_of\_intervals \leftarrow set\_of\_intervals \cup \mathcal{A}v(\mathcal{C})$
5: **end for**
6: $\gamma \leftarrow$ alg $1(set\_of\_intervals)$
7: $defended \leftarrow false$
8: **while** $defended = false \wedge \gamma \neq \emptyset$ **do**
9:     Choose X from γ such as X^- is the lower endpoint of all intervals in γ
10:     Remove X from γ
11:     **if** $\delta \,\mathrm{d}\, X \ \vee\ \delta \,\mathrm{s}\, X \ \vee\ \delta \,\mathrm{f}\, X$ **then**
12:         $defended \leftarrow true$
13:     **end if**
14: **end while**
15: **return** $defended$

Computing defenses

Given a timed argument $\mathcal{X}$ and its attacker $\mathcal{Y}$, the search for enough defenders for $\mathcal{X}$ is not an easy task. Algorithm 2 checks if a set of arguments include the necessary defenders of an attacked argument.

Let us apply Algorithm 2 to the framework of Example 2, with the set $Args$ and arguments $\mathcal{E}$ and $\mathcal{F}$, as $(\mathcal{F}, \mathcal{E}) \in AttainableAtts_\Phi$ (see Example 4). The threat interval $\tau_{\mathcal{F}}^{\mathcal{E}}$ is $[5, 15]$. In lines 3-5, a set containing the availability interval of every attacker of $\mathcal{F}$ (and thus, a possible defender of $\mathcal{E}$) is built. In this case the set finally contains two intervals $\mathcal{A}v(\mathcal{G})$ and $\mathcal{A}v(\mathcal{H})$. In line 6, the algorithm calculates the union of these intervals. This leads to only one interval: $[0, \infty)$, since $\mathcal{A}v(\mathcal{G})$ is $[0, 12]$ and $\mathcal{A}v(\mathcal{H})$ is $[13, \infty)$. In order to defend $\mathcal{E}$, the intervals obtained from the union must cover the interval $[5, 15]$ completely. Clearly in this framework, $\mathcal{E}$ is defended in $Args$ attack since $[5, 15] \,\mathrm{d}\, [0, \infty)$.

Algorithm 2 takes into account only one attack and it finds whether an argument is defended or not in a given set. It does not consider further defenses (*i.e.*, defenders of defenders). Algorithm 3 goes deeper. It takes a particular argument $\mathcal{A}$ and an interval I an determines if it is possible to defend $\mathcal{A}$ *to the grounds*, *i.e.*, to defend all of its direct or indirect defenders, as it is done in the classical grounded extension (Dung 1993) (for simplicity, we are not including the required controls about cycles in argumentation). Thus, it is possible to find out if an argument is defended in its entire availability interval (its "*lifetime*").

Algorithm 3 Defended to the grounds

Require: $\Phi = \langle Args, Atts, \mathcal{A}v \rangle$ be a TAF, $\mathcal{A}$ and interval I where $\mathcal{A}$ is available
Ensure: $\mathcal{A}$ is granted from Φ at interval I

$\mathcal{A}v(\mathcal{A}) = I$
response $\leftarrow$ true
for all $\mathcal{B} \in Args$ such that $(\mathcal{B}, \mathcal{A}) \in AttainableAtts_\Phi$ **do**
    Create a set ρ containing every defender of $\mathcal{A}$ against $\mathcal{B}$
    for all $\mathcal{C} \in \rho$ **do**
        $I \leftarrow \mathcal{A}v(\mathcal{C}) \cap \tau_{\mathcal{B}}^{\mathcal{A}}$
        if $\neg$ alg $3(\Phi, \mathcal{C}, I)$ **then**
            remove $\mathcal{C}$ from ρ
        end if
    end for
    response = response $\wedge$ alg 2 $(\Phi, \mathcal{A}, \mathcal{B}, \rho)$
end for
return response

Argument extensions

Acceptability is an important notion in classical argumentation semantics, and it is the basis of admissible extensions. For the extended framework presented in this paper, admissibility must be slightly adapted.

Definition 13 *[Conflict free] Let $\langle Args, Atts, \mathcal{A}v \rangle$ be a TAF, and let $S \subseteq Args$. The set S is said to be conflict-free at moment i if there are no two arguments $\mathcal{A}$ and $\mathcal{B}$ such that $(\mathcal{A}, \mathcal{B}) \in AttainableAtts_\Phi(i)$.*

The previous definition refers to a particular moment i. As it is reasonable, conflict-freeness can be generalized using availability intervals of the involved arguments.

Definition 14 *[Conflict free (revisited)] Let $\langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, and let $S \subseteq Args$. The set S is said to be conflict-free within the framework if there are no two argument $\mathcal{A}$ and $\mathcal{B}$ such that $\exists i \in Av(\mathcal{A}) : (\mathcal{A}, \mathcal{B}) \in AttainableAtts_{\Phi}(i)$.*

Conflict freeness is not enough. Argument defense is required whenever an attack is present. Acceptability at a moment i requires a specific defense in a concrete period of time.

Definition 15 *[Acceptable at moment i] Let $\langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, $\mathcal{A} \in Args$ and $S \subseteq Args$. Argument $\mathcal{A}$ is said to be acceptable with respect to S at moment i if for every attacker $\mathcal{B} \in Args$ such that $(\mathcal{B}, \mathcal{A}) \in AttainableAtts_{AF}(i)$, there is a defender $\mathcal{C} \in Args$, with $(\mathcal{C}, \mathcal{B}) \in AttainableAtts_{AF}(i)$.*

Clearly, if an argument is acceptable with respect to a set S at moment i, there exists at least a defender $\mathcal{X}$, such that interval where $\mathcal{X}$ actually provides defense includes the moment i. Note that an argument may not be defended in its entire availability interval, and yet acceptable with respect to a set in a particular moment.

Definition 16 *[Admissible set at moment i] Let $\langle Args, Atts, \mathcal{A}v\rangle$ be a TAF, and let $S \subseteq Args$. The set S is said to be admissible at moment i if*

- *it is conflict-free at moment i.*
- *every argument in S is acceptable with respect to S at moment i.*

The classic grounded extension (Dung 1993) is built from arguments with no attackers, by progressively adding arguments that can be defended from the partial set. In a similar way, it is possible to obtain a *grounded extension at moment* i, by applying the following characteristic function:

$$F_i(S) = \{\mathcal{A} : \mathcal{A} \text{ is acceptable wrt } S \text{ at moment } i\}$$

We are interested, however, in the consideration of intervals. Given the time restrictions of every argument in a TAF, the notion of "free of attackers" or "defended" is relative to a particular period of time. Thus, in a given interval of time $[a, b]$, we identify two sets of arguments: those included in every grounded extension obtained in every moment $i \in [a, b]$ and those included in some of these grounded extensions. These sets can be obtained by following Algorithm 4.

Algorithm 4 returns two sets. The set Λ contains the arguments that are present in the grounded extension of every moment $i \in I$, and the set Ψ contains the arguments that are present in some of the extensions, since they are not available or defended in every moment of I.

Consider the framework depicted in Example 2 and $I = [6, 14]$. The first step (line 1) locally redefines $\mathcal{A}v$ like this:

Algorithm 4 Full and partial grounded

Require: $\Phi = \langle Args, Atts, \mathcal{A}v\rangle$ be a TAF and interval I

Ensure: Λ the set of argument present in every grounded extension of I; Ψ the set of argument present in some grounded extension of I.

1: Redefine $\mathcal{A}v(\mathcal{X})$ as the intersection between $\mathcal{A}v(\mathcal{X})$ and I, $\forall \mathcal{X} \in Args$
2: **for all** $\mathcal{A} \in Args$ such that there is no $(\mathcal{B}, \mathcal{A}) \in AttainableAtts_{\Phi}$ **do**
3:     **if** $I \textcircled{e} \mathcal{A}v(\mathcal{A})$ **then**
4:         $\Lambda = \Lambda \cup \mathcal{A}$
5:     **else**
6:         $\Psi = \Psi \cup \mathcal{A}$
7:     **end if**
8: **end for**
9: **while** There is $\mathcal{C} \in Args$ such that $\mathcal{C} \notin \Lambda \cup \Psi$ and there is not $(\mathcal{C}, \mathcal{B}), (\mathcal{B}, \mathcal{A}) \in AttainableAtts_{\Phi}$ with $\mathcal{A} \notin \Lambda \cup \Psi$ **do**
10:     $defended = true$
11:     **for all** $\mathcal{B} \in Args$ such that $(\mathcal{C}, \mathcal{B}) \in AttainableAtts_{\Phi}$ **do**
12:         Determine, ϕ, the set of defenders of $\mathcal{C}$ from set $\Lambda \cup \Psi$ (all $\mathcal{A}$ such that $(\mathcal{B}, \mathcal{A}) \in AttainableAtts_{\Phi} \wedge ArgA \in \Lambda \cup \Psi$)
13:         **if** there is a full defender in ϕ **then**
14:             $defended = defended \wedge true$
15:         **else**
16:             $defended = defended \wedge$ call algorithm $2(\Phi, \mathcal{C}, \mathcal{B}, \Lambda \cup \Psi)$
17:         **end if**
18:     **end for**
19:     **if** $defended = true \wedge \mathcal{A}v(\mathcal{C}) \textcircled{e} I$ **then**
20:         $\Lambda = \Lambda \cup \mathcal{C}$
21:     **else**
22:         **if** $defended = true$ **then**
23:             $\Psi = \Psi \cup \mathcal{C}$
24:         **end if**
25:     **end if**
26: **end while**
27: **return** Λ, Ψ

$Args$	$\mathcal{A}v$	$Args$	$\mathcal{A}v$
$\mathcal{A}$	$[6, 14]$	$\mathcal{B}$	$[10, 14]$
$\mathcal{C}$	$[6, 7]$	$\mathcal{D}$	$[6, 14]$
$\mathcal{E}$	$[6, 14]$	$\mathcal{F}$	$[6, 14]$
$\mathcal{G}$	$[6, 12]$	$\mathcal{H}$	$[13, 14]$

This local redefinition simplifies the control over the framework since it focus the attention in the interval of interest, ignoring irrelevant periods of time. It only reduces the interval of a given argument, and thus every argument is still considered within its availability period of time. This redefinition does not implies a change in the definition of the availability function of the framework, as it only applies to the internal operation of the algorithm. The iteration of the **for**-block in line 2, returns: $\Lambda = \{\mathcal{D}\}$ and $\Psi = \{\mathcal{G}, \mathcal{H}\}$. $\mathcal{D}$ has no attackers and its (redefined) available function is equal to interval I. $\mathcal{G}$ and $\mathcal{H}$ are also unattacked but their availability do not cover interval I. Now the algorithm is ready to determine which other arguments can be added in these sets, *i.e* which arguments are defended by the ones that we already identified as defensibles. We choose an argument that can be defended by $\mathcal{D}$, $\mathcal{G}$ and/or $\mathcal{H}$. Lets consider $\mathcal{E}$; $\mathcal{A}$ can not be chosen because it has an attacked relation (from $\mathcal{B}$) with still no defense. Now the algorithm looks for all of the attackers of $\mathcal{E}$, which is only $\mathcal{F}$ in this case. It asks if $\mathcal{E}$ is defended from this attack looking for defenders only in $\Lambda \cap \Psi$. Argument $\mathcal{G}$ and $\mathcal{H}$ both defend $\mathcal{E}$ in all $\tau_{\mathcal{F}}^{\mathcal{E}}$ (although they are both partial defenders). Since $\mathcal{E}$ is defended from all the attackers and its availability is equal to I then $\mathcal{E}$ is added to Λ set. The algorithm tries to find out other argument that may be defended by the arguments in $\Lambda \cap \Psi$. It makes a similar analysis over $\mathcal{A}$ but in this case it must use the inner **for**-block since $\mathcal{A}$ has two attackers. However, $\mathcal{A}$ is defended by $\mathcal{E}$ and $\mathcal{D}$. Since $\mathcal{A}$ is defended from all the attackers and its availability is equal to I then $\mathcal{A}$ is added to Λ set. Finally $\Lambda = \{\mathcal{A}, \mathcal{E}, \mathcal{D}\}$ and $\Psi = \{\mathcal{G}, \mathcal{H}\}$.

Conclusions and future work

In this work we proposed a novel abstract argumentation framework where arguments are only valid for consideration in a given period of time, which is defined for every individual argument. Thus, the attainability of attacks and defenses is related to time. We introduced the notion of timed arguments and its time-related defense conditions.

The classical notions of admissibility and the grounded extension are adapted to this form of temporal reasoning. In a given interval of time $[a, b]$, we identify two sets of arguments: those taking part of the grounded extension of every moment $i \in [a, b]$, and those taking part of some of these extensions. Algorithms for the computation of these semantic concepts are presented.

Future work has several directions. Evaluation principles for extension-based semantics (as defined in (Baroni and Giacomin 2006)) are to be studied in the context of this new timed framework. We are formalizing new timed argument extensions capturing different possible outcomes of the framework. Finally, an analysis of complexity of the presented algorithms is going to be addressed.

In this paper we work with the simplest representation of time. The framework can be reformulated using other time representations, such as dense time instead of discrete time, or periods of time where endpoints are not set up explicitly.

Acknowledgements

The authors would like to thank the anonymous referees for they useful insights and suggestions.

References

Allen, J. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM* (26):832–843.

Amgoud, L., and Cayrol, C. 2002. A reasoning model based on the production of acceptable arguments. In *Annals of Mathematics and Artificial Intelligence*, volume 34, 1-3. 197–215.

Augusto, J. C., and Simari, G. R. 2001. Temporal defeasible reasoning. *Knowl. Inf. Syst.* 3(3):287–318.

Baroni, P., and Giacomin, M. 2006. Evaluation and comparison criteria for extension-based argumentation semantics. In *Proc. of I International Conf. on Computational Models of Arguments, COMMA 2006*, 157–168.

Baroni, P., and Giacomin, M. 2008. Resolution-based argumentation semantics. In *Proc. of 2nd International Conference on Computational Models of Argument (COMMA 2008)*, 25–36.

Bench-Capon, T. 2002. Value-based argumentation frameworks. In *Proc. of Nonmonotonic Reasoning*, 444–453.

Caminada, M. 2006. Semi-stable semantics. In *Proceedings of I International Conference on Computational Models of Arguments, COMMA 2006*, 121–130.

Dechter, R.; Meiri, I.; and Pearl, J. 1989. Temporal constaints networks. In *Proceedings KR '89*, 83–93.

Dung, P. M. 1993. On the Acceptability of Arguments and its Fundamental Role in Nomonotonic Reasoning and Logic Programming. In *Proc. of the 13th. IJCAI 93.*, 852–857.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence* 77(2):321–358.

Jakobovits, H. 1999. Robust semantics for argumentation frameworks. *Journal of Logic and Computation* 9(2):215–261.

Mann, N., and Hunter, A. 2008. Argumentation using temporal knowledge. In *Proc. of 2nd International Conference on Computational Models of Argument (COMMA 2008)*, 204–215.

Martínez, D. C.; García, A. J.; and Simari, G. R. 2007. Modelling well-structured argumentation lines. In *Proc. of XX IJCAI-2007.*, 465–470.

Meiri, I. 1992. Combining qualitative and quantitative contraints in temporal reasoning. In *Proceedings of AAAI '92*, 260–267.

Vreeswijk, G. A. W. 1997. Abstract argumentation systems. *Artificial Intelligence* 90(1–2):225–279.