Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

Formal Analysis of Dialogues on Infinite Argumentation Frameworks

Francesco Belardinelli
Laboratoire IBISC
Université d'Evry, France
belardinelli@ibisc.fr

Davide Grossi
Department of Computer Science
University of Liverpool, UK
d.grossi@liverpool.ac.uk

Nicolas Maudet
Sorbonne Universités
UPMC Univ Paris 06
CNRS, UMR 7606, LIP6
F-75005, Paris, France
nicolas.maudet@lip6.fr

Abstract

The paper analyses multi-agent strategic dialogues on possibly infinite argumentation frameworks. We develop a formal model for representing such dialogues, and introduce FO_A-ATL, a first-order extension of alternating-time logic, for expressing the interplay of strategic and argumentation-theoretic properties. This setting is investigated with respect to the model checking problem, by means of a suitable notion of bisimulation. This notion of bisimulation is also used to shed light on how static properties of argumentation frameworks influence their dynamic behaviour.

1 Introduction

The dialectical and dynamic dimensions of argumentation have been object of scrutiny since the inception of Dung's abstract argumentation theory (cf. [Dung, 1994; 1995]). However, while the definition and analysis of 'static' justifiability criteria (i.e., argumentation semantics [Baroni *et al.*, 2011]) has come to form the bulk of abstract argumentation theory, comparatively little work within Dung's framework has been dedicated to a systematic study of forms of dynamic and multi-agent interaction. Some research has focused on operationalizations of argumentation semantics via two-player games (see [Modgil and Caminada, 2009] for an overview), while some other has attempted an analysis of strategic behavior in abstract forms of argumentation games (in particular [Procaccia and Rosenschein, 2005; Riveret *et al.*, 2010; Thimm and Garcia, 2010]). This paper pursues further the understanding of multi-agent argumentation over abstract argumentation frameworks (AF) capitalizing on techniques from logic and multi-agent verification.

Contribution of the paper. The paper focuses on the formal analysis of multi-agent strategic interactions on possibly infinite argumentation frameworks. Agents are assumed to exchange arguments from possibly infinite AF. They hold private argumentation frameworks representing their 'views' on how arguments attack one another. They interact by taking turns and attacking relevant arguments expanding the framework underlying the interaction. This set up, which we call dynamic argumentation systems (DAS), is general enough to model a wide range of dialogue protocols and games on abstract AF. We analyse this setting formally by introducing FO_A-ATL, a novel first-order extension of the (turn-based) game logic ATL [Alur *et al.*, 2002]. This logic allows us to specify dynamic properties of strategic interactions in argumentation. The objectives of the paper consist in: *(i)* the development of techniques to tackle the model-checking problem of FO_A-ATL; *(ii)* the development of techniques to analyse how static properties of argumentation frameworks influence their dynamic behavior. In other words, we are interested in isolating a class of AF for which their structure allows us to predict their dynamic behaviours. We provide preliminary positive results to both questions.

Technically, the paper models DAS as a special type of infinite-state data-aware systems [Deutsch *et al.*, 2009; Hariri *et al.*, 2013]. This allows us to capitalize on recent results on the formal verification of artifact-centric systems [Belardinelli *et al.*, 2014], thus obtaining truth-preserving bisimulations for FO_A-ATL.

Related work. The paper contributes to several current strands of research in abstract argumentation.

Dynamics of argumentation. How argumentation framework may change by performing operations on their structure has been object of several recent works (e.g., [Baumann, 2012; Bisquert *et al.*, 2013; Booth *et al.*, 2013; Doutre *et al.*, 2014]). However, all mentioned papers assume finite AF, as they rely on the possibility of encoding them into propositional formulas [Besnard and Doutre, 2004]. Neither do they incorporate agency, as they analyse argumentation dynamics from a belief revision paradigm. Our contribution relaxes the finiteness assumption and models agents explicitly as 'protocols' dinamically modifying the structure of AF.

Infinite argumentation frameworks. The bulk of known results in abstract argumentation typically pertain to finite AF. However, infinite AF are gaining attention and have been object of several recent contributions [Baroni *et al.*, 2012; 2013; Baumann and Spanring, 2015], which essentially focus on how known results for the finite case generalize to the infinite. Allowing an infinity of arguments is critical in applications where upper bounds on the number of available arguments cannot be established a priori. Our paper contributes to the understanding of infinite AF.

Logics for abstract argumentation. Recently, several formalizations of argumentation theory have been put forward (e.g., [Caminada and Gabbay, 2009; Grossi, 2010] for early contributions). These works typically focus on finding logical languages (from modal to many-valued logics) that are sufficiently expressive to represent argumentation semantics. Our focus here is rather to specify the strategic abilities of agents engaging in a dialogue/dispute, such as: 'the proponent is able to respond to all attacks by maintaining a conflict-free set of arguments' or 'the opponent has a strategy to force proponent to run out of arguments'. In this respect the paper is a first contribution to the specification of multi-agent argumentation frameworks by means of temporal logics, as well as to their formal verification (cf. [Lomuscio *et al.*, 2009]).

Outline of the paper. Section 2 introduces the dynamics of (multi-agent) argumentation frameworks and the specification language FO_A-ATL; then we state the corresponding model checking problem. Section 3 contains the main technical result, namely, bisimilar dynamic argumentation systems (DAS) satisfy the same formulas in FO_A-ATL. We also investigate the impact of static features of DAS on their dynamics. We conclude in Section 4 and point to future work. For reason of space all proofs are omitted.

2 The Dynamics of AF

In this section we introduce dynamic argumentation systems (DAS). Then, we present FO_A-ATL, a first-order version of the alternating-time temporal logic ATL [Alur *et al.*, 2002], and state the corresponding model checking problem. We first present the basic terminology to be used in the paper.

2.1 Abstract Argumentation

The theoretical setting is built on abstract argumentation, as introduced in [Dung, 1994; 1995]. In what follows we assume a finite set $Ag = \{a_0, \ldots, a_n\}$ of names for agents.

Definition 1 (Argumentation Framework) *Given a set Ag of agent names, a* (multi-agent) argumentation framework *is a tuple $\mathcal{A} = \langle A, \{\leftarrow_a\}_{a \in Ag}\rangle$ s.t.* (i) *A is a (possibly infinite) set of arguments, and* (ii) *for every $a \in Ag$, $\leftarrow_a \subseteq A^2$ is an attack relation between arguments.*

Notice that Def. 1 allows argumentation frameworks, or AF, that include infinitely many arguments. This choice, while providing greater modelling flexibility, also reflects the fact that upper bounds on the number of arguments available to each agent cannot be easily established in general.

We define $\mathcal{F}(A, Ag)$ as the set of all AF built on sets A of arguments and Ag of agent names. Hereafter, we simply write $\mathcal{F}(Ag)$ whenever A is clear. Also, given set $Ag = \{p, o\}$ containing the proponent p and opponent o, $\mathcal{F}(p)$ (resp. $\mathcal{F}(o)$) are shorthands for $\mathcal{F}(\{p\})$ (resp. $\mathcal{F}(\{o\})$), i.e., the set of AF for proponent p (resp. opponent o). Finally, we define the (unindexed) attack relation $\leftarrow = \bigcup_{a \in Ag} \leftarrow_a$.

To express relevant properties of AF, we introduce FO_A, a first-order language geared towards expressing properties of attack relations. Let A_a, for $a \in Ag$, be binary predicate symbols, $P_0, P_1, \ldots$ unary predicate symbols, and *Var* a countable set of individual variables. We define first-order formulas φ and the set $fr(\varphi)$ of free variables by mutual recursion.

Definition 2 (FO_A) *The formulas φ and the set of free variables $fr(\varphi)$ are defined as follows:*

- *if P is a predicate symbol and $x \in Var$ a variable, then $P(x)$ is a formula with $fr(P(x)) = \{x\}$;*
- *if φ is a formula, then also $\neg\varphi$ is, and $fr(\neg\varphi) = fr(\varphi)$;*
- *if φ and φ' are formulas with $fr(\varphi) = fr(\varphi')$, then also $\varphi \to \varphi'$ is, and $fr(\varphi \to \varphi') = fr(\varphi) = fr(\varphi')$;*
- *if φ is a formula with $fr(\varphi) = \{y\}$, then also $\forall y(A_a(y, x) \to \varphi)$ and $\forall y\varphi$ are, with $fr(\forall y(A_a(y, x) \to \varphi)) = \{x\}$ and $fr(\forall y\varphi) = \emptyset$.*

In the following we use the standard abbreviations $\wedge$, $\vee$, and $\exists$. Specifically, $\exists y(A_a(y, x) \wedge \varphi)$ (resp. $\exists y\varphi$) is a shorthand for $\neg\forall y(A_a(y, x) \to \neg\varphi)$ (resp. $\neg\forall y\neg\varphi$). Notice that every FO_A formula φ has at most one free variable.

Finally, we extend argumentation frameworks with interpretations π that assign a subset of A to each predicate symbol P, i.e., $\pi(P) \subseteq A$. An *interpreted argumentation framework* is defined as a couple $(\mathcal{A}, \pi)$. We can now introduce the semantics of our first-order language.

Definition 3 (Semantics of FO_A) *We define whether an argument $u \in A$* satisfies *an FO_A-formula φ in an interpreted AF $(\mathcal{A}, \pi)$, or $(\mathcal{A}, \pi, u) \models \varphi$, as follows (clauses for propositional connectives are straightforward, thus omitted):*

$$
\begin{array}{ll}
(\mathcal{A}, \pi, u) \models P(x) & \textit{iff } u \in \pi(P) \\
(\mathcal{A}, \pi, u) \models \forall y(A_a(y, x) \to \psi) & \textit{iff for every } v \in A, \\
 & \quad u \leftarrow_a v \textit{ implies } (\mathcal{A}, \pi, v) \models \psi \\
(\mathcal{A}, \pi, u) \models \forall y\psi & \textit{iff for every } v \in A,\ (\mathcal{A}, \pi, v) \models \psi
\end{array}
$$

An FO_A-formula φ is *true* in an interpreted AF $(\mathcal{A}, \pi)$, or $(\mathcal{A}, \pi) \models \varphi$, iff $(\mathcal{A}, \pi, u) \models \varphi$ for every argument $u \in A$; φ is *valid* in $\mathcal{A}$, or $\mathcal{A} \models \varphi$, iff $(\mathcal{A}, \pi) \models \varphi$ for every interpretation π.

From a technical point of view, the language FO_A is the dyadic fragment of first-order logic with one free variable.[1] Table 1 shows how this simple logic suffices to formalize several of the key notions from [Dung, 1995] (see also [Grossi, 2010]).

2.2 Dynamic Argumentation Frameworks

In this paper we are interested in analysing dialogues modeled as multi-agent processes over argumentation frameworks. As dialogues progress agents build a 'shared' (or 'public') argumentation framework which is used to evaluate the available arguments.

Agents and global states

To introduce the dynamics of argumentation frameworks we start with the notion of *agent*.

Definition 4 (Agent) *Given sets Ag of agent names and A of arguments, an* agent *is a tuple $a = \langle \mathcal{A}, Act, Pr\rangle$ where*

- *$\mathcal{A} \in \mathcal{F}(a)$ is the agent's* argumentation framework*;*
- *the set Act of* actions *contains action $attack(x, x')$ to attack argument x' with argument x, and action skip;*

[1] As such, it is equivalent to the multi-modal logic K with the universal modality [Blackburn *et al.*, 2001].

$\pi(P)$ is conflict-free in $\mathcal{A}$	iff	$(\mathcal{A},\pi) \models \forall x(P(x) \rightarrow \neg(\exists y(A(y,x) \wedge P(y)))$	$CFr(P)$
$\pi(P)$ is acceptable in $\mathcal{A}$	iff	$(\mathcal{A},\pi) \models \forall x(P(x) \rightarrow \forall y(A(y,x) \rightarrow \exists z A(z,y) \wedge P(z)))$	$CFree(P)$
$\pi(P)$ is admissible in $\mathcal{A}$	iff	$\pi(P)$ is conflict-free and acceptable	$Adm(P)$
$\pi(P)$ is complete in $\mathcal{A}$	iff	$\pi(P)$ is conflict-free and $(\mathcal{A},\pi) \models \forall x(P(x) \leftrightarrow \forall y(A(y,x) \rightarrow \exists z A(z,y) \wedge P(z)))$	$Cmp(P)$
$\pi(P)$ is a stable in $\mathcal{A}$	iff	$(\mathcal{A},\pi) \models \forall x(P(x) \leftrightarrow \neg(\exists y(A(y,x) \wedge P(y)))$	$Stb(P)$

Table 1: Some key notions of abstract argumentation theory and their formalization in FO_A (A is interpreted over $\leftarrow$).

- *Pr* : $\bigcup_{A' \subseteq A} \mathcal{F}(A', Ag) \mapsto 2^{Act(A)}$ *is the* local protocol function*, where* $Act(A)$ *is the set of* ground actions $\alpha(\vec{u})$*, and for every* $\mathcal{A}' \in \mathcal{F}(A', Ag)$*,* $attack(u,u') \in Pr(\mathcal{A}')$ *only if* $u' \in A'$ *and* $u' \leftarrow_a u$ *holds in* $\mathcal{A}$*; while the* skip *action is always enabled.*

By Def. 4 agent a's local state is modelled as an argumentation framework $\mathcal{A}$, and she is assumed to perform the actions in *Act* according to protocol *Pr*. In particular, by definition of *Pr*, attacks must be relevant, i.e., the attacked argument u' has to appear in the current argumentation framework $\mathcal{A}'$, and they must reflect actual attacks available in the agent's AF $\mathcal{A} \in \mathcal{F}(a)$. These assumptions on protocols model the basic requirements of relevance and truthfulness in dialogues.

To model agent interaction we need to define their composition, beginning with the notion of global state.

Definition 5 (Global State) *Given a set* Ag *of agents* $a_i = \langle \mathcal{A}_i, Act_i, Pr_i \rangle$ *defined on the same (possibly infinite) set* A *of arguments, a* global state *is a couple* (s,a) *where* (i) $s \in \mathcal{F}(A', Ag)$ *is an argumentation framework for some* $A' \subseteq A$*; and* (ii) $a \in Ag$.

In Def. 5 all agents are defined on the same set of arguments. However, some literature on agents and argumentation suppose that each agent is endowed with a distinct set of arguments (e.g., [Rahwan and Larson, 2011]). We remark that our requirement is not a limitation because, firstly we can always consider the union of the sets of arguments for each agent, and secondly the characterising feature of AF is really the attack relation. Moreover, this assumption simplifies the presentation hereafter. Also, in a state (s,a), a is the *active* agent, i.e., the agent to act next; it will be omitted whenever clear from the context. We write $\mathcal{G}$ to denote the set of all global states. Finally, any set $A' \subseteq A$ can be seen as a global state where, for every $a \in Ag$, $\leftarrow_a$ is empty.

Argument Dynamics

We now introduce the dynamics of argumentation frameworks. In the rest of the paper we suppose that the only two agents are proponent p and opponent o. This assumption can be lifted but it crucially simplifies the presentation of the key (conceptual and technical) contributions of the paper.

Definition 6 (DAS) *Given set* $Ag = \{p, o\}$ *of agents defined on the same (possibly infinite) set* A *of arguments, a* dynamic argumentation system *is a tuple* $\mathcal{P} = \langle Ag, I, \tau, \pi \rangle$ *where*

- $I \subseteq A \times \{o\}$ *is the set of* initial global states (s_0, o)*;*
- $\tau : \mathcal{G} \times (Act_p(A) \cup Act_o(A)) \mapsto \mathcal{G}$ *is the* transition function*, where* $\tau((s,a), attack_{a'}(\vec{u}))$ *is defined iff* $a = a'$ *and* $attack_{a'}(\vec{u}) \in Pr_{a'}(s)$*. Moreover,* $(s', a') = \tau((s,a), attack(u,u'))$ *iff* $a' \neq a$ *and* $s' = \langle A', \leftarrow' \rangle$ *for* $A' = A \cup \{u\}$ *and* $\leftarrow'_a = \leftarrow'_a \cup \{(u', u)\}$*; while* $(s', a') = \tau((s,a), \textsf{skip})$ *iff* $a' \neq a$ *and* $s' = s$*.*
- π *is an interpretation of predicate symbols* P *as above.*

A DAS evolves from an initial state $(s_0, o) \in I$ as specified by the transition function τ, which returns a successor state for the current state and selected action. DAS are turn-based, with each agent taking one action at a time. Hereafter we assume that the opponent always moves first, that is, I is intuitively the set of arguments supported by the proponent. One key feature of DAS is that, since the set A of arguments is infinite in general, they are infinite-state systems normally. Indeed, as the dialogue unfolds, agents can introduce an infinite number of arguments from A, thus going through infinitely many states during the system's execution.

We now fix some notation. The *transition relation* $\rightarrow_a$ is defined so that $s \rightarrow_a s'$ iff for some action $\alpha_a(\vec{u}) \in Act_a(A)$, for $a \in \{o, p\}$, we have that $s \xrightarrow{\alpha_a(\vec{u})} s'$, i.e., $\tau((s,a), \alpha_a(\vec{u})) = (s', a')$ with $a' \neq a$. A *run* λ from a state (s,a), or (s,a)-run, is an infinite sequence $s^0 \rightarrow_a s^1 \rightarrow_{a'} \ldots$, s.t. $s^0 = s$. Notice that, since agents o and p take turns in performing actions, the sequence of agents in an (s,a)-run is uniquely determined by a, so we omit agents whenever these are clear by the context. Hence, we will often denote a state (s,a) simply as s, whenever agent a is understood. For $n, m \in \mathbb{N}$, with $n \leq m$, we define $\lambda(n) = s^n$ and $\lambda[n,m] = s^n, s^{n+1}, \ldots, s^m$. A state s' is *reachable from* s iff for some s-run λ and $i \geq 0$, $s' = \lambda(i)$. Let $\mathcal{S}$ be the set of states reachable from any initial state $s_0 \in I$. Notice also that the transition relation is serial, as each agent has a skip action enabled at each local state.

DAS, dialogues and games

It is worth comparing DAS briefly with known structures in the literature on dialogue games for abstract argumentation. In the terminology of [Thimm and Garcia, 2010] DAS can be viewed as defining a dialectical game protocol, where at each step an active agent attacks some of the available arguments. DAS can be seen as extensive *game forms* with infinite horizon (though they are not games, as they do not specify payoffs). They also generalize the structures underpinning the extensive-form argument games of [Procaccia and Rosenschein, 2005] by allowing agents to hold an infinite argumentation framework and to attack any of the available arguments, not only the last one uttered by some other agent. We finally show how standard dialogue games (e.g., for the grounded extension [Dung, 1994] or for credulous admissibility [Vreeswijk and Prakken, 2000]) can be modeled as DAS.

Example 1 In the game for the grounded extension two

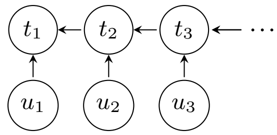


Figure 1: An infinite AF: each u_i and t_{i+1} attack each t_i.

agents o and p hold the same private argumentation framework, i.e., $\mathcal{A}_o = \mathcal{A}_p$. These games are two-player zero-sum perfect information games with possibly infinite horizon. We can view them as DAS. Consider for instance the AF in Figure 1. The DAS corresponding to the game for grounded played on this AF, starting at argument t_1 can be defined as follows. Fix $I = \{t_1\}$. For both agents we can define the following protocol: if the current framework contains t_i then attack t_i with u_i or t_{i+1}, otherwise skip. This protocol encodes the 'legal' moves of o and p according to the game for the grounded extension. The possible runs of this DAS contain all possible sub-graphs of the above AF generated from t_1, e.g.: $(\{t_1\}, \emptyset), (\{t_1, u_1\}, \{(u_1, t_1)\}), \mathsf{skip}, \ldots$; or $(\{t_1\}, \emptyset)$, $(\{t_1, t_2\}, \{(t_2, t_1)\}), \ldots, (\{t_1, \ldots, t_i\}, \{(t_i, t_{i-1})\}), \ldots$. These runs correspond to possible dialogues in the game for the grounded extension. ■

Given a DAS $\mathcal{P}$ on a (possibly infinite) set A of arguments, we define the corresponding (joint) argumentation framework $\mathcal{A}_\mathcal{P} = \langle A, \{\leftarrow_a\}_{a \in Ag}\rangle$ so that $u \leftarrow_a u'$ holds in $\mathcal{A}_\mathcal{P}$ iff $u \leftarrow_a u'$ holds in the AF $\mathcal{A}_a$ for agent $a \in Ag$. By the definition of protocols (Def. 4) it is apparent that every reachable state in $\mathcal{P}$ is a subgraph of $\mathcal{A}_\mathcal{P}$, that is,

if $u \leftarrow_a v$ holds in some $s \in \mathcal{S}$, for some $a \in Ag$, then $u \leftarrow_a v$ holds in $\mathcal{A}_\mathcal{P}$. $(*)$

Thus, states in $\mathcal{P}$ are truthful, yet partial, representations of $\mathcal{A}_\mathcal{P}$. However, the converse of $(*)$ does not hold in general, i.e., $\mathcal{P}$ needs not to include all subgraphs of $\mathcal{A}_\mathcal{P}$ as states. To appreciate this consider the following:

Example 2 [Example 1 continued] Now endow o with a more restrictive protocol: if the current framework contains t_i then attack t_i with u_i, otherwise skip. This protocol makes o play more rationally, selecting arguments to which p cannot reply. The only possible run of this DAS is: $(\{t_1\}, \emptyset), (\{t_1, u_1\}, \{(u_1, t_1)\}), \mathsf{skip}, \ldots$. ■

This asymmetry motivates the next definition. An agent is *naive* if its protocol allows it to move any available attack (cp. this to the more restrictive protocol of o in Example 1).

Definition 7 (Naive Agent) *An agent a is* naive *iff for every $\mathcal{A}' \in \mathcal{F}(A', Ag)$, $attack(u, u') \in Pr(\mathcal{A}')$ iff $u' \in A'$ and $u' \leftarrow_a u$ holds in $\mathcal{A}_a$.*

Intuitively, naive agents perform every possible attack available to them, irrespectively of the strategic behaviour of other agents. A DAS $\mathcal{P}$ is *naive* iff every agent in Ag is naive. Then, we remark that for naive DAS the converse of $(*)$ actually holds.

2.3 The Specification Language FO_A-ATL

We now introduce a formal language to specify properties of interest of DAS. Arguments in DAS call for the use of first-order logic, whereas strategy operators are needed to account for agents' behaviour. To the best of our knowledge, no logic combining these two features has ever been studied.

Definition 8 (FO_A-ATL) *The FO_A-ATL formulas φ, with free variables $fr(\varphi)$, are defined as follows:*

- *if ψ is an FO_A-formula, then it is an FO_A-ATL formula;*
- *if φ is a formula and $N \subseteq Ag$, then $\neg\varphi$, $\langle\langle N\rangle\rangle X\varphi$, and $\langle\langle N\rangle\rangle G\varphi$ are formulas with $fr(\neg\varphi) = fr(\langle\langle N\rangle\rangle X\varphi) = fr(\langle\langle N\rangle\rangle G\varphi) = fr(\varphi)$;*
- *if φ and φ' are formulas with $fr(\varphi) = fr(\varphi')$ and $N \subseteq Ag$, then $\varphi \rightarrow \varphi'$ and $\langle\langle N\rangle\rangle \varphi U \varphi'$ are formulas with $fr(\varphi \rightarrow \varphi') = fr(\langle\langle N\rangle\rangle \varphi U \varphi') = fr(\varphi) = fr(\varphi')$;*
- *if φ is a formula with $fr(\varphi) = \{y\}$, then $\forall y(A_a(y, x) \rightarrow \varphi)$ and $\forall y \varphi$ are formulas with $fr(\forall y(A_a(y, x) \rightarrow \varphi)) = \{x\}$ and $fr(\forall y \varphi) = \emptyset$.*

The language FO_A-ATL is a first-order extension of the alternating-time temporal logic ATL [Alur *et al.*, 2002]. The FO_A-ATL formulas *(i)* $\langle\langle N\rangle\rangle X\varphi$, *(ii)* $\langle\langle N\rangle\rangle G\varphi$ and *(iii)* $\langle\langle N\rangle\rangle \varphi U \varphi'$ are read as "the agents in N have a strategy to..." *(i)* "...enforce φ at the next state", *(ii)* "...always enforce φ", and *(iii)* "...enforce φ until φ'".

To intepret FO_A-ATL formulas on DAS we need to introduce the notion of a *strategy* for a set N of agents. An *N-strategy* is a mapping $f_N : \mathcal{S}^+ \mapsto \bigcup_{a \in N} Act_a(A)$ s.t. $f_N(\kappa \cdot (s, a)) \in Pr_a(s)$ for every $\kappa \in \mathcal{S}^*$. Intuitively, a strategy returns an enabled action in $Act_a(A)$ for every non-empty, finite sequence of states in $\mathcal{S}^+$. We remark that, according to standard terminology in concurrent game models [Bulling *et al.*, 2010], the agents in DAS have *perfect information* and *perfect recall*, that is, their strategies are determined by all information available at each global state, for all states visited up to the current state. Further, the *outcome* of strategy f_N at state (s, a), or $out((s, a), f_N)$, is the set of all (s, a)-runs λ s.t. for every $b \in N$, $(\lambda(i+1), b') = \tau((\lambda(i), b), f_N(\lambda[0, i]))$ for all $i \geq 0$. As above, we assume that strategies respect agents' turns and simply write $out(s, f_N)$, thus omitting agents whenever the latter are clear by the context.

Definition 9 (Semantics of FO_A-ATL) *We define whether an argument u* satisfies *a formula φ at state s in a DAS $\mathcal{P}$, or $(\mathcal{P}, s, u) \models \varphi$, as follows (clauses for propositional connectives are straightforward and thus omitted):*

$(\mathcal{P}, s, u) \models \psi$ *iff* $(s, \pi, u) \models \psi$, *if ψ is an FO_A-formula*
$(\mathcal{P}, s, u) \models \langle\langle N\rangle\rangle X\varphi$ *iff for some N-strategy f_N, for all $\lambda \in out(s, f_N)$, $(\mathcal{P}, \lambda(1), u) \models \varphi$*
$(\mathcal{P}, s, u) \models \langle\langle N\rangle\rangle G\varphi$ *iff for some N-strategy f_N, for all $\lambda \in out(s, f_N)$, $i \geq 0$, $(\mathcal{P}, \lambda(i), u) \models \varphi$*
$(\mathcal{P}, s, u) \models \langle\langle N\rangle\rangle \varphi U \varphi'$ *iff for some N-strategy f_N, for all $\lambda \in out(s, f_N)$, for some $k \geq 0$, $(\mathcal{P}, \lambda(k), u) \models \varphi'$, and for all j, $0 \leq j < k$ implies $(\mathcal{P}, \lambda(j), u) \models \varphi$*
$(\mathcal{P}, s, u) \models \forall y(A_a(y, x) \rightarrow \varphi)$ *iff for every $v \in s$, $u \leftarrow_a v$ implies $(\mathcal{P}, s, v) \models \varphi$*
$(\mathcal{P}, s, u) \models \forall y \varphi$ *iff for every $v \in s$, $(\mathcal{P}, s, v) \models \varphi$*

A formula φ is *true* at s, or $(\mathcal{P}, s) \models \varphi$, if $(\mathcal{P}, s, u) \models \varphi$ for every arguments $u \in s$; φ is *true* in $\mathcal{P}$, or $\mathcal{P} \models \varphi$, if $(\mathcal{P}, s_0) \models \varphi$ for all $s_0 \in I$.

We illustrate now the expressiveness of FO_A-ATL through some example properties that involve strategic and argumentation-theoretic features.

Example 3 The following formula states that opponent o can force proponent p to run out of moves in the next state:

$$\langle\langle o\rangle\rangle X \forall x \neg \exists y A_p(y, x) \tag{1}$$

This formula is for instance true at argument t_1 in the DAS of Example 1. ■

Example 4 The following formula states that proponent p has a strategy enforcing the set of arguments in P, which includes the current argument, to be conflict-free (respectively, acceptable, admissible, complete, stable):

$$P(x) \wedge \langle\langle p\rangle\rangle G\, \chi(P) \tag{2}$$

where $\chi \in \{Cfr, Acc, Adm, Cmp, Stb\}$.[2] Let us focus for instance on $P(x) \wedge \langle\langle p\rangle\rangle G\ Adm(P)$. This is a desirable requirement for proponent's strategies in dialogue games for the grounded extension (recall Example 1) or credulous admissibility [Vreeswijk and Prakken, 2000]. This statement is false of the DAS of Example 1 (at argument t_1), where o, by moving $attack(u_1, t_1)$, forces p to run out of moves, no matter what interpretation π of P is selected. In fact p has no strategy to force any set of arguments containing t_1 (the initial argument) to be admissible. That is: $P(x) \rightarrow \langle\langle o\rangle\rangle G\ \neg Adm(P)$ or, equivalently, $P(x) \rightarrow \neg\langle\langle p\rangle\rangle G\ Adm(P)$ for any P.[3] ■

3 Verifying Dynamic Argumentation Systems

In Section 2 we introduced DAS as a general model for (abstract) argument-based multi-agent interaction. We tailored an influential game logic in order to provide DAS with the specification language FO_A-ATL. We are then able to define a model checking problem whereby strategic and argumentation-theoretic properties, such as those expressed in (1) and (2) above, can be checked against a given DAS.

Definition 10 (Model Checking Problem) *Given a DAS $\mathcal{P}$ and an FO_A-ATL-formula φ, determine whether for every $s_0 \in I$, $(\mathcal{P}, s_0, u_0) \models \varphi$ for some argument u_0.*

In case that φ is a sentence with no free variable, the model checking problem reduces to verify whether $\mathcal{P} \models \varphi$.

Model checking general data-aware systems is known to be undecidable [Deutsch *et al.*, 2007]. In [Belardinelli *et al.*, 2012; 2014] the same problem is proved decidable for *bounded* and *uniform* systems. Without going into details, we remark that DAS do not normally satisfy these requirements. Therefore new techniques need to be developed.

In this section we introduce a notion of bisimulation to explore under which circumstances two DAS satisfy the same formulas. In particular, we show that bisimilar DAS satisfy the same FO_A-ATL formulas. Then, we investigate how the static properties of AF impact on the dynamics of DAS. This kind of results is key to tackle the model checking problem, as they allow to verify a DAS $\mathcal{P}$ by model checking a bisimilar DAS $\mathcal{P}'$.

[2] Recall the properties in Table 1.

[3] It is a known validity of turn-based asynchronous ATL that $\langle\langle N\rangle\rangle G\phi \leftrightarrow \langle\langle Ag\backslash N\rangle\rangle\neg\phi$ [Alur *et al.*, 2002].

3.1 Bisimulations for Argument Frameworks

A notion of bisimulation can naturally be defined on AF (cf. [Grossi, 2010]).

Definition 11 (Static Bisimulation) *Let $(\mathcal{A}, \pi) = \langle A, \{\leftarrow_a\}_{a\in Ag}, \pi\rangle$ and $(\mathcal{A}', \pi') = \langle A', \{\leftarrow'_a\}_{a\in Ag}, \pi'\rangle$ be interpreted AF defined on a set Ag of agents.* A static bisimulation *is a relation $S \subseteq A \times A'$ s.t. for $u \in A$, $u' \in A'$, $S(u, u')$ implies*

(i) *for every predicate symbol P, $u \in \pi(P)$ iff $u' \in \pi'(P)$;*

(ii) *for every $v \in A$, if $u \leftarrow_a v$ then for some $v' \in A'$, $u' \leftarrow'_a v'$ and $S(v, v')$;*

(iii) *for every $v' \in A'$, if $u' \leftarrow'_a v'$ then for some $v \in A$, $u \leftarrow_a v$ and $S(v, v')$.*

Two arguments $u \in A$, $u' \in A'$ are *bisimilar*, or $u \simeq u'$, iff $S(u, u')$ for some static bisimulation S. Finally, two AF $\mathcal{A}$ and $\mathcal{A}'$ are *statically bisimilar* (or simply bisimilar) iff *(i)* for every $u \in A$, $u \simeq u'$ for some $u' \in A'$; and *(ii)* for every $u' \in A'$, $u' \simeq u$ for some $u \in A$. We denote this as $\mathcal{A} \simeq \mathcal{A}'$.

We can now show that bisimilar states satisfy exactly the same FO_A-formulas.

Lemma 1 *Given bisimilar interpreted AF $(\mathcal{A}, \pi)$ and $(\mathcal{A}', \pi')$, and bisimilar arguments $u \in A$ and $u' \in A'$, then for every FO_A-formula φ,*

$$(\mathcal{A}, \pi, u) \models \varphi \quad \textit{iff} \quad (\mathcal{A}', \pi', u') \models \varphi$$

As a result, bisimilar AF cannot be distinguished by FO_A-formulas. In the following we explore the conditions under which this applies to DAS as well. In particular, we say that DAS $\mathcal{P}$ and $\mathcal{P}'$ are *statically bisimilar* iff $\mathcal{A}_{\mathcal{P}} \simeq \mathcal{A}_{\mathcal{P}'}$ for some static bisimilation that maps initial states into initial states.

3.2 Bisimulations for DAS

We first introduce a notion of dynamic bisimulation, and then explore its properties in the context of DAS. In the rest of the section we let $\mathcal{P} = \langle Ag, I, \tau, \pi\rangle$ and $\mathcal{P}' = \langle Ag, I', \tau', \pi'\rangle$ be two DAS defined on the same set Ag of agent names. Notice that, albeit agents may have the same name, they might differ as to their argumentation frameworks, actions, or protocols.

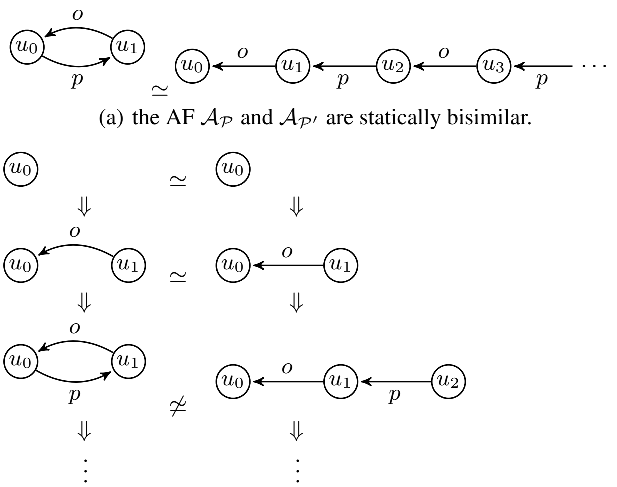
Definition 12 (Dynamic Simulation) *Given DAS $\mathcal{P}$ and $\mathcal{P}'$, a* dynamic simulation *is a relation $R \subseteq \mathcal{S} \times \mathcal{S}'$ s.t. for $s \in \mathcal{S}$, $s' \in \mathcal{S}'$, $R(s, s')$ implies:*

1. *$s \simeq s'$ for some static bisimulation S;*
2. *for every $t \in \mathcal{S}$, if $s \rightarrow_a t$ then for some $t' \in \mathcal{S}'$, $s' \rightarrow'_a t'$, $t \simeq t'$ for some bisimulation $S' \supseteq S$, and $R(t, t')$.*

In Def. 12 we implicitly assume that the simulation relation relates states with turns for the same active agent in Ag. Simulations can then be naturally extended to bisimulations.

Definition 13 (Dynamic Bisimulation) *A relation $D \subseteq \mathcal{S} \times \mathcal{S}'$ is a* dynamic bisimulation *iff both D and $D^{-1} = \{\langle s', s\rangle \mid D(s, s')\}$ are dynamic simulations.*

Two states $s \in \mathcal{S}$ and $s' \in \mathcal{S}'$ are *dynamically bisimilar*, or $s \approx s'$, iff $D(s, s')$ for some bisimulation relation D. It can be shown that $\approx$ is the largest dynamic bisimulation, and an equivalence relation, on $\mathcal{S} \cup \mathcal{S}'$. DAS $\mathcal{P}$ and $\mathcal{P}'$ are *dynamically bisimilar*, or $\mathcal{P} \approx \mathcal{P}'$, iff *(i)* for every $s_0 \in I$, $s_0 \approx s'_0$



(a) the AF $\mathcal{A}_{\mathcal{P}}$ and $\mathcal{A}_{\mathcal{P}'}$ are statically bisimilar.

Figure 2: the DAS $\mathcal{P}$ and $\mathcal{P}'$ are statically bisimilar, but not dynamically bisimilar.

for some $s'_0 \in I'$, and *(ii)* for every $s'_0 \in I'$, $s_0 \approx s'_0$ for some $s_0 \in I$. Notice that arguments u and u' are or are not bisimilar always w.r.t. some states $s \in \mathcal{S}$ and $s \in \mathcal{S}'$. We state this explicitly by saying that $u \simeq u'$ w.r.t. s and s'. In particular, by Def. 13, if $s \approx s'$ and $u \simeq u'$ w.r.t. s and s', then u and u' are still bisimilar in all subsequent bisimilar states. This is a key feature for proving preservation of satisfaction for FO_A-ATL formulas.

The first result we prove on bisimulations shows that being statically bisimilar does not imply dynamic bisimilarity, not even in the case of naive agents.

Lemma 2 *Static bisimilarity does not imply dynamic bisimilarity, that is, there exist naive, statically bisimilar DAS $\mathcal{P}$ and $\mathcal{P}'$ such that $\mathcal{P} \not\approx \mathcal{P}'$.*

In Fig. 2 we report DAS $\mathcal{P}$ and $\mathcal{P}'$, whose underlying AF are statically bisimilar, but that are not dynamically bisimilar.

We now show that dynamically bisimilar states satisfy the same FO_A-ATLformulas.

Theorem 3 *Suppose that $s \approx s'$, and $u \simeq u'$ w.r.t. s and s'. Then for every FO_A-ATL formula φ,*

$$(\mathcal{P}, s, u) \models \varphi \quad \textit{iff} \quad (\mathcal{P}', s', u') \models \varphi$$

As a result, dynamic bisimulations preserve the satisfaction of FO_A-ATL formulas. In particular, we can tackle the problem of model checking a DAS $\mathcal{P}$ by verifying a bisimilar DAS $\mathcal{P}'$, and then transfering the result by Theorem 3. In particular, DAS $\mathcal{P}'$ may exhibit nice structural properties (such as being finite) that can make the verification task feasible.

3.3 From Static Properties to Dynamics

In this section we make use of bisimulations to explore how the static properties of DAS determine their dynamic features. In Lemma 2 we showed that this relationship is not straightforward, not even in the case of naive DAS. However, in some specific cases there is indeed a correspondence. The first result reduces dynamic bisimulations to static bisimulations, together with some constraints on temporal transitions. As usual, we assume that turns are respected.

Theorem 4 *Let $\mathcal{P}$ and $\mathcal{P}'$ be DAS. Suppose that $\mathcal{P}'$ is naive and for every $u \in s \in \mathcal{S}$, $u' \in s' \in \mathcal{S}'$, if $s \simeq s'$, $u \simeq u'$ w.r.t. s and s', and $u \leftarrow_a v$ in $\mathcal{A}_{\mathcal{P}}$ for some $v \in A$, then $u' \leftarrow'_a v'$ in $\mathcal{A}_{\mathcal{P}'}$ for some $v' \in A'$ and either*

1. *$v \in s$ and either* (i) *$v' \in s'$ and $v \simeq v'$ w.r.t. s and s', or* (ii) *$v' \notin s'$ and for no $w \in s$, $v \leftarrow_a w$ in s,*
2. *or $v \notin s$ and either* (i) *$v' \notin s'$, or* (ii) *$v' \in s'$ and for no $w' \in s'$, $v' \leftarrow'_a w'$ in s'.*

Then, $D = \{(s, s') \mid s \simeq s'\}$ is a dynamic simulation between $\mathcal{P}$ and $\mathcal{P}'$.

Notice that the assumptions in Theorem 4 imply that DAS $\mathcal{P}$ and $\mathcal{P}'$ are statically bisimilar. However, they have also to satisfy extra conditions (1) and (2).

The main contribution of Theorem 4 is to show that the notion of dynamic bisimulation can be reduced to static bisimulation, together with some assumptions on the structural properties of AF $\mathcal{A}_{\mathcal{P}}$ and $\mathcal{A}_{\mathcal{P}'}$, in case that we consider naive DAS. Hence, in order to verify DAS we can simply model check statically bisimilar systems, and then transfer the result by using Theorem 4.

It is of interest to analyse DAS that actually satisfy the conditions above. For example, it is easy to check that whenever the underlying AF $\mathcal{A}_{\mathcal{P}}$ and $\mathcal{A}_{\mathcal{P}'}$ of naive DAS $\mathcal{P}$ and $\mathcal{P}'$ are directed acyclic graphs (DAG), where every argument is attacked by some other argument, then the conditions in Theorem 4 do indeed hold. So, for this class of naive DAS structural, static features do determine their dynamic properties.

Corollary 5 *Suppose that DAS $\mathcal{P}$ and $\mathcal{P}'$ are naive and statically bisimilar, and that $\mathcal{A}_{\mathcal{P}}$ and $\mathcal{A}_{\mathcal{P}'}$ are DAG where every argument is attacked by some other argument. Then, $\mathcal{P}$ and $\mathcal{P}'$ are dynamically bisimilar and therefore satisfy the same FO_A-ATL formulas.*

4 Conclusions and Further Work

In many dialectical situations it is difficult to assume beforehand a bound on the number of arguments that agents have at their disposal. While some properties of such infinite argumentation frameworks have recently been investigated, their implication on debates amongst agents had not been analysed before. In this paper we set up and explored the framework of dynamic argumentation systems through a logic lens. We showed that in general *static* bisimilarity is not strong enough to capture equivalence of DAS (in terms of FO_A-ATL formulas they satisfy), and introduced a novel notion of *dynamic* bisimilarity. For some specific structures (in particular acyclic ones), the static notion remains powerful enough, at least for the naive agents studied here. While in this paper we have focused on such agents (unconstrained in their argumentative moves as long as they are relevant and truthful), an interesting direction of research is to investigate how more restricted protocols would also impact the dynamics of DAS. Another noteworthy feature of our framework is that formulas in FO_A-ATL can express the ability for a *group* of agents to ensure, for instance, that some arguments gets accepted–thus paving the way for a logical analysis of multiparty protocols, see *e.g.* [Bonzon and Maudet, 2011].

Acknowledgements.

The third author benefited from the support of the project AMANDE ANR-13-BS02-0004 of the French National Research Agency (ANR).

References

[Alur *et al.*, 2002] R. Alur, T. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49(5):672–713, 2002.

[Baroni *et al.*, 2011] P. Baroni, M. Caminada, and M. Giacomin. An Introduction to Argumentation Semantics. *The Knowledge Engineering Review*, 26(4):365–410, 2011.

[Baroni *et al.*, 2012] P. Baroni, F. Cerutti, P. E. Dunne, and M. Giacomin. Computing with Infinite Argumentation Frameworks: The Case of AFRAs. In *Theory and Applications of Formal Argumentation*, pages 197–214, 2012.

[Baroni *et al.*, 2013] P. Baroni, F. Cerutti, P. E. Dunne, and M. Giacomin. Automata for Infinite Argumentation Structures. *Journal of Artificial Intelligence*, 203:104–150, 2013.

[Baumann and Spanring, 2015] R. Baumann and C. Spanring. Infinite Argumentation Frameworks. In *Advances in Knowledge Representation, Logic Programming, and Abstract Argumentation*, pages 281–295, 2015.

[Baumann, 2012] R. Baumann. What does it take to enforce an argument? Minimal Change in Abstract Argumentation. In *Proceedings of ECAI'12*, pages 127–132, 2012.

[Belardinelli *et al.*, 2012] F. Belardinelli, A. Lomuscio, and F. Patrizi. An Abstraction Technique for the Verification of Artifact-Centric Systems. In *Proc. of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR'12)*, pages 319 – 328, 2012.

[Belardinelli *et al.*, 2014] F. Belardinelli, F. Patrizi, and A. Lomuscio. Verification of Agent-Based Artifact Systems. *Journal of Artificial Intelligence Research*, 51:333–77, 2014.

[Besnard and Doutre, 2004] P. Besnard and S. Doutre. Checking the Acceptability of a Set of Arguments. In *Proceedings of the 10th International Workshop on Non-Monotonic Reasoning (NMR'04)*, pages 59–64, 2004.

[Bisquert *et al.*, 2013] P. Bisquert, C. Cayrol, F. Bannay, and M.-C. Lagasquie-Schiex. Enforcement in Argumentation is a Kind of Update. In *Proceedings of the International Conference on Scalable Uncertainty Management*, pages 30–43, 2013.

[Blackburn *et al.*, 2001] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.

[Bonzon and Maudet, 2011] E. Bonzon and N. Maudet. On the Outcomes of Multiparty Persuasion. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2011)*. IFAAMAS, 2011.

[Booth *et al.*, 2013] R. Booth, S. Kaci, T. Rienstra, and L. van der Torre. A logical theory about dynamics in abstract argumentation. In *Proceedings of the International Conference on Scalable Uncertainty Management*, pages 148–161. Springer, 2013.

[Bulling *et al.*, 2010] N. Bulling, J. Dix, and W. Jamroga. Model Checking Logics of Strategic Ability: Complexity*. In *Specification and Verification of Multi-agent Systems*, pages 125–159. Springer US, 2010.

[Caminada and Gabbay, 2009] M. Caminada and D. Gabbay. A Logical Account of Formal Argumentation. *Studia Logica*, 93(2):109–145, 2009.

[Deutsch *et al.*, 2007] A. Deutsch, L. Sui, and V. Vianu. Specification and Verification of Data-Driven Web Applications. *Journal of Computer and System Science*, 73(3):442–474, 2007.

[Deutsch *et al.*, 2009] A. Deutsch, R. Hull, F. Patrizi, and V. Vianu. Automatic Verification of Data-Centric Business Processes. In *Proc. of International Conference on Database Theory*, 2009.

[Doutre *et al.*, 2014] S. Doutre, A. Herzig, and L. Perrussel. A Dynamic Logic Framework for Abstract Argumentation. In *Proc. of the 14th International Conference on Principles of Knowledge Representation and Reasoning (KR'14)*, 2014.

[Dung, 1994] P. Dung. Logic Programming as Dialogue Games. Technical report, Division of Computer Science, Asian Institute of Technology, 1994.

[Dung, 1995] P. Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games. *Artificial Intelligence*, 77(2):321–358, 1995.

[Grossi, 2010] D. Grossi. On the Logic of Argumentation Theory. In *Proc. of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, pages 409–416. IFAAMAS, 2010.

[Hariri *et al.*, 2013] B. Bagheri Hariri, D. Calvanese, G. De Giacomo, A. Deutsch, and M. Montali. Verification of Relational Data-centric Dynamic Systems with External Services. In *Proc. of the Symposium on Principles of Database Systems*, pages 163–174, 2013.

[Lomuscio *et al.*, 2009] A. Lomuscio, H. Qu, and F. Raimondi. MCMAS: A Model Checker for the Verification of Multi-Agent Systems. In *Proc. of the International Conference on Computer-Aided Verification*, pages 682–688, 2009.

[Modgil and Caminada, 2009] S. Modgil and M. Caminada. Proof Theories and Algorithms for Abstract Argumentation Frameworks. In *Argumentation in Artificial Intelligence*, pages 105–132. Springer, 2009.

[Procaccia and Rosenschein, 2005] A. Procaccia and J. Rosenschein. Extensive-From Argumentation Games. In *Proc. of European Conference on Multi-Agent Systems*, pages 312–322, 2005.

[Rahwan and Larson, 2011] I. Rahwan and K. Larson. Logical Mechanism Design. *The Knowledge Engineering Review*, 26(1):61–69, 2011.

[Riveret *et al.*, 2010] R. Riveret, H. Prakken, A. Rotolo, and G. Sartor. Heuristics in Argumentation: A Game Theory Investigation. In *Proceedings of COMMA'10*, pages 324–335, 2010.

[Thimm and Garcia, 2010] M. Thimm and A. J. Garcia. Classification and Strategical Issues of Argumentation Games on Structured Argumentation Frameworks. In *Proceedings of AAMAS'10*. IFAAMAS, 2010.

[Vreeswijk and Prakken, 2000] G. Vreeswijk and H. Prakken. Credulous and Sceptical Argument Games for Preferred Semantics. In *Proc. of the 7th European Workshop on Logic for Artificial Intelligence (JELIA'00)*, pages 239–253. Springer, 2000.