



Inteligencia Artificial. Revista Iberoamericana
de Inteligencia Artificial

ISSN: 1137-3601

revista@aepia.org

Asociación Española para la Inteligencia
Artificial
España

Bulling, Nils; Dix, Jürgen

Modelling and Verifying Coalitions using Argumentation and ATL

Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial, vol. 14, núm. 46, 2010, pp. 45-
73

Asociación Española para la Inteligencia Artificial
Valencia, España

Available in: <http://www.redalyc.org/articulo.oa?id=92513108004>

- How to cite
- Complete issue
- More information about this article
- Journal's homepage in redalyc.org

redalyc.org

Scientific Information System

Network of Scientific Journals from Latin America, the Caribbean, Spain and Portugal

Non-profit academic project, developed under the open access initiative



Modelling and Verifying Coalitions using Argumentation and ATL

Nils Bulling and Jürgen Dix

Department of Informatics, Clausthal University of Technology, Germany
{bulling,dix}@in.tu-clausthal.de

Abstract During the last decade argumentation has evolved as a successful approach to formalize commonsense reasoning and decision making in multiagent systems. In particular, recent research has shown that argumentation can be used to provide a suitable framework for reasoning about *coalition formation*: which coalitions can be formed using different argumentation semantics. At the same time *Alternating-time Temporal Logic* (ATL for short) has been successfully used to reason about the behavior and abilities of coalitions of agents. However, an important limitation of ATL operators is that they account only for the *existence of successful strategies* of coalitions, not considering whether coalitions can be actually formed.

This paper is an attempt to combine both frameworks in order to develop a logical system through which we can reason at the same time (1) about abilities of coalitions of agents and (2) about the formation of coalitions. In order to achieve this, we provide a formal extension of ATL, called *Coalitional ATL* (COALATL for short), in which the actual computation of the coalition is modelled in terms of argumentation semantics. Moreover, we integrate goals as agents' incentive to join coalitions and examine the model checking complexity. Particularly, we show that model checking COALATL is Δ_2^P -complete in the most natural cases.

Keywords: multi-agent systems, argumentation, coalition formation, game theory, strategic logic, temporal logic

1 Introduction and motivations

During the last decade, argumentation frameworks [35, 17] have evolved as a successful approach to formalize commonsense reasoning and decision making in multiagent systems (MAS). Application areas include issues such as joint deliberation, persuasion, negotiation, knowledge distribution and conflict resolution (e.g. [40, 36, 37, 11, 31]), among many others. Particularly, recent research by Leila Amgoud [3, 4] has shown that argumentation provides a sound setting to model *reasoning about coalition formation* in MAS. The approach is based on using conflict and preference relationships among coalitions to determine which coalitions should be adopted by the agents. This is done according to a particular argumentation semantics.

Alternating-time Temporal Logic (ATL) [2] is a temporal logic which can be used for reasoning about the behavior and abilities of agents under various rationality assumptions [26, 27, 15]. In ATL the key construct has the form $\langle\langle A \rangle\rangle\phi$, which expresses that a coalition A of agents can *enforce* the formula ϕ . Under a model theoretic viewpoint, $\langle\langle A \rangle\rangle\phi$ holds whenever the agents in A have a winning strategy for ensuring that ϕ holds (independently of the behavior of A 's opponents). However, this operator accounts only for the *theoretical existence* of such a strategy, not taking into account whether the coalition A can be actually formed. Indeed, in order to join a coalition, agents usually require some kind of *incentive* (e.g. sharing common goals, getting rewards, etc.), since usually forming a coalition does not come for

free (fees have to be paid, communication costs may occur, etc.). Consequently, several possible coalition structures among agents may arise, from which the best ones should be adopted according to some rationally justifiable procedure.

In this paper we present an argumentative approach to extend ATL for modelling coalitions. We provide a formal extension of ATL, CoALATL , by including a new construct $\langle\!\langle A \rangle\!\rangle\phi$ which denotes that *the group A of agents is able to build a coalition B, $A \cap B \neq \emptyset$, such that B can enforce ϕ* . That is, it is assumed that agents in A work together and try to form a coalition B. The actual computation of the coalition is modelled in terms of a given argumentation semantics [20] in the context of coalition formation [3]. In a second step, we enrich CoALATL with goals. We address the question *why* agents should cooperate. Goals refer to agents' subjective incentive to join coalitions. We show that the model checking problem of our framework is an extension of the model checking procedure used in ATL and we prove that it is Δ_2^P -complete for a natural class of argumentation semantics.

The rest of the paper is structured as follows. Section 2 summarizes the main concepts of alternating-time temporal logic (ATL). In Section 3 we introduce the notion of *coalitional framework* [3] as well as some fundamental concepts from argumentation theory. Section 4 provides an argumentation-based view of coalition formation by merging ATL and the coalitional framework introduced in Sections 2 and 3. In Section 5 we incorporate goals to CoALATL , turning to model checking in Section 6. Finally, in Sections 7 and 8 we discuss related and future work and conclude.

About this paper. This article is based on [12], [13], and on an invited paper, combining both papers just mentioned, presented in [14]. Sections 2-4 and 6-8 are slight variations of [12]. However, some minor conceptual issues have been modified (in accordance with [13]) or added; discussions on future and related work has also been updated. Section 5 presents the work given in [13]. In this article, we substantially elaborated the work on the computational complexity of the model checking problem presented in Section 6. We included full proofs, especially about the Δ_2^P -hardness, and added some additional complexity results.

2 ATL

Alternating-time Temporal Logic (ATL) [2] enables reasoning about temporal properties and strategic abilities of agents. The language of ATL is defined as follows.

Definition 1 (\mathcal{L}_{ATL} [2]) *Let $\text{Agt} = \{a_1, \dots, a_k\}$ be a nonempty finite set of all agents, and Π be a set of propositions (with typical element p). We denote by “ a ” a typical agent, and by “ A ” a typical group of agents from Agt . The language $\mathcal{L}_{ATL}(\text{Agt}, \Pi)$ is defined by the following grammar: $\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\!\langle A \rangle\!\rangle\gamma$ where $\gamma ::= \varphi \mid \neg\gamma \mid \gamma \wedge \gamma \mid \gamma \mathcal{U} \gamma \mid \bigcirc\gamma \mid \Box\gamma$, $A \subseteq \text{Agt}$, and $p \in \Pi$.*

Formulae φ (resp. γ) are called state (resp. path) formulae.

Informally, $\langle\!\langle A \rangle\!\rangle\gamma$ expresses that agents A have a *collective strategy to enforce γ* . ATL formulae include the usual temporal operators: \bigcirc (“in the next state”), \Box (“always from now on”) and \mathcal{U} (strict “until”). Additionally, \Diamond (“now or sometime in the future”) can be defined as $\Diamond\varphi \equiv \top \mathcal{U} \varphi$.

The semantics of ATL is defined by *concurrent game structures*.

Definition 2 (CGS [2]) *A concurrent game structure (CGS) is a tuple $\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o \rangle$, consisting of: a set $\text{Agt} = \{a_1, \dots, a_k\}$ of agents; set Q of states; set Π of atomic propositions; valuation of propositions $\pi : Q \rightarrow \mathcal{P}(\Pi)$; set Act of actions. Function $d : \text{Agt} \times Q \rightarrow \mathcal{P}(\text{Act})$ indicates the actions available to agent $a \in \text{Agt}$ in state $q \in Q$. We often write $d_a(q)$ instead of $d(a, q)$, and use $d(q)$ to denote the set $d_{a_1}(q) \times \dots \times d_{a_k}(q)$ of action profiles in state q . Finally, o is a transition function which maps each state $q \in Q$ and action profile $\vec{\alpha} = \langle \alpha_1, \dots, \alpha_k \rangle \in d(q)$ to another state $q' = o(q, \vec{\alpha})$.*

A path $\lambda = q_0 q_1 \dots \in Q^\omega$ is an infinite sequence of states such that there is a transition between each q_i, q_{i+1} . We define $\lambda[i] = q_i$ to denote the i -th state of λ . The set of all paths starting in q is defined by $\Lambda_{\mathfrak{M}}(q)$.

A (memoryless) *strategy* of agent a is a function $s_a : Q \rightarrow \text{Act}$ such that $s_a(q) \in d_a(q)$. We denote the set of such functions by Σ_a . A *collective strategy* s_A for team $A \subseteq \text{Agt}$ specifies an individual strategy for each agent $a \in A$; the set of A 's collective strategies is given by $\Sigma_A = \prod_{a \in A} \Sigma_a$ and $\Sigma := \Sigma_{\text{Agt}}$.

The *outcome* of strategy s_A in state q is defined as the set of all paths that may result from executing s_A : $out(q, s_A) = \{\lambda \in \Lambda_{\mathfrak{M}}(q) \mid \forall i \in \mathbb{N}_0 \exists \vec{\alpha} = \langle \alpha_1, \dots, \alpha_k \rangle \in d(\lambda[i]) \forall a \in A (\alpha_a = s_A^a(\lambda[i]) \wedge o(\lambda[i], \vec{\alpha}) = \lambda[i+1])\}$, where s_A^a denotes agent a 's part of the collective strategy s_A .

The semantics of ATL is as follows.

Definition 3 (ATL Semantics [2]) Let a CGS $\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, Act, d, o \rangle$ and $q \in Q$ be given. The semantics of state formulae is given by a satisfaction relation \models as follows:

$$\mathfrak{M}, q \models p \text{ iff } p \in \pi(q)$$

$$\mathfrak{M}, q \models \neg \varphi \text{ iff } \mathfrak{M}, q \not\models \varphi$$

$$\mathfrak{M}, q \models \varphi \wedge \psi \text{ iff } \mathfrak{M}, q \models \varphi \text{ and } \mathfrak{M}, q \models \psi$$

$$\mathfrak{M}, q \models \langle\langle A \rangle\rangle \gamma \text{ iff there is } s_A \in \Sigma_A \text{ such that } \mathfrak{M}, \lambda \models \gamma \text{ for all } \lambda \in out(q, s_A)$$

$$\mathfrak{M}, \lambda \models \varphi \text{ iff } \mathfrak{M}, \lambda[0] \models \varphi;$$

and for path formulae by

$$\mathfrak{M}, \lambda \models \bigcirc \varphi \text{ iff } \mathfrak{M}, \lambda[1] \models \varphi$$

$$\mathfrak{M}, \lambda \models \Box \varphi \text{ iff } \mathfrak{M}, \lambda[i] \models \varphi \text{ for all } i \in \mathbb{N}_0$$

$$\mathfrak{M}, \lambda \models \varphi \mathcal{U} \psi \text{ iff there is an } i \in \mathbb{N}_0 \text{ with } \mathfrak{M}, \lambda[i] \models \psi, \text{ and } \mathfrak{M}, \lambda[j] \models \varphi \text{ for all } 0 \leq j < i.$$

3 Coalitions and Argumentation

In this section we provide an argument-based characterization of coalition formation that will be used later to extend ATL. We follow an approach similar to [3], where an argumentation framework for generating coalition structures is defined. Our approach is a generalization of the framework of Dung for argumentation [20], extended with a *preference relation*. The basic notion is that of a *coalitional framework*, which contains a set of elements \mathfrak{C} (usually seen as agents or coalitions), an attack relation (for modelling conflicts among elements of \mathfrak{C}), and a preference relation between elements of \mathfrak{C} (to describe favorite agents/coalitions).

Definition 4 (Coalitional framework [3]) A coalitional framework is a triple $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ where \mathfrak{C} is a non-empty set of elements, $\mathcal{A} \subseteq \mathfrak{C} \times \mathfrak{C}$ is an attack relation, and \prec is a preorder on \mathfrak{C} representing preferences on elements in \mathfrak{C} .

Let S be a non-empty set of elements. $\mathbb{CF}(S)$ denotes the set of all coalitional frameworks where elements are taken from the set S , i.e. for each $(\mathfrak{C}, \mathcal{A}, \prec) \in \mathbb{CF}(S)$ we have that $\mathfrak{C} \subseteq S$.

The set \mathfrak{C} in Definition 4 is intentionally generic, accounting for various possible alternatives. One alternative is to consider \mathfrak{C} as a set of agents $\text{Agt} = \{a_1, \dots, a_k\}$: $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec) \in \mathbb{CF}(\text{Agt})$. Then, a *coalition* is given by $C = \{a_{i_1}, \dots, a_{i_l}\} \subseteq \mathfrak{C}$ and “agent” can be used as an intuitive reference to elements of \mathfrak{C} . Another alternative is to use a coalitional framework $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ based on $\mathbb{CF}(\mathcal{P}(\text{Agt}))$. Now elements of $\mathfrak{C} \subseteq \mathcal{P}(\text{Agt})$ are *groups* or *coalitions* (where we consider singletons as groups too) of agents. Under this interpretation a coalition $C \subseteq \mathfrak{C}$ is a *set of sets* of agents. Although “coalition” is already used for $C \subseteq \mathfrak{C}$, we also use the intuitive reading “coalition” or “group” to address elements in \mathfrak{C} .¹ Yet another way is not to use the specific structure for elements in \mathfrak{C} , assuming it just consists of abstract elements, e.g. c_1, c_2 , etc. One may think of these elements as individual agents or coalitions. This approach is followed in [3].

In the rest of this paper we mainly follow the first alternative when informally speaking about coalitional frameworks, i.e. we consider \mathfrak{C} as a set of agents.

¹The first interpretation is a special case of the second (coalitional frameworks are members $\mathbb{CF}(\mathcal{P}(\text{Agt}))$).

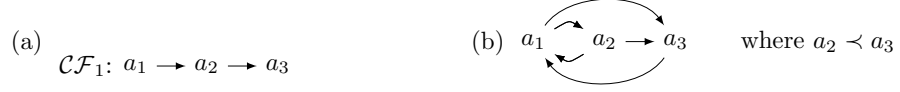


Figure 1: Figure (a) (resp. (b)) corresponds to the coalitional frameworks defined in Example 1 (resp. 3 (b)). Nodes represent agents and arrows between nodes stand for the attack relation.

Example 1 Consider the following two coalitional frameworks: (i) $\mathcal{CF}_1 = (\mathfrak{C}, \mathcal{A}, \prec)$ where $\mathfrak{C} = \{a_1, a_2, a_3\}$, $\mathcal{A} = \{(a_3, a_2), (a_2, a_1), (a_1, a_3)\}$ and agent a_3 is preferred over a_1 , i.e. $a_1 \prec a_3$; and (ii) $\mathcal{CF}_2 = (\mathfrak{C}', \mathcal{A}', \prec')$ where $\mathfrak{C}' = \{\{a_1\}, \{a_2\}, \{a_3\}\}$, $\mathcal{A}' = \{(\{a_3\}, \{a_2\}), (\{a_2\}, \{a_1\}), (\{a_1\}, \{a_3\})\}$ and group $\{a_3\}$ is preferred over $\{a_1\}$, i.e. $\{a_1\} \prec' \{a_3\}$. They capture the same scenario and are isomorphic but $\mathcal{CF}_1 \in \mathbb{CF}(\{a_1, a_2, a_3\})$ and $\mathcal{CF}_2 \in \mathbb{CF}(\mathcal{P}(\{a_1, a_2, a_3\}))$; that is, the first framework is defined regarding single agents and the latter over (trivial) coalitions. Figure 1 (a) shows a graphical representation of the first coalitional framework.

Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework. For $C, C' \in \mathfrak{C}$, we say that C attacks C' iff CAC' . The attack relation represents conflicts between elements of \mathfrak{C} ; for instance, two agents may rely on the same (unique) resource or they may have disagreeing goals, which prevent them from cooperation. However, the notion of attack may not be sufficient for modelling conflicts, as some elements (resp. coalitions) in \mathfrak{C} may be preferred over others. This leads to the notion of *defeater* which combines the notions of attack and preference.

Definition 5 (Defeater) Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework and let $C, C' \in \mathfrak{C}$. We say that C defeats C' if, and only if, C attacks C' and C' is not preferred over C (i.e., not $C \prec C'$). We also say that C is a defeater for C' .

Attacks and defeats are defined between *single* elements of \mathfrak{C} . As we are interested in the formation of coalitions it is reasonable to consider conflicts between coalitions. Members in a coalition may prevent attacks to members in the same coalition; they protect each other. The concept of defence, introduced next, captures this idea of mutual protection.

Definition 6 (Defence) Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework and $C \in \mathfrak{C}$. We say that C defends itself, if it is preferred over all attackers C' of it.

Furthermore, C is defended by a set $\mathfrak{S} \subseteq \mathfrak{C}$ of elements of \mathfrak{C} if, and only if, for all C' defeating C there is a coalition $C'' \in \mathfrak{S}$ defeating C' .

A minimal requirement one should impose on a coalition is that its members do not defeat each other; otherwise, the coalition may be unstable and break up sooner or later because of conflicts among its members. This is formalized in the next definition.

Definition 7 (Conflict-free) Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework and $\mathfrak{S} \subseteq \mathfrak{C}$ a set of elements in \mathfrak{C} . Then, \mathfrak{S} is called conflict-free if, and only if, there is no $C \in \mathfrak{S}$ defeating some member of \mathfrak{S} .

It must be remarked that our notions of “defence” and “conflict-free” are defined in terms of “defeat” rather than “attack”.² Given a coalitional framework \mathcal{CF} we will use argumentation to compute coalitions with desirable properties. In argumentation theory many different semantics have been proposed to define ultimately accepted arguments [20, 16]. We apply this rich framework to provide different ways to coalition formation. A semantics can be defined as follows.

Definition 8 (Coalitional framework semantics) A semantics for a coalitional framework $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ is a (isomorphism invariant) mapping \mathbf{sem} which assigns to a given coalitional framework $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ a set of subsets of \mathfrak{C} , i.e., $\mathbf{sem}(\mathcal{CF}) \subseteq \mathcal{P}(\mathfrak{C})$.

²In [3, 4] these notions are defined the other way around, resulting in a different characterization of stable semantics.

Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework. To formally characterize different semantics we define a function $\mathcal{F}_{\mathcal{CF}} : \mathcal{P}(\mathfrak{C}) \rightarrow \mathcal{P}(\mathfrak{C})$ which assigns to a set of coalitions $\mathfrak{S} \in \mathcal{P}(\mathfrak{C})$ the coalitions defended by \mathfrak{S} .

Definition 9 (Characteristic function \mathcal{F}) Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework and $\mathfrak{S} \subseteq \mathfrak{C}$. The function \mathcal{F} defined by

$$\begin{aligned} \mathcal{F}_{\mathcal{CF}} : \mathcal{P}(\mathfrak{C}) &\rightarrow \mathcal{P}(\mathfrak{C}) \\ \mathcal{F}_{\mathcal{CF}}(\mathfrak{S}) &= \{C \in \mathfrak{C} \mid C \text{ is defended by } \mathfrak{S}\} \end{aligned}$$

is called characteristic function.³

\mathcal{F} can be applied recursively to coalitions resulting in new coalitions. For example, $\mathcal{F}(\emptyset)$ provides all undefeated coalitions and $\mathcal{F}^2(\emptyset)$ constitutes the set of all elements of \mathfrak{C} which members are undefeated or are defended by undefeated coalitions.

Example 2 Consider again the coalitional framework \mathcal{CF}_1 given in Example 1. The characteristic function applied on the empty set results in $\{a_3\}$ since the agent is undefeated, $\mathcal{F}(\emptyset) = \{a_3\}$. Applying \mathcal{F} on $\mathcal{F}(\emptyset)$ determines the set $\{a_1, a_3\}$ because a_1 is defended by a_3 . It is easy to see that $\{a_1, a_3\}$ is a fixed point of \mathcal{F} .

We now introduce the first concrete semantics called coalition structure semantics, which was originally defined in [3].

Definition 10 (Coalition structure \mathbf{sem}_{cs} [3]) Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework. Then

$$\mathbf{sem}_{cs}(\mathcal{CF}) := \left\{ \bigcup_{i=1}^{\infty} \mathcal{F}_{\mathcal{CF}}^i(\emptyset) \right\}$$

is called coalition structure semantics or just coalition structure for \mathcal{CF} .

For a coalitional framework $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ with a finite set \mathfrak{C}^4 the characteristic function \mathcal{F} is continuous [20, Lemma 28]. Since \mathcal{F} is also monotonic it has a least fixed point given by $\mathcal{F}(\emptyset) \uparrow^\omega$ (according to Knaster-Tarski). We have the following straightforward properties of coalition structure semantics.

Proposition 1 (Coalition structure) Let $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework with a finite set \mathfrak{C} . There is always a unique coalition structure for \mathcal{CF} . Furthermore, if no element of $C \in \mathfrak{C}$ defends itself then the coalitional structure is empty, i.e. $\mathbf{sem}_{cs}(\mathcal{CF}) = \{\emptyset\}$.

Example 3 The following situations illustrate the notion of coalitional structure:

- (a) Consider Example 2. Since $\{a_1, a_3\}$ is a fixed point of $\mathcal{F}_{\mathcal{CF}_1}$ the coalitional framework \mathcal{CF}_1 has $\{a_1, a_3\}$ as coalitional structure.
- (b) $\mathcal{CF}_3 := (\mathfrak{C}, \mathcal{A}, \prec) \in \mathbb{CF}(\{a_1, a_2, a_3\})$ (shown in Figure 1(b)), is a coalitional framework with $\mathfrak{C} = \{a_1, a_2, a_3\}$, $\mathcal{A} = \{(a_1, a_2), (a_1, a_3), (a_2, a_1), (a_2, a_3), (a_3, a_1)\}$ and a_3 is preferred over a_2 , $a_2 \prec a_3$, has the empty coalition as associated coalition str., i.e. $\mathbf{sem}_{cs}(\mathcal{CF}) = \{\emptyset\}$.

Since the coalition structure is often very restrictive, it seems reasonable to introduce other less restrictive semantics. Each of the following semantics are well-known in argumentation theory [20] and can be used as a criterion for coalition formation (cf. [3]).

Definition 11 (Argumentation Semantics) Let $(\mathfrak{C}, \mathcal{A}, \prec)$ be a coalitional framework, $\mathfrak{S} \subseteq \mathfrak{C}$ a set of elements of \mathfrak{C} . \mathfrak{S} is called

³We omit the subscript \mathcal{CF} if it is clear from context.

⁴Actually, it is enough to assume that \mathcal{CF} is finitary (cf. [20, Def. 27]).

- (a) admissible extension iff \mathfrak{S} is conflict-free and \mathfrak{S} defends all its elements, i.e. $\mathfrak{S} \subseteq \mathcal{F}(\mathfrak{S})$.
- (b) complete extension iff \mathfrak{S} is conflict-free and $\mathfrak{S} = \mathcal{F}(\mathfrak{S})$.
- (c) grounded extension iff \mathfrak{S} is the smallest (wrt. to set inclusion) complete extension.
- (d) preferred extension iff \mathfrak{S} is a maximal (wrt. to set inclusion) admissible extension.
- (e) stable extension iff \mathfrak{S} is conflict-free and it defeats all arguments not in \mathfrak{S} .

Let \mathbf{sem}_{cs} (resp. $\mathbf{sem}_{complete}$, $\mathbf{sem}_{grounded}$, $\mathbf{sem}_{preferred}$ and \mathbf{sem}_{stable}) denote the semantics which assigns to a coalitional structure \mathcal{CF} all its admissible (resp. complete, grounded, preferred, and stable) extensions.

There is only one unique coalition structure (possibly the empty one) for a given coalitional framework, but there can be several stable and preferred extensions. The existence of at least one preferred extension is assured which is not the case for the stable semantics. Thus, the possible coalitions very much depend on the used semantics.

Example 4 For \mathcal{CF}_3 from Example 3 the following holds:

$$\begin{aligned} \mathbf{sem}_{cs}(\mathcal{CF}) &= \{\emptyset\} \\ \mathbf{sem}_{admissible}(\mathcal{CF}) &= \{\{a_1\}, \{a_2\}, \{a_3\}, \{a_2, a_3\}\} \\ \mathbf{sem}_{complete}(\mathcal{CF}) &= \mathbf{sem}_{grounded}(\mathcal{CF}) = \{\{a_1\}, \{a_2, a_3\}\} = \\ \mathbf{sem}_{preferred}(\mathcal{CF}) &= \mathbf{sem}_{stable}(\mathcal{CF}) = \{\{a_1\}, \{a_2, a_3\}\} \end{aligned}$$

Analogously, for the coalitional framework \mathcal{CF}_1 from Example 1 there exists one complete extension $\{a_1, a_3\}$ which is also a grounded, preferred, and stable extension.

4 Coalitional ATL

In this section we combine *argumentation for coalition formation* and ATL and introduce *Coalitional ATL* (CoALATL). This logic extends ATL by new operators $\langle\!\langle A \rangle\!\rangle$ for each subset $A \subseteq \text{Agt}$ of agents. These new modalities combine, or rather integrate, coalition formation into the original ATL cooperation modalities $\langle\!\langle A \rangle\!\rangle$. The intended reading of $\langle\!\langle A \rangle\!\rangle\varphi$ is that the group A of agents is able to form a coalition $B \subseteq \text{Agt}$ such that some agents of A are also members of B , $A \cap B \neq \emptyset$, and B can enforce φ . Coalition formation is modelled by the formal argumentative approach in the context of coalition formation, as described in Section 3, based on the framework developed in [3].

Our main motivation for this logic is to make it possible to reason about the ability of building coalition structures, and not only about an *a priori* specified group of agents (as it is the case for $\langle\!\langle A \rangle\!\rangle\varphi$). The new modality $\langle\!\langle A \rangle\!\rangle$ provides a rather subjective view of the agents in A and their power to create a group B , $A \cap B \neq \emptyset$, which in turn is used to reason about the ability to enforce a given property.

The language of CoALATL is as follows.

Definition 12 ($\mathcal{L}_{\text{ATL}^c}$) Let $\text{Agt} = \{a_1, \dots, a_k\}$ be a finite, nonempty set of agents, and Π be a set of propositions (with typical element p). We use the symbol “ a ” to denote a typical agent, and “ A ” to denote a typical group of agents from Agt . The language $\mathcal{L}_{\text{ATL}^c}(\text{Agt}, \Pi)$ is defined by the following grammar:

$$\begin{aligned} \varphi ::= & p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\!\langle A \rangle\!\rangle\bigcirc\varphi \mid \langle\!\langle A \rangle\!\rangle\Box\varphi \mid \langle\!\langle A \rangle\!\rangle\varphi\mathcal{U}\varphi \mid \\ & \langle\!\langle A \rangle\!\rangle\bigcirc\varphi \mid \langle\!\langle A \rangle\!\rangle\Box\varphi \mid \langle\!\langle A \rangle\!\rangle\varphi\mathcal{U}\varphi \end{aligned}$$

We extend concurrent game structures by *coalitional frameworks* and an *argumentative semantics*. A coalitional framework is assigned to each state of the model capturing the current “conflicts” among agents. In doing so, we allow that conflicts change over time, being thus *state dependent*. Moreover, we assume that coalitional frameworks depend on the agents. Two initial groups of agents may have different skills to form coalitions. Consider for instance the following example.

Example 5 *Imagine the two agents a_1 and a_2 are not able (because they do not have the money) to convince a_3 and a_4 to join. But a_1 , a_2 and a_3 together have the money and all four can enforce a property φ . So $\{a_1, a_2\}$ are not able to build a greater coalition to enforce φ ; but $\{a_1, a_2, a_3\}$ are. So we are not looking at coalitions per se, but how they evolve from others.*

We assume that the argumentative semantics is the same for all states.

Definition 13 (CGM) *A coalitional game model (CGM) is given by a tuple*

$$\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem} \rangle$$

where $\langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o \rangle$ is a CGS, $\zeta : \mathcal{P}(\text{Agt}) \rightarrow (Q \rightarrow \mathbb{CF}(\text{Agt}))$ is a function which assigns a coalitional framework over Agt to each state of the model subjective to a given group of agents, and sem is an (argumentative) semantics as defined in Definition 8. The set of all such models is given by $\mathbb{M}(Q, \text{Agt}, \Pi, \text{sem}, \zeta)$.

A model provides an argumentation semantics sem which assigns all formable coalitions to a given coalitional framework. As argued before we require from a valid coalition that it is not only justified by the argumentation semantics but that it is also not disjunct with the predetermined starting coalition. This leads to the notion *valid coalition*.

Definition 14 (Valid coalition) *Let $A, B \subseteq \text{Agt}$ be groups of agents, $\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem} \rangle$ be a CGM and $q \in Q$.*

We say that B is a valid coalition with respect to A, q , and \mathfrak{M} whenever $B \in \text{sem}(\zeta(A)(q))$ and if $A \neq \emptyset$ then $A \cap B \neq \emptyset$. Furthermore, we use $\text{vc}_{\mathfrak{M}}(A, q)$ to denote the set of all valid coalitions regarding A, q , and \mathfrak{M} . The subscript \mathfrak{M} is omitted if clear from the context.

Remark 2 *Note, that in [12] we assume that the members of the initial group A work together, whatever the reasons might be. So group A was added to the semantics. This ensured that agents in A can enforce ψ on their own, if they are able to do so. Even if A is not accepted originally by the argumentation semantics, i.e. $A \notin \text{sem}(\zeta(A)(q))$. Here, we drop this requirement. As pointed out in [13] the “old” semantics is just a special case of this new one: The operator from [12] can be defined as $\langle A \rangle \gamma \vee \langle A \rangle \gamma$.*

Moreover, we changed the condition that the predefined group given in the coalitional operator must be a subset of the formed coalition, $A \subseteq B$, to the requirement that some member of the initial coalition (if $A \neq \emptyset$) should be in the new one, $A \cap B \neq \emptyset$. Both definitions make sense in different scenarios; however, the new one seems to be more generic.

The semantics of the new modality is given by

Definition 15 (CoalATL Semantics) *Let a CGM $\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem} \rangle$ a group of agents $A \subseteq \text{Agt}$, and $q \in Q$ be given. The semantics of Coalitional ATL extends that of ATL, given in Definition 3, by the following rule ($\langle A \rangle \psi \in \mathcal{L}_{\text{ATL}^c}(\text{Agt}, \Pi)$):*

$\mathfrak{M}, q \models \langle A \rangle \psi$ *iff there is a coalition $B \in \text{vc}(A, q)$ such that $\mathfrak{M}, q \models \langle B \rangle \psi$.*

Remark 3 (Different Semantics, \models_{sem}) *We have just defined a whole class of semantic rules for modality $\langle \cdot \rangle$. The actual instantiation of the semantics sem , for example $\text{sem}_{\text{stable}}$, sem_{pref} , and sem_{cs} defined in Section 3, affects the semantics of the cooperation modality.*

For the sake of readability, we sometimes annotate the satisfaction relation \models with the presently used argumentation semantics. That is, given a CGM \mathfrak{M} with an argumentation semantics sem we write \models_{sem} instead of \models .

The underlying idea of the semantic definition of $\langle A \rangle \psi$ is as follows. A given (initial) group of agents $A \subseteq \text{Agt}$ is able to form a *valid coalition* B (where A and B must not be disjoint), with respect to a given coalitional framework \mathcal{CF} and a particular semantics sem , such that B can enforce ψ .

Similarly to the alternatives to our definition of valid coalitions there are other sensible semantics for CoalATL. The semantics we presented here is not particularly dependent on time; i.e., except from the selection of a valid coalition B at the initial state there is no further interaction between time and coalition formation. We have chosen this simplistic definition to present our main idea – the connection of ATL and coalition formation by means of argumentation – as clear as possible.

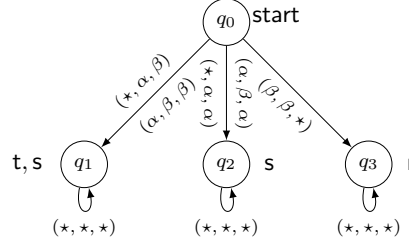


Figure 2: A simple CGS defined in Example 6.

Proposition 4 ([12]) *Let $A \subseteq \text{Agt}$ and $\langle A \rangle \psi \in \mathcal{L}_{ATL^c}(\text{Agt}, \Pi)$. Then $\langle A \rangle \psi \rightarrow \bigvee_{B \in \mathcal{P}(\text{Agt}), A \subseteq B} \langle B \rangle \psi$ is valid with respect to CGM's.*

Compared to ATL, a formula like $\langle A \rangle \varphi$ does *not* refer to the ability of A to enforce φ , but rather to the ability of A to *constitute* a coalition B , such that $A \cap B \neq \emptyset$, and then, in a second step, to the ability of B to enforce φ . Thus, two different notions of ability are captured in these new modalities. For instance, $\langle A \rangle \psi \wedge \neg \langle \emptyset \rangle \psi$ expresses that group A of agents can enforce ψ , but there is no *reasonable* coalition which can enforce ψ (particularly not A , although they possess the theoretical power to do so).

Example 6 *There are three agents a_1, a_2 , and a_3 which prefer different outcomes. Agent a_1 (resp. a_2, a_3) desires to get outcome r (resp. s, t). One may assume that all outcomes are distinct; for instance, a_1 is not satisfied with an outcome x whenever $x \neq r$. Each agent can choose to perform action α or β . Action profiles and their outcomes are shown in Figure 2. The $*$ is used as a placeholder for any of the two actions, i.e. $*$ $\in \{\alpha, \beta\}$. For instance, the profile $(\beta, \beta, *)$ leads to state q_3 whenever agent a_1 and a_2 perform action β and a_3 either does α or β .*

According to the scenario depicted in the figure, a_1 and a_2 cannot commonly achieve their goals. The same holds for a_1 and a_3 . On the other hand, there exists a situation, q_1 , in which both agents a_2 and a_3 are satisfied. One can formalize the situation as the coalitional game $\mathcal{CF} = (\mathfrak{C}, \mathcal{A}, \prec)$ given in Example 3(b), that is, $\mathfrak{C} = \text{Agt}$, $\mathcal{A} = \{(a_1, a_2), (a_1, a_3), (a_2, a_1), (a_2, a_3), (a_3, a_1)\}$ and $a_2 \prec a_3$.

We formalize the example as the CGM $\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem} \rangle$ where $\text{Agt} = \{a_1, a_2, a_3\}$, $Q = \{q_0, q_1, q_2, q_3\}$, $\Pi = \{r, s, t\}$, and $\zeta(A)(q) = \mathcal{CF}$ for all states $q \in Q$ and groups $A \subseteq \text{Agt}$. Transitions and the state labeling can be seen in Figure 2. Furthermore, we do not specify a concrete semantics sem yet, and rather adjust it in the remainder of the example.

We can use pure ATL formulas, i.e. formulas not containing the new modalities $\langle \cdot \rangle$, to express what groups of agents can achieve. We have, for instance, that agents a_1 and a_2 can enforce a situation which is undesirable for a_3 : $\mathfrak{M}, q_0 \models \langle \langle a_1, a_2 \rangle \rangle \bigcirc r$. Indeed, $\{a_1, a_2\}$ and the grand coalition Agt (since it contains $\{a_1, a_2\}$) are the only coalitions which are able to enforce $\bigcirc r$; we have

$$\mathfrak{M}, q_0 \models \neg \langle \langle X \rangle \rangle \bigcirc r \quad (1)$$

for all $X \subset \text{Agt}$ and $X \neq \{a_1, a_2\}$. Outcomes s or t can be enforced by a_2 : $\mathfrak{M}, q_0 \models \langle \langle a_2 \rangle \rangle \bigcirc (s \vee t)$. Agents a_2 and a_3 also have the ability to enforce a situation which agrees with both of their desired outcomes: $\mathfrak{M}, q_0 \models \langle \langle a_2, a_3 \rangle \rangle \bigcirc (s \wedge t)$

These properties do not take into account the coalitional framework, that is, whether specific coalitions can be formed or not. By using the coalitional framework, we get

$$\mathfrak{M}, q_0 \models_{\text{sem}} \langle \langle a_1, a_2 \rangle \rangle \bigcirc r \wedge \neg \langle \langle a_1 \rangle \rangle \bigcirc r \wedge \neg \langle \langle a_2 \rangle \rangle \bigcirc r$$

for any semantics sem introduced in Definition 8 and calculated in Example 4. The possible coalition (resp. coalitions) containing a_1 (resp. a_2) is $\{a_1\}$ (resp. are $\{a_2\}$ and $\{a_2, a_3\}$). But neither of these can enforce $\bigcirc r$ (in q_0) because of (1). Thus, although it is the case that the coalition $\{a_1, a_2\}$ has the theoretical ability to enforce r in the next moment (which is a “losing” situation for a_3), a_3 should not consider it as sensible since agents a_1 and a_2 would not agree to constitute a coalition (according to the coalitional framework \mathcal{CF}).

The decision for a specific semantics is a crucial point and depends on the actual application. The next example shows that with respect to a particular argumentation semantics, agents are able to form a coalition which can successfully achieve a given property, whereas another argumentative semantics does not allow that.

Example 7 COALATL can be used to determine whether a coalition for enforcing a specific property exists. Assume that **sem** represents the grounded semantics. For instance, the statement

$$\mathfrak{M}, q_0 \models_{\text{sem}_{\text{grounded}}} \langle \emptyset \rangle \bigcirc t$$

expresses that there is a grounded coalition (i.e. a coalition wrt to the grounded semantics) which can enforce $\bigcirc t$, namely the coalition $\{a_2, a_3\}$. This result does not hold for all semantics; for instance, we have

$$\mathfrak{M}, q_0 \not\models_{\text{sem}_{cs}} \langle \emptyset \rangle \bigcirc t$$

with respect to the coalition structure semantics, since the coalition structure is the empty coalition and $\mathfrak{M}, q_0 \not\models \langle \emptyset \rangle \bigcirc t$.

In the following section we sketch of the language can be extended by an *update mechanism*, in order to compare different argumentative semantics using formulae inside the object language.

4.1 An update mechanism

In Example 7 we have shown that the underlying semantics of the coalition framework is crucial for the truth of a formula. We showed, for instance, that $\langle \emptyset \rangle \bigcirc t$ is true wrt the *grounded semantics* but false regarding the *coalition structure semantics*. This comparison took place on the meta-level; two CGM's were defined, using grounded and coalition structure semantics, respectively. In this section, we introduce semantics as *first-class citizens* in the object language. Therefore, we extend the language by *semantic terms*, out of a set Ω , and an update operator (**set-sem** \cdot). Semantically, a CGM \mathfrak{M} is enriched by a *denotation function* $\llbracket \cdot \rrbracket : \Omega \rightarrow (\text{CF}(\text{Agt}) \rightarrow \mathcal{P}(\mathcal{P}(\text{Agt})))$ which maps semantic terms to an argumentation semantics. The idea is that (**set-sem** sem) resets the semantics in \mathfrak{M} to $\llbracket \text{sem} \rrbracket$, where $\text{sem} \in \Omega$. The intended reading for (**set-sem** sem) φ is that φ holds if the argumentation semantics is given by $\llbracket \text{sem} \rrbracket$. We formally define the new language and its models.

Definition 16 ($\mathcal{L}_{\text{ATL}^c}$ plus update) *As in Definition 12, let $\text{Agt} = \{a_1, \dots, a_k\}$ and Π be given. Let Ω be a non-empty set, its elements are called semantic symbols (with typical element sem).*

*The logic $\mathcal{L}_{\text{ATL}^c+u}(\text{Agt}, \Pi, \Omega)$ is given by all formulas of $\mathcal{L}_{\text{ATL}^c}(\text{Agt}, \Pi)$ and for all $\varphi \in \mathcal{L}_{\text{ATL}^c+u}(\text{Agt}, \Pi, \Omega)$ we also have (**set-sem** sem) $\varphi \in \mathcal{L}_{\text{ATL}^c+u}$, where $\text{sem} \in \Omega$.*

Remark 5 (Standard semantic terms) *We assume that for all semantics defined in Definitions 8 and 10 there is a corresponding semantics term in Ω . For example, for the grounded semantics $\text{sem}_{\text{grounded}}$ there is a term $\text{sem}_{\text{grounded}}$ in Ω .*

We need to define the denotation of the new syntactic objects.

Definition 17 (CGM + update) *A coalitional concurrent game structure with update ($\text{CCGS}+u$) is given by a tuple*

$$\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem}, \Omega, \llbracket \cdot \rrbracket \rangle$$

where $\langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem} \rangle$ is a CGM, Ω is a non-empty set of semantic terms, and $\llbracket \cdot \rrbracket : \Omega \rightarrow (\text{CF}(\text{Agt}) \rightarrow \mathcal{P}(\mathcal{P}(\text{Agt})))$ is a denotation function, such that $\llbracket \text{sem} \rrbracket$ is an argumentation semantics over $\text{CF}(\text{Agt})$ (cf. Definition 8) for all $\text{sem} \in \Omega$.

In accordance with Remark 5 we assume that the denotation of semantic terms belonging to one of the “standard” semantics connects the terms with their semantics. That is, we assume, for instance, that the denotation of $\text{sem}_{\text{grounded}}$ is $\text{sem}_{\text{grounded}}$.

In addition to all semantic rules given before, we also need to interpret (**set-sem** \cdot).

Definition 18 (Semantics) Let $\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem}, \Omega, [\cdot] \rangle$ be a CCGS_{+U} and $\text{sem} \in \Omega$. The semantics of CoALATL plus update extends that of CoALATL , given in Definition 15, by the following rule ($\psi \in \mathcal{L}_{\text{ATL}^{c+u}}(\text{Agt}, \Pi, \Omega)$):

$$\mathfrak{M}, q \models (\text{set-sem sem})\psi \text{ iff } \mathfrak{M}^{[\text{sem}]}, q \models \psi$$

where $\mathfrak{M}^{[\text{sem}]}$ is a CCGS_{+U} equal to \mathfrak{M} but its argumentation semantics is given by (set to) $[\text{sem}]$.

Example 8 Let \mathfrak{M}' be the CCGS_{+U} which corresponds to the CGM \mathfrak{M} given in Example 7 extended by a set Ω of semantic terms and denotation function $[\cdot]$. We can state the relation between grounded and coalition structure semantics directly on the object level:

$$\mathfrak{M}, q_0 \models (\text{set-sem sem}_{\text{grounded}}) \langle \emptyset \rangle \bigcirc \mathbf{t} \wedge \neg (\text{set-sem sem}_{\text{cs}}) \langle \emptyset \rangle \bigcirc \mathbf{t}.$$

5 Cooperation and Goals

Why should agents join coalitions? They must have reasons to do so. Here, we consider *goals* as the driving force, and consequently, we assume that agents act to reach their goals. Firstly, we propose an *abstract goal framework*. Secondly, we use specific languages for goals and objectives, and we propose ATL as a suitable language to capture agents' goals. Finally, we *implement goals* into the semantics of CoALATL , discuss its benefits and illustrate it with an example.

5.1 Goals and Agents

Pro-activeness and *social ability* are among the widely accepted characteristics of intelligent agents [41]. In BDI frameworks, *goals* (or *desires*) and *beliefs* play an important role [10, 38].

We believe that also *the social ability to join coalitions*, should be based on some incentive. Agents are usually not developed to offer their services for free. Also in the agent programming community several types of goals (e.g. *achievement* or *maintenance* goals) are commonly considered as an agent's driving force. Here, we present a simple abstract framework to deal with these notions.

Definition 19 (\mathcal{G} , goal mapping \mathbf{g}) Let \mathcal{G}_a be a non-empty set of elements (set of goals), one for each agent $a \in \text{Agt}$, and $\mathcal{G} := \bigcup_{a \in \text{Agt}} \mathcal{G}_a$. By “ g ” we denote a typical element from \mathcal{G} . A goal mapping is a function $\mathbf{g} : \text{Agt} \rightarrow (Q^+ \rightarrow \mathcal{P}(\mathcal{G}))$ assigning a set of goals to a given sequence of states and agent.

So, a goal mapping assigns a set of goals to a *history*, depending on an agent. This is needed to introduce goals into CGM 's. The history dependency can be used, for instance, to model when a goal should be removed from the list: An agent having a goal $\Diamond s$ may drop it after reaching a state in which s holds. Alternatively, a model update mechanism can be used to achieve the same regarding state-based goal mappings; however, in our opinion the former is more elegant.

An agent might have several goals. Often, goals can not be reached simultaneously which requires means to decide which goal should be selected first. We model this by a preference ordering.

Definition 20 (Goal preference relation) A goal preference ordering (*gp-ordering*) \preceq over a set of goals $\mathcal{G}' \subseteq \mathcal{G}$ is a complete, transitive, antisymmetric, and irreflexive relation $\preceq \subseteq \mathcal{G}' \times \mathcal{G}'$. We say that a goal g is preferred to g' if $g \preceq g'$.

Given a goal mapping \mathbf{g}_a for $a \in \text{Agt}$ we assume that there implicitly also is a *gp-ordering* \preceq_a (a 's *gp-ordering*).

So far, we did not say how goals can be actually used to form coalitions. We assume, given some task, that agents having goals satisfied or partly satisfied by the outcome of the task are willing to cooperate to bring about the task. In the following we will use the notion *objective* (or objective formula) to refer to both the task itself and the outcome of it. A typical objective is written as o . Agents which have goals fulfilled or at least partly supported by objective o are possible candidates to participate in a coalition aiming at o .

We say that an objective o *satisfies* goal g , $o \hookrightarrow g$, if the complete goal g is fulfilled after o has been accomplished. If a goal is (partly) satisfied by o we say that o *supports* g , $o \hookrightarrow^s g$; i.e. there is another goal g' which is a *subgoal* of g and which is satisfied. These notions will be made precise in the following sections. Intuitively, an objective $\Box t$ satisfies goal $\Box (t \vee s)$ and supports goal $\Diamond t$.

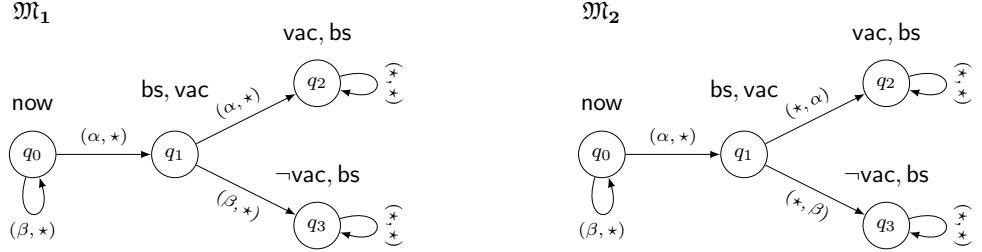


Figure 3: Two simple models showing that ATL goals are useful. $\star \in \{\alpha, \beta\}$ is used as a placeholder for any of the two actions.

5.2 Specifying Goals and Objectives

In this section, we propose to use ATL-path formulas for specifying *goals*. It has been shown that temporal logics like *linear-time temporal logic* (LTL) and *computation tree logic* (CTL) can be used as goal specification languages [5, 8, 7].

Goals formulated in LTL are very intuitive. Formulae like \Diamond rich (eventually being rich), \bigcirc takeUmbrella (take umbrella in the next moment), or $\Box \Diamond$ sleep (going to sleep again and again) have clear interpretations. But goals formulated in CTL can be ambiguous. A goal like $A \Diamond$ rich⁵ does not seem fundamentally different from \Diamond rich from the agent's point of view. Its goal of being rich in the future can be read implicitly as being rich in all possible futures; only one of them can actually become true and in that particular one the agent wants to be rich.

In this section we will use ATL for expressing agents' goals. At first glance, this seems to contradict the statement made above since CTL can be seen as a special case (the one agent fragment) of ATL. But this is not the case: CTL refers to a purely temporal setting whereas ATL talks about *abilities of agents*. Here is a clarifying example. Assume that there are two agents a and b both having access to the same critical section; that is, either a or b should access this section but not both. In such a case it is reasonable that agent a has the goal of preventing b to enter this section on its own: $\neg \langle\langle b \rangle\rangle \Diamond$ critical. However, it might be acceptable for a that b together with another agent c enters the critical section because then c has to unlock resources a could use instead. Let us consider a more detailed example.

Example 9 (ATL-goals) In the example we consider two agents a and b . Both agents can perform actions α and β . The first agent, leader of a research group, would like to get a better salary (bs) and wants to retain the power to decide when to take vacation (vac). So, a 's goal can be expressed as $\gamma \equiv \Box (\neg \text{now} \rightarrow \text{bs} \wedge \text{vac})$. Interpreting the models shown in Figure 3 purely temporally (i.e. without action profiles) the CTL formula $E\gamma$ is satisfied in q_0 in both models: There are q_0 -paths which satisfy γ . On the other hand, $A\gamma$ is false in both models in q_0 .

Now agent b enters the scene. A higher salary would require a to move to a company in which the agent has a boss who might be able to decide on a 's vacation (depending on the contract). Actually, although a would like to have a better salary he prefers to decide on his vacation on his own. Thus, his goal can be reformulated to $\gamma' \equiv \Box (\neg \text{now} \rightarrow \text{bs} \wedge \neg \langle\langle b \rangle\rangle \Diamond \neg \text{vac})$, or equivalently in this example $\Box (\neg \text{now} \rightarrow \text{bs} \wedge \langle\langle a \rangle\rangle \Box \text{vac})$. Now, it is easy to see that $\mathfrak{M}_1, q_0 \models \langle\langle a \rangle\rangle \gamma'$ but $\mathfrak{M}_2, q_0 \not\models \langle\langle a \rangle\rangle \gamma'$. In the first model b does not have the power to decide on a 's vacation but b has this ability in the second model.

This quite simplistic example shows that ATL formulae can make sense as goal specification language.

Definition 21 (ATL-Goal) Let γ, γ' be ATL path formulae. An ATL-goal has the form γ or $\gamma \wedge \gamma'^6$.

Note that goals can easily be defined as Coalitional ATL formulae; however, due to simplicity we stick to pure ATL formulae.

It remains to define the objective language. Consider the CoalATL formula $\langle\langle A \rangle\rangle \gamma$. The question is whether there is a *rational* group to bring about γ ; thus, only agents which gain advantage when γ is

⁵The operator A refers to all possible paths starting in a state. In ATL this operator can be expressed as $\langle\langle \emptyset \rangle\rangle$.

⁶Note that $\gamma \wedge \gamma'$ is not an ATL path formula anymore.

fulfilled should cooperate. Hence, we consider γ as objective; so, in general all Coalitional ATL path formulae.

Definition 22 (CoalATL-objective) *An CoalATL-objective is an \mathcal{L}_{ATL^c} path formula.*

5.3 Goals as a Means for Cooperation

In this section we link together Coalitional ATL with the goal framework described above. The syntax of the logic is given as in Section 4. The necessary change takes place in the semantics. We redefine what it means for a coalition to be *valid*.

Up to now valid coalitions were solely determined by coalitional frameworks. Conflicts represented by such frameworks are a coarse, but necessary, criterion for a successful coalition formation process. However, nothing is said about incentives to *join* coalitions, only why coalitions should *not* be joined.

Goals allow to capture the first issue. For a given objective formula o and a finite sequence of states, called *history*, we do only consider agents which have some goal supported by the current objective. CGM's *with goals* are given as a straightforward extension of CGM's (cf. Definition 13).

Definition 23 (CGM with goals) *A CGM with goals (CGM_G) \mathfrak{M} is given by a model of $\mathbb{M}(Q, \text{Agt}, \Pi, \text{sem}, \zeta)$ extended by a set of goals \mathcal{G} and a goal mapping \mathbf{g} over \mathcal{G} . The set of all such models is denoted $\mathbb{M}^g(Q, \text{Agt}, \Pi, \text{sem}, \zeta, \mathcal{G}, \mathbf{g})$ or just \mathbb{M}^g if we assume standard naming.*

To define the semantics we need some additional notation. Given a path $\lambda \in Q^\omega$ we use $\lambda[i, j]$ to denote the sequence $\lambda[i]\lambda[i+1] \dots \lambda[j]$ for $i, j \in \mathbb{N}_0 \cup \{\infty\}$ and $i < j$. A *history* is a finite sequence $h = q_1 \dots q_n \in Q^+$, $h[i]$ denotes state q_i if $n \geq i$, q_n for $i \geq n$, and ε for $i < 0$ where $i \in \mathbb{Z} \cup \{\infty\}$. Furthermore, given a history h and a path or history λ the combined path/history starting with h extended by λ is denoted by $h \circ \lambda$.

Finally, we present the semantics of CoalATL *with* goals. It is similar to Definition 15. Here, however, it is necessary to keep track of the steps (i.e. visited states) made to determine the goals of the agents.

Definition 24 (Goal-based semantics of $\mathcal{L}_{ATL}^{\text{vcg}}$) *Let \mathfrak{M} be a CGM_G , q a state, φ, ψ state-, γ a path formula, and $i, j \in \mathbb{N}_0$. Semantics to $\mathcal{L}_{ATL}^{\text{vcg}}$ formulae is given as follows:*

$$\mathfrak{M}, q, \tau \models \mathbf{p} \text{ iff } \mathbf{p} \in \pi(q)$$

$$\mathfrak{M}, q, \tau \models \varphi \wedge \psi \text{ iff } \mathfrak{M}, q, \tau \models \varphi \text{ and } \mathfrak{M}, q, \tau \models \psi$$

$$\mathfrak{M}, q, \tau \models \neg \varphi \text{ iff not } \mathfrak{M}, q, \tau \models \varphi$$

$$\mathfrak{M}, q, \tau \models \langle\langle A \rangle\rangle \gamma \text{ iff there is a strategy } s_A \in \Sigma_A \text{ such that for all } \lambda \in \text{out}(q, s_A) \text{ it holds that } \mathfrak{M}, \lambda, \tau \models \varphi$$

$$\mathfrak{M}, q, \tau \models \langle A \rangle \gamma \text{ iff there is } A' \in \text{vcg}(q, A, \gamma, \tau) \text{ such that } \mathfrak{M}, q, \tau \models \langle\langle A \rangle\rangle \gamma$$

$$\mathfrak{M}, \lambda, \tau \models \varphi \text{ iff } \mathfrak{M}, \lambda[0], \tau \models \varphi$$

$$\mathfrak{M}, \lambda, \tau \models \Box \varphi \text{ iff for all } i \text{ it holds that } \mathfrak{M}, \lambda[i], \tau \circ \lambda[1, i] \models \varphi$$

$$\mathfrak{M}, \lambda, \tau \models \bigcirc \varphi \text{ iff it holds that } \mathfrak{M}, \lambda[1], \tau \circ \lambda[1] \models \varphi$$

$$\mathfrak{M}, \lambda, \tau \models \varphi \mathcal{U} \psi \text{ iff there is a } j \text{ such that } \mathfrak{M}, \lambda[j], \tau \circ \lambda[1, j] \models \psi \text{ and for all } 0 \leq i < j \text{ it holds that } \mathfrak{M}, \lambda[i], \tau \circ \lambda[1, i] \models \varphi.$$

Ultimately, we are interested in $\mathfrak{M}, q \models \varphi$ defined as $\mathfrak{M}, q, q \models \varphi$.

We have to define when a goal is satisfied. Although the definition of *support* can be defined similarly, we focus on the former notion only.

Definition 25 (Satisfaction of goals) *Let g be an ATL-goal, o an $\mathcal{L}_{ATL}^{\text{vcg}}$ -objective, and $\tau \in Q^+$. We say that objective o satisfies g , for short $o \hookrightarrow_{\mathfrak{M}, \tau, B} g$, with respect to \mathfrak{M}, τ , and B if, and only if, there is a strategy $s_B \in \Sigma_B$ such that*

1. for all $\lambda \in \text{out}(\tau[\infty], s_B)$ it holds that $\mathfrak{M}, \lambda, \tau \models o$ implies $\mathfrak{M}, \lambda \models g$, and
2. that there is some path $\lambda \in \text{out}(\tau[\infty], s_B)$ with $\mathfrak{M}, \lambda, \tau \models o$.

A goal is satisfied by an objective if each path (enforceable by B) that satisfies the objective does also satisfy the goal. That is, satisfaction of the objective will guarantee that the goal becomes true. The second condition ensures that the coalition actually has a way to bring about the goal. We show later, however, that the second condition is superfluous using the semantics defined in Definition 24.

It remains to define the semantics for combined (by conjunction) ATL path formulae. Therefore, we extend the ordinary semantics of ATL (given in Definition 3) by the following semantic rule: $\mathfrak{M}, \lambda \models \gamma_1 \wedge \dots \wedge \gamma_n$ iff $\mathfrak{M}, \lambda \models \gamma_i$ for $i = 1, \dots, n$.

All the new functionality provided by goals is captured by the new valid coalition function vc^g

Definition 26 (Valid coalitions, $\text{vc}^g(q, A, o, \tau)$) Let $\mathfrak{M} \in \mathbb{M}^g$, $\tau \in Q^+$, $A, B \subseteq \text{Agt}$, o an COALATL objective.

We say that B is a valid coalition after τ with respect to A , o , and \mathfrak{M} if, and only if,

1. $B \in \text{sem}(\zeta(A)(\tau[\infty]))$, $A \cap B \neq \emptyset$ if $A \neq \emptyset$, and
2. there are goals $g_{b_i} \in \mathfrak{g}_{b_i}(\tau)$, one per agent $b_i \in B$, such that $o \hookrightarrow_{\mathfrak{M}, \tau, B} g_{b_1} \wedge \dots \wedge g_{b_{|B|}}$

The set $\text{vc}^g(q, A, o, \tau)$ consists of all such valid coalitions wrt to \mathfrak{M} .

Thus, for the definition of valid coalitions among other things, a goal mapping, a function ζ and a sequence of states τ are required. The intuition of τ is that it represents the history (the sequence of states visited so far including the current state). So, τ is used to determine which goals of the agents are still active.

Coming back to the satisfaction of goals. Let $\langle A \rangle'$ denote a coalitional operator with semantics equal to $\langle A \rangle$ but without the second condition in Definition 25 (note that this definition is needed to determine the valid coalitions).

Proposition 6 Let γ be an COALATL path formula, $\mathfrak{M} \in \mathbb{M}^g$, $q \in Q_{\mathfrak{M}}$, $A \subseteq \text{Agt}$, and $\tau \in Q^+$. Then we have that $\mathfrak{M}, q, \tau \models \langle A \rangle \gamma$ if, and only if, $\mathfrak{M}, q, \tau \models \langle A \rangle' \gamma$

Proof. Firstly, observe that $\forall \lambda \in \text{out}(q, s_B) (\lambda \models \gamma)$ implies $\exists \lambda \in \text{out}(q, s_B) (\lambda \models \gamma)$ since the outcome is never empty. Then we have $\mathfrak{M}, q, \tau \models \langle A \rangle \gamma$ iff $\exists B \in \text{vc}^g \exists s_B \in \Sigma_B \forall \lambda \in \text{out}(q, s_B) (\lambda, \tau \models \gamma)$ iff $\exists B \in \text{vc}^g \exists s_B \in \Sigma_B (\forall \lambda \in \text{out}(q, s_B) (\lambda, \tau \models \gamma) \text{ and } \exists \lambda \in \text{out}(q, s_B) (\lambda, \tau \models \gamma)) (\star)$ iff $\exists B \in \text{vc}^{g'} (q, \tau \models \langle B \rangle \gamma)$ iff $q, \tau \models \langle A \rangle' \gamma$ where $\text{vc}^{g'}$ is equal to vc^g except for condition 2 in Definition 25. Finally, it is easy to observe that the equivalence (\star) holds since we can just take s_B as goal satisfaction strategy. (We omitted the model and the parameters of vc^g ad $\text{vc}^{g'}$.) ■

Proposition 7 We have that $\mathfrak{M}, q, \tau \models \langle A \rangle \gamma$ iff there is a coalition $B \in \text{vc}(A, q)$ and goals $g_b \in \mathfrak{g}_b(\tau)$ one for each $b \in B$ such that $\mathfrak{M}, q, \tau \models \langle B \rangle (\gamma \wedge \bigwedge_{b \in B} g_b)$

Remark 8 (Preferred goals) In the abstract goal framework presented in Section 5.1 we defined a preference ordering over goals. The gp-orderings highly influence the coalition formation process. However, for this paper we decided to focus on the pure goal framework since the interplay between the formation process becomes much more sophisticated if preferences are taken into account. We just give a brief motivation for preferences and why they increase the complexity of coalition building.

The set of valid coalitions consists of all coalitions which are acceptable/conflict-free (according to a coalitional framework) and in which all agents have an incentive to join the coalition (that is, some goal has to be satisfied/supported). Let us consider two valid groups B and B' both containing the agent a . Both groups are somewhat appealing for a since they satisfy some of his goals, say B (resp. B') can bring about g (resp. g'). In our framework B and B' are treated equally good. Is this reasonable? From an abstract level it is; however, a finer grained analysis should incorporate the preferences between goals. If, for instance, g is preferred over g' agent a should rather go for coalition B instead of B' . The agent would prefer to bring about g thus joining B . On the other hand, if a refuses to join B' it might be possible, by

a symmetric argument, that another agent, say b , refuses to take part in B , such that in the end neither B nor B' will form. Of course, in such a situation both agents prefer to bring about their less preferred goals. This is still better than getting nothing.

This reasoning very much reminds on game theoretic rationality concepts. For example, the motivation behind a Nash equilibrium strategy shows a strong connection: No agent has an incentive to unilaterally choose another strategy. Even closer are concepts from cooperative game theory. This discussion shows how interesting the incorporation of a preference ordering over goals is. However, this is also the reason why we did not incorporate it in the formation process here, it would be out of the scope of this paper. Our current research deals with this issue.

5.4 Progression of ATL goals

A goal mapping takes the history into account to be able to reflect if a goal has become fulfilled. For example, if an agent has goal $\Diamond p$ and p became satisfied in a state on the current history the goal should be marked as completed in the following state. (Of course, a new goal in this state can again be $\Diamond p$.) Another, more practical but also restricted option, is to consider an initial goal base \mathcal{GB} and modify, specialize or remove, the formulae according to the steps taken. So, goal $\Diamond p \wedge \Box q$ should be specialized to $\Box q$ if a state is reached in which p holds. In [6] such a progression procedure is presented for first-order LTL.

6 Model Checking ATL^c

In this section we present an algorithm for model checking CoALATL formulae. The model checking problem is given by the question whether a given CoALATL formula follows from a given CGM \mathfrak{M} and state q , i.e. whether $\mathfrak{M}, q \models \varphi$ [18]. In [2] it is shown that model checking ATL is **P**-complete, with respect to the number of transitions of \mathfrak{M} , m , and the length of the formula, l , and can be done in time $\mathcal{O}(m \cdot l)$.

6.1 Easiness

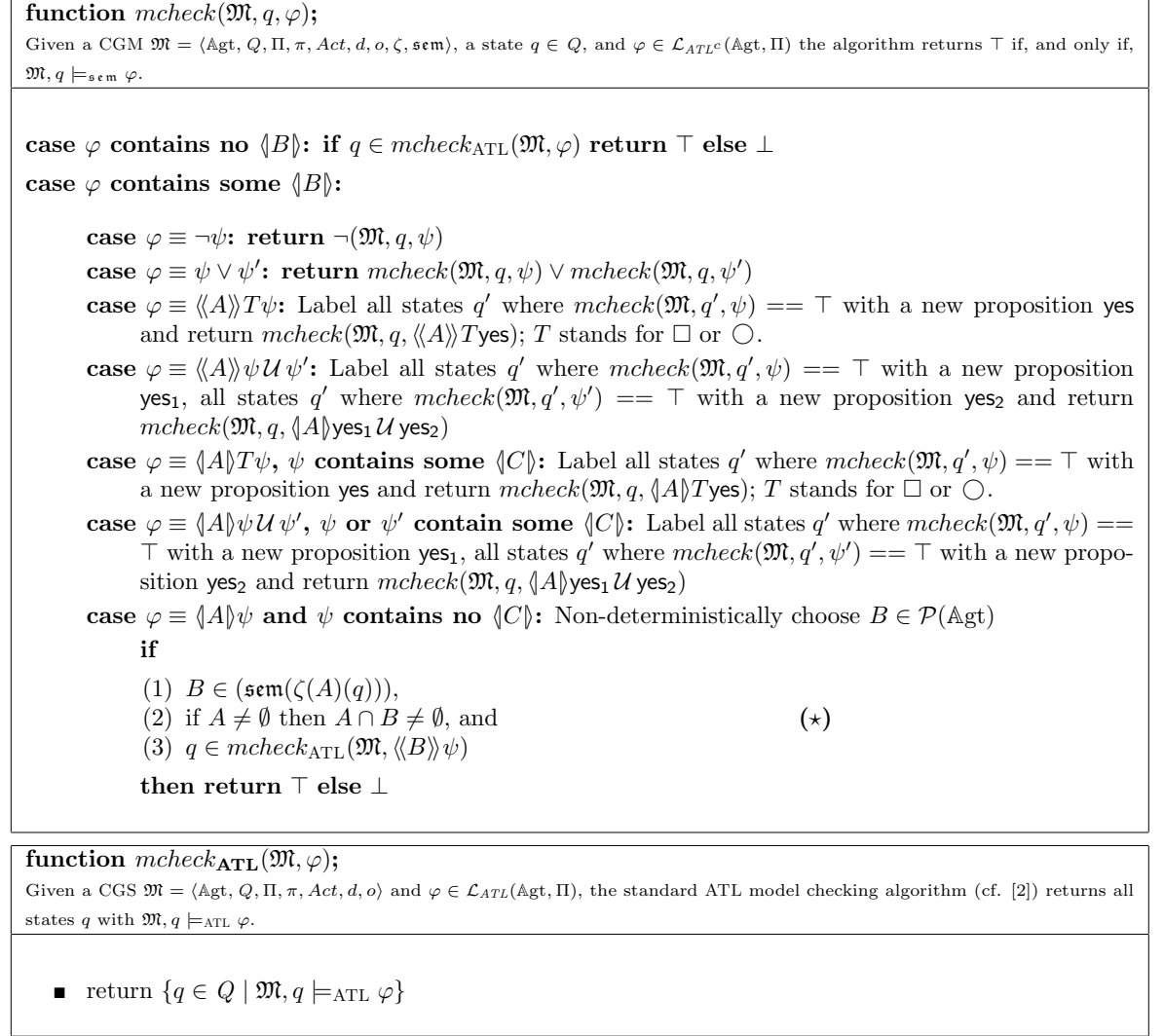
For CoALATL we also have to treat the new coalitional modalities in addition to the normal ATL constructs. Let us consider the formula $\langle A \rangle \psi$. According to the semantics of $\langle A \rangle$, given in Definition 15, we must check whether there is a coalition B such that (i) if $A \neq \emptyset$ then $A \cap B \neq \emptyset$, (ii) B is acceptable by the argumentation semantics, and (iii) $\langle B \rangle \psi$. The number of possible candidate coalitions B which satisfy (i) and (ii) is bounded by $|\mathcal{P}(\text{Agt})|$. Thus, in the worst case there might be *exponentially* many calls to a procedure checking whether $\langle B \rangle \psi$. Another source of complexity is the time needed to compute the argumentation semantics.

Both considerations together suggest that the model checking complexity has two computationally hard parts: exponentially many calls to $\langle B \rangle \psi$ and the computation of the argumentation semantics. Indeed, Theorem 10 will support this intuition. However, we show that it is possible to “combine” both computationally hard parts to obtain an algorithm which is in $\Delta_2^P = \mathbf{P}^{\mathbf{NP}}$, if the computational complexity to determine whether a given coalition is acceptable is in **NP**.

For the rest of this section, we will denote by $\mathcal{VER}_{\text{sem}}(\mathcal{CF}, A)$ the *verification problem* (cf. [22]) which represents the problem whether for a given argumentation semantics sem , a coalitional framework \mathcal{CF} , and coalition $A \subseteq \text{Agt}$ we have that $A \in \text{sem}(\mathcal{CF})$. Given some complexity class \mathcal{C} , we use the notation “ $\mathcal{VER}_{\text{sem}} \in \mathcal{C}$ ” to state that the verification problem with respect to the semantics sem is in \mathcal{C} .

In [12] it is stated that $\mathcal{VER}_{\text{sem}} \in \mathbf{P}$ for all semantics introduced in Definition 11. Unfortunately, this result is incorrect for the preferred semantics. There is a flaw in the way maximal sets of coalitions are treated. Actually, from [20] it follows that $\mathcal{VER}_{\text{sem}} \in \mathbf{P}$ for $\text{sem} \in \{\text{sem}_{\text{admissible}}, \text{sem}_{\text{grounded}}, \text{sem}_{\text{stable}}\}$ and in [19] it was shown that $\mathcal{VER}_{\text{sem}_{\text{preferred}}} \in \mathbf{coNP}$ -complete. In the following proposition, we summarize these results and also treat the complete semantics. For an overview we refer to [22].

Proposition 9 ([20, 19]) *We have that $\mathcal{VER}_{\text{sem}} \in \mathbf{P}$ for all semantics $\text{sem} \in \{\text{sem}_{\text{admissible}}, \text{sem}_{\text{grounded}}, \text{sem}_{\text{stable}}, \text{sem}_{\text{complete}}\}$ and $\mathcal{VER}_{\text{sem}_{\text{preferred}}} \in \mathbf{coNP}$ -complete.*

Figure 4: A model checking algorithm for CoALATL

Proof. It remains to show the case for $\mathcal{VER}_{\text{semcomplete}}$. Note that computing $\mathcal{F}_{CF}(\mathfrak{S})$ for a given \mathfrak{S} can be done in polynomial time. Therefore, to check whether a set of coalitions \mathfrak{S} is complete, we can check that it is admissible and that all elements of $\mathcal{F}_{CF}(\mathfrak{S})$ are already contained in \mathfrak{S} . Both checks can be done in polynomial time. Thus $\mathcal{VER}_{\text{semcomplete}} \in \mathbf{P}$. ■

In Figure 4 we propose a model checking algorithm for CoALATL . The complexity result given in the next theorem is modulo the complexity needed to solve the verification problem $\mathcal{VER}_{\text{sem}}$.

Theorem 10 (Model checking CoALATL [12]) *Let a CGM $\mathfrak{M} = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem} \rangle$ be given, $q \in Q$, $\varphi \in \mathcal{L}_{ATL^c}(\text{Agt}, \Pi)$, and $\mathcal{VER}_{\text{sem}} \in \mathcal{C}$. Model checking CoALATL with respect to the argumentation semantics sem^τ is in \mathbf{P}^{NP^c} .*

Proof. The algorithm $mcheck_{ATL}$ is \mathbf{P} -complete [2]. The number of modalities $\langle\!\langle \cdot \rangle\!\rangle$ is bounded by $|\varphi|$. Thus, the last case can only be performed a polynomial number of times (wrt. the length of φ). The complexity of the last case is as follows. Firstly, B is guessed and then verified. The verification is

⁷That is, whether $\mathfrak{M}, q \models_{\text{sem}} \varphi$.

performed by an oracle call (with complexity \mathcal{C}) to check whether $B \in \mathbf{sem}(\zeta(A)(q))$ and two additional steps which can be performed by a deterministic Turing machine in polynomial time. ■

The last theorem gives an upper bound for model checking CoALATL with respect to an arbitrary but fixed semantics \mathbf{sem} . A finer grained classification of the computational complexity of $\mathcal{VER}_{\mathbf{sem}}$ allows to improve the upper bound given in Theorem 10. Assume that $\mathcal{VER}_{\mathbf{sem}} \in \mathbf{NP}$; then, a witness can be non-deterministically guessed together with the coalition $B \in \mathcal{P}(\text{Agt})$ and then, it is checked whether B satisfies the three conditions (1-3) in (\star) . Each of the three cases can be done in deterministic polynomial time. Hence, the verification of $\mathfrak{M}, q \models \langle A \rangle \psi$, in the last case, meets the “guess and verify” principle which is characteristic for problems in \mathbf{NP} . This brings the overall complexity of the algorithm to $\Delta_2^{\mathbf{P}}$.

Corollary 11 ([12]) *If $\mathcal{VER}_{\mathbf{sem}} \in \mathbf{NP}$ then model checking CoALATL is in $\Delta_2^{\mathbf{P}}$ with respect to \mathbf{sem} .*

Proof. Since $\mathcal{VER}_{\mathbf{sem}}$ is in \mathbf{NP} there is a deterministic Turing machine M that runs in polynomial time $p(n)$ (where n is the length of the input (A, \mathcal{CF})) that accepts $((A, \mathcal{CF}), w)$ for some witness w of length less or equal $p(n)$ iff $A \in \mathbf{sem}(\mathcal{CF})$ (otherwise it does not accept $((A, \mathcal{CF}), w)$ for all witnesses w with $|w| \leq p(n)$). Now, in the model checking algorithm, we extend the non-deterministic guess of the coalition B in (\star) by also guessing a witness w for the input of machine M . This can be implemented by a single non-deterministic machine (e.g., using the deterministic machine M as oracle or implementing it directly). Then, the whole algorithm is in $\mathbf{P}^{\mathbf{NP}^{\mathbf{P}}} = \mathbf{P}^{\mathbf{NP}} = \Delta_2^{\mathbf{P}}$. ■

In the line with Proposition 9 we modify the result from [12] as follows.

Corollary 12 *Model checking CoALATL is in $\Delta_2^{\mathbf{P}}$ for $\mathbf{sem}_{\text{admissible}}$, $\mathbf{sem}_{\text{complete}}$ and $\mathbf{sem}_{\text{stable}}$.*

The following result is immediate as $\mathcal{VER}_{\mathbf{sem}_{\text{preferred}}}$ is \mathbf{coNP} -complete.

Corollary 13 *Model checking CoALATL is in $\Delta_3^{\mathbf{P}}$ for $\mathbf{sem}_{\text{preferred}}$.*

As the next proposition shows, the model checking algorithm can also be improved in the cases that only polynomially many coalitions are acceptable wrt to the semantics and that all these coalitions can be computed in polynomial time.

Proposition 14 *Model checking CoALATL is \mathbf{P} -complete for semantics \mathbf{sem} that only accept polynomially many coalitions and for which it is possible to enumerate all theses coalitions in polynomial time with respect to the size of the model and the length of the formula.*

Proof. For a formula $\langle A \rangle \gamma$ we verify whether $\langle B \rangle \gamma$ for all the polynomially many coalitions B acceptable by \mathbf{sem} . Completeness follows from the completeness of model checking pure ATL [2]. ■

Since the grounded semantics is characterized by the smallest fixed point there only is a unique coalition. So, the following result is immediate.

Corollary 15 *Model checking CoALATL is \mathbf{P} -complete for $\mathbf{sem}_{\text{grounded}}$.*

6.2 Definitions and NP/coNP-hardness

In the next section we show that model checking CoALATL is $\Delta_2^{\mathbf{P}}$ -hard even for the very simple argumentation semantics that can essentially characterizes *all* truth assignments of Boolean formulae. In this section, we firstly show that the model checking problem is \mathbf{NP} -hard and \mathbf{coNP} -hard by reducing SAT [33] (satisfiability of Boolean formulae (in positive normal form)) to model checking CoALATL and introduce the basic definitions needed for the $\Delta_2^{\mathbf{P}}$ -hardness proof presented in the following section.

Let $\varphi = \varphi(X)$ be a Boolean formula in positive normal form⁸ over the Boolean variables $X := \{x_1, \dots, x_n\}$. A *truth assignment* of a Boolean formula $\varphi(X)$ is a mapping $X \rightarrow \{0, 1\}$. We identify a truth assignment with the set $X' \subseteq X$ of variables that are assigned to 1 (true). Firstly, we define a necessary condition on the expressiveness of argumentation semantics which forces the problem to become complete.

⁸That is, negation symbols do only occur at variables. Note that the positive normal form can be established in polynomial time.

Definition 27 (Reduction-suitable semantics, \mathbf{sem} -witness) Let X_1 and X_2 be two non-empty and disjoint sets of the same size and $f : X_1 \rightarrow X_2$ a bijective mapping between these sets and let x be an element not in $X_1 \cup X_2$. Moreover, let $X_1 \cup X_2 \cup \{x\} \subseteq Y$ for some set Y .

We call a semantics \mathbf{sem} over Y reduction-suitable if for any sets $X_1, X_2, \{x\}$ satisfying the properties given above there is a coalitional framework whose size is polynomial in Y such that $E \in \mathbf{sem}(\mathcal{CF})$ iff (1) $E = X \cup \{x\}$, (2) $X \subseteq X_1 \cup X_2$ and (3) $\forall x \in X_1 \cup X_2$ ($x \in X$ iff $f(x) \notin X$).

Moreover, we call a coalitional framework that witnesses that a semantics is reduction-suitable a \mathbf{sem} -witness coalitional framework.

A reduction-suitable semantics allows to describe all truth assignments of a formula in a compact way. The intuition is that X_1 and X_2 represent the variables and their negations, respectively. The set X with $X \cup \{x\} \in \mathbf{sem}(\mathcal{CF})$ encodes a truth assignment; i.e. the *literals* assigned 1. The variable x is a technicality needed in the reduction. In the following proposition we make the observation that reduction-suitable semantics allow to represent exactly all truth assignments.

Proposition 16 Let $\varphi = \varphi(X)$ be a Boolean formula and let $\bar{X} := \{\bar{x} \mid x \in X\}$ and let \mathbf{sem} be a reduction-suitable semantics over Y where $X \cup \bar{X} \cup \{x\} \subseteq Y$ and $x \notin X$. Then, there is a coalitional framework \mathcal{CF} such that for each $T \in \mathbf{sem}(\mathcal{CF})$, $T \cap X$ is a truth assignment of φ and for each truth assignment T of φ there is a $Z \in \mathbf{sem}(\mathcal{CF})$ such that $Z \cap X = T$.

Proof. Let \mathcal{CF} be a \mathbf{sem} -witness. Firstly, let $T \in \mathbf{sem}(\mathcal{CF})$, clearly $T \cap X$ does only contain elements of X and thus is a truth assignment. Now let, T be a truth assignment of φ . We show that $Z \cup \{x\} \in \mathbf{sem}(\mathcal{CF})$ with $Z := T \cup X'$, $X' := \{\bar{x} \mid x \notin T\}$ and $(Z \cup \{x\}) \cap X = T$. Clearly, we have that $Z \subseteq X \cup \bar{X}$ and also that $(x \in Z \text{ iff } \bar{x} \notin Z)$ for all $\forall x \in X \cup \bar{X}$. This shows that both conditions of reduction-suitable semantics are satisfied. ■

In the following we introduce some notation to refer to subformulae of a formula and to the outermost logical Boolean connector \vee or \wedge .

Definition 28 (Notation for subformulae of φ , relevance) Let φ be a Boolean formula in positive normalform. We define $lc(\varphi)$ as the outermost logical connector in φ ; that is,

$$lc(\varphi) := \begin{cases} \wedge & \text{if } \varphi = \psi_1 \wedge \psi_2 \\ \vee & \text{if } \varphi = \psi_1 \vee \psi_2 \\ \epsilon & \text{if } \varphi \text{ is a literal.} \end{cases}$$

Moreover, we define $ls(\varphi)$ (resp. $rs(\varphi)$) as the subformula on the left hand side (resp. right hand side) of $lc(\varphi)$ provided that $lc(\varphi) \neq \epsilon$. In the case of $lc(\varphi) = \epsilon$ we set $ls(\varphi) = rs(\varphi) = \varphi$. Note that we have that $\varphi = ls(\varphi)lc(\varphi)rs(\varphi)$ whenever $lc(\varphi) \neq \epsilon$.

Now, we can assign a string over $\{1, 2\}$ to refer to a subformula of φ , where 1 (resp. 2) stands for the left (resp. right) subformula wrt to the outermost logical connector. Formally, we define a function χ^φ from $\{1, 2\}^+ \cup \{0\}$ into the subformulae of φ as follows (we write χ_w^φ for $\chi^\varphi(w)$ where $w \in \{1, 2\}^+ \cup \{0\}$):

$$\chi_w^\varphi := \begin{cases} \varphi & \text{if } w = 0 \\ ls(\varphi) & \text{if } w = 1 \\ rs(\varphi) & \text{if } w = 2 \\ ls(\chi_x^\varphi) & \text{if } w = x1, x \in \{1, 2\}^+ \\ rs(\chi_x^\varphi) & \text{if } w = x2, x \in \{1, 2\}^+. \end{cases}$$

Finally, we call a string $w \in \{1, 2\}^+$ relevant for φ iff $lc(\chi_x^\varphi) \neq \epsilon$ for $x \in \{1, 2\}^*$ and $w = xi$ with $i \in \{1, 2\}$; or if $w = 0$. We will also just write χ_w if the formula φ is clear from context.

Given the formula $\varphi = ((x_1 \wedge x_2) \vee \neg x_3) \wedge (\neg x_1 \vee x_3)$, for instance, we have that $\chi_2^\varphi = \neg x_1 \vee x_3$, $\chi_{112}^\varphi = x_2$, and $\chi_{21}^\varphi = \neg x_1$,

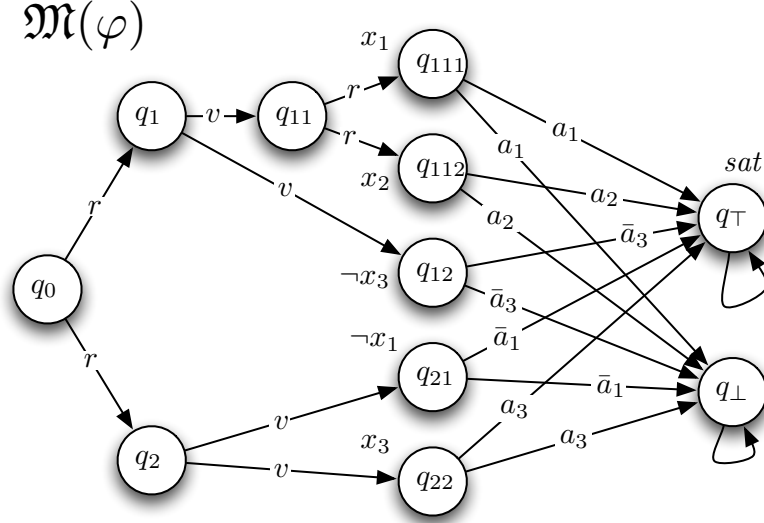


Figure 5: The construction of $\mathfrak{M}(\varphi)$ for $\varphi = ((x_1 \wedge x_2) \vee \neg x_3) \wedge (\neg x_1 \vee x_3)$. Transitions labeled with an agent indicate that this agent can force the transition disregarding the behavior of the other agents.

We proceed with our reduction. Inspired by [15, 28] we construct a CGM corresponding to $\varphi(X)$ which essentially corresponds to the parse tree of $\varphi(X)$ and implements the game semantics of Boolean formulae (cf. [25]).

That is, we construct the CGM $\mathfrak{M}(\varphi)$ corresponding to $\varphi(X)$ with $2 + 2|X|$ players: *verifier* \mathbf{v} , *refuter* \mathbf{r} , and agents a_i and \bar{a}_i for each variable $x_i \in X$. The CGM is turn-based, that is, every state is “governed” by a single player who determines the next transition. Each subformula $\chi_{i_1 \dots i_l}$ of φ has a corresponding state $q_{i_1 \dots i_l}$ in $\mathfrak{M}(\varphi)$ for $i_k \in \{0, 1\}$. If the outermost logical connective of φ is \wedge , i.e. $lc(\varphi) = \wedge$, the refuter decides at q_0 which subformula χ_i of φ is to be satisfied (i.e. whether χ_1 or χ_2), by proceeding to the “subformula” state q_i corresponding to χ_i . If the outermost connective is \vee , the verifier decides which subformula χ_i of φ will be attempted at q_0 . This procedure is repeated until all subformulae are single literals. In the following we refer to the states corresponding to literals as *literal states*.

The difference from the construction from [28] is that formulae are in positive normal form (rather than CNF) and from [28, 15] in the way in that literal states are treated: literal states are governed by the agents a_i or \bar{a}_i . The values of the underlying propositional variables x, y are declared at the literal states, and the outcome is computed. That is, if a_j executes \top for a positive literal, i.e. $\chi_{i_1 \dots i_l} = x_j$, at $q_{i_1 \dots i_l}$, then the system proceeds to the “winning” state q_\top ; otherwise, the system goes to the “sink” state q_\perp . Analogously, if \bar{a}_j executes \perp for a negative literal, i.e. $\chi_{i_1 \dots i_l} = \neg x_j$, at $q_{i_1 \dots i_l}$, then the system proceeds to the “winning” state q_\top ; otherwise, the system goes to the “sink” state q_\perp .

Finally, the idea is to use *sem*-witness coalitional frameworks (for some reduction-suitable semantics) to represent all valuations of $\varphi(X)$ such that there is a “successful coalition” among these coalitions (representing the valuations of φ) iff $\varphi(X)$ is satisfiable. An example of the construction is shown in Figure 5. Formally, the model is defined as follows.

Definition 29 ($\mathfrak{M}(\varphi)$) *Let $\varphi(X)$ be given. The model $\mathfrak{M}(\varphi) = \langle \text{Agt}, Q, \Pi, \pi, \text{Act}, d, o, \zeta, \text{sem} \rangle$ is defined as follows:*

- $\text{Agt} := \{\mathbf{v}, \mathbf{r}\} \cup \{a_i, \bar{a}_i \mid x_i \in X\}$
- $Q := \{q_0, q_\top, q_\perp\} \cup \{q_w \mid w \in \{1, 2\}^+ \text{ relevant for } \varphi\}$
- $\Pi := \{\text{sat}\}$

- $\pi(q_\top) = \{\text{sat}\}$
- $\text{Act} := \{1, 2, \top, \perp\}$
- $d_r(q_w) = \{1, 2\}$ for each w relevant for φ with $lc(\chi_w^\varphi) = \wedge$; $d_v(q_w) = \{1, 2\}$ for each w relevant for φ with $lc(\chi_w^\varphi) = \vee$; $d_{a_i}(q_w) = \{\top, \perp\}$ for each w relevant for φ with $\chi_w^\varphi = x_i$; $d_{\bar{a}_i}(q_w) = \{\top, \perp\}$ for each w relevant for φ with $\chi_w^\varphi = \neg x_i$; $d_x(q) = \{\top\}$ for $q \in \{q_\top, q_\perp\}$ and $x \in \text{Agt}$.
- Note that the model is turn-based (except for the states q_\top and q_\perp); that is, in each state only one agent can execute actions. Hence, transitions do only depend on single actions and not on action profiles: $o(q_0)(1) = q_1$, $o(q_0)(2) = q_2$, $o(q_w)(1) = q_{w1}$ if $w1$ is relevant for φ , $o(q_w)(2) = q_{w2}$ if $w2$ is relevant for φ ; $o(q_w)(\top) = q_\top$ (resp. $o(q_w)(\perp) = q_\perp$) if $\chi_w^\varphi \in X$; $o(q_w)(\top) = q_\perp$ (resp. $o(q_w)(\perp) = q_\top$) if $\neg\chi_w^\varphi \in X$ (we identify x with $\neg\neg x$); and $o(q)(\top) = q$ for $q \in \{q_\top, q_\perp\}$.
- **sem** is a reduction-suitable semantics over Agt . Finally, we set $\zeta(A)(q) = \mathcal{CF}$ where \mathcal{CF} is some **sem**-witness coalitional framework for the sets $X_1 := \{a_i \mid x_i \in X\}$, $X_2 := \{\bar{a}_i \mid x_i \in X\}$, and $x := \{v\}$ for all $A \subseteq \text{Agt}$ and $q \in Q$.

We define a **v**-coice of $\mathfrak{M}(\varphi)$ as “the graph” that occurs if from states controlled by the verifier **v** all transitions but one are removed (i.e. at each state controlled by the verifier it does only have one action to execute). With other words, we fix a strategy of **v**. The following lemma is essential for our reduction.

Lemma 17 *Let $\varphi(X)$ be a Boolean formula in positive normal form.*

- If T is a satisfying truth assignment of φ , then there is a **v**-choice of $\mathfrak{M}(\varphi)$ such that for the set L of all literal states reachable from q_0 it holds that $\{x \in X \mid q_w \in L \text{ and } \chi_w = x\} \subseteq T$ and $\{x \in X \mid q_w \in L \text{ and } \chi_w = \neg x\} \cap T = \emptyset$.
- If there is a **v**-choice of $\mathfrak{M}(\varphi)$ such that for the set L of all literal states reachable from q_0 we have that for any $q_w, q_v \in L$ the formula $\chi_w \wedge \chi_v$ is satisfiable (i.e. there are no complementary literals) then the set $\{x \in X \mid q_w \in L \text{ and } \chi_w = x\}$ is a satisfying truth assignment of φ .

Proof.

- Let T be a satisfying truth assignment of φ . We construct a **v**-choice with the stated property. Firstly, we call a state q_w a **v**-state (resp. **r**-state) if q_w is controlled by **v** (resp. **r**). Next, we consider a state q_w such that q_{w1} and q_{w2} are literal states. Firstly, let us consider the case that q_w is a **v**-state. Then, we label it with \top if for some i we have that $x \in T$ (resp. $x \notin T$) if $\chi_{w_i} = x$ (resp. $\chi_{w_i} = \neg x$) for $x \in X$; otherwise we label the state with \perp . Secondly, if q_w is a **r**-state we label it with \top if the above condition holds for *both* i ; otherwise with \perp .

In the second step, we label each **v**-state with two labelled successor states with \top if at least one of its successors has the label \top ; otherwise we assign to it the label \perp . On the other hand, if it is an **r**-state, if both successors have the label \top we label it \top ; otherwise we assign to it the label \perp . We proceed like this until all states q_w with a relevant w are labelled.

Now, in lexicographical order we go through all w relevant for φ and perform the following steps. If q_w is a **v**-state reachable from q_0 , we remove the transition to the successor which is labelled \perp and if both transitions are labeled \top we remove any of its transitions. Else, if q_w is a **v**-state and *not* reachable from q_0 we remove any of its transitions. Following this procedure, we have to show that there is no reachable **v**-state such that both successors are labelled \perp . Suppose such a state q_w is reachable. Since the verifier has no strategy to avoid this state (note that w is lexicographically minimal) the disjunction $\chi_{w1} \vee \chi_{w2}$ must be true under T in order to make φ true. However, since both states q_{w1} and q_{w2} are labelled \perp the truth assignment T falsifies χ_{w1} and χ_{w2} (that can easily be seen by induction). Contradiction!

Hence, the construction yields a **v**-choice. We need to show that there is a **v**-choice such that $S := \{x \in X \mid q_w \in L \text{ and } \chi_w = x\}$ is a subset of T . Firstly, assume there is some $x \in S$ not in T . Let q_{wi} be some reachable literal state with $\chi_{wi} = x$. By construction of the **v**-choice, this

implies that the state q_w is an **r**-state. (Otherwise, the other alternative would have been selected (according to the labelling algorithm); or, both successors of q_w would be labelled \perp what is not possible as shown above.) But this means, that the formula $\chi_{w1} \wedge \chi_{w2}$ needs to be satisfied and hence $x \in T$. Secondly, assume that $x \in \{x \in X \mid q_w \in L \text{ and } \chi_w = \neg x\} \cap T$ and let q_{wi} be the state reachable with $\chi_{wi} = \neg x$. Following the same reasoning as above the state q_w must be an **r**-state and this contradicts the fact the the formula $\chi_{w1} \wedge \chi_{w2}$ must be true under T .

- (b) Let us consider a **v**-choice v of $\mathfrak{M}(\varphi)$ with the stated property and suppose that $T := \{x \in X \mid q_w \in L \text{ and } \chi_w = x\}$ is not a satisfying truth assignment of φ . Note that the **v**-choice corresponds to the selection of the left (resp. right) hand subformula of any subformula χ_w with $lc(\chi_w) = \vee$. We say that a subformula χ_w is reachable if the state q_w is reachable in the very **v**-choice v . Since T is not a satisfying truth assignment there is some reachable subformula (possibly φ itself) not satisfied by T . Let χ_w be such a reachable subformula with lexicographically maximal w (so, it is a relative “small” subformula). We consider two cases. Firstly, suppose that χ_w is not satisfiable. Then, due to the maximality of w , χ_w must be a conjunction of literals among that are two complementary ones, we denote them by χ_v and $\chi_{v'}$ (otherwise it would be satisfiable); hence, we have that $q_v, q_{v'} \in L$. Contradiction!

So, suppose χ_w is satisfiable but false under T . Then, due to the maximality of w , χ_w can either be x_i for some $x_i \notin T$ or be $\neg x_j$ for some $x_j \in T$. Suppose $\chi_w = x_i$; then q_w is reachable and thus $x_i \in T$. Contradiction! On the other hand, if $\chi_w = \neg x_j$ then $q_w \in L$ and since $x_j \in T$ there is some other state $q_v \in L$ with $\chi_v = x_j$ and $\chi_w \wedge \chi_v$ not satisfiable. Contradiction! ■

Similar to [15, 28], we have the following result which shows that the construction above is a polynomial-time reduction of SAT to model checking CoALATL .

Proposition 18 *The model $\mathfrak{M}(\varphi)$ is constructible in polynomial-time wrt the size of φ and we have that*

$$\varphi(X) \text{ is satisfiable if, and only if, } \mathfrak{M}(\varphi), q_0 \models \langle \mathbf{v} \rangle \Diamond \text{ sat.}$$

Proof. Firstly, we analyze the construction to show that its length is polynomially wrt the input. The number of agents is polynomially in X ; the number of states is polynomially in the number of subformulae of φ and for each state there are at most two transitions (apart from the states q_\perp and q_\top).

“ \Rightarrow ”: Let $\varphi(X)$ be satisfiable and let $T \subseteq X$ be a satisfying truth assignment. By construction of $\mathfrak{M}(\varphi)$ we have that $C := \{\mathbf{v}, a_i, \bar{a}_j \mid x_i \in T, x_j \notin T\} \in \mathbf{sem}(\zeta(\{\mathbf{v}\})(q_0))$; hence, it suffices to show that $\mathfrak{M}(\varphi), q_0 \models \langle \mathbf{v} \rangle \Diamond \text{ sat}$. Now by Lemma 17(a) there is a strategy s_v of **v** such that for the set L of all literal states reachable from q_0 we have that $S := \{x \mid q_w \in L \text{ and } \chi_w = x\} \subseteq T$ and $\{x \mid q_w \in L \text{ and } \chi_w = \neg x\} \cap T = \emptyset$. Hence, for any $x_i \in S$ the agent a_i is in C and for any $x_i \in \{x \mid q_w \in L \text{ and } \chi_w = \neg x\}$ the agent \bar{a}_i is in C . Finally, the strategy s_a of $a \in C \setminus \{\mathbf{v}\}$ is to execute \top if $a = a_i$ and \perp if $a = \bar{a}_i$ in the states controlled by a . The complete strategy profile consisting of s_b for $b \in C$ ensures $\Diamond \text{ sat}$.

“ \Leftarrow ”: Suppose that $\mathfrak{M}(\varphi), q_0 \models \langle \mathbf{v} \rangle \Diamond \text{ sat}$. Then there is a coalition $C \in \mathbf{sem}(\zeta(\{\mathbf{v}\})(q_0))$ such that $\mathfrak{M}(\varphi), q_0 \models \langle C \rangle \Diamond \text{ sat}$. By Lemma 17(b) it remains to show that for the set L of all literal states reachable from q_0 we have that for any $q_w, q_v \in L$ the formula $\chi_w \wedge \chi_v$ is satisfiable. Suppose the contrary; that is, there are $q_w, q_v \in L$ such that $\chi_w \wedge \chi_v \equiv \perp$; say $\chi_w = x_i$. By Proposition 16, $C \setminus \{\mathbf{v}\}$ does correspond to a truth assignment of φ ; hence, either $a_i \in C$ or $\bar{a}_i \in C$. Hence, if $\bar{a}_i \in C$ then $a_i \notin C$ and the opponents (to whom a_i belongs) have a strategy to reach q_\perp from q_w . But this contradicts $\mathfrak{M}(\varphi), q_0 \models \langle C \rangle \Diamond \text{ sat}$. The same reasoning is applied if $a_i \in C$ and $\bar{a}_i \notin C$. ■

The following result is obvious, we can reduce SAT and UNSAT to model checking $\mathfrak{M}(\varphi), q_0 \models \langle \mathbf{v} \rangle \Diamond \text{ sat}$ and $\mathfrak{M}(\varphi), q_0 \models \neg \langle \mathbf{v} \rangle \Diamond \text{ sat}$, respectively.

Theorem 19 *Model checking CoALATL is NP-hard and coNP-hard for any reduction-suitable semantics.*

6.3 Δ_2^P -hardness

Finally, we show our main result, the Δ_2^P -hardness of model checking CoALATL for reduction-suitable semantics. We do so by reducing SNSAT_1 , a typical Δ_2^P -complete problem (cf. [32]).

Definition 30 (SNSAT₁ [32])

Input: p sets of propositional variables $X^r = \{x_1^r, \dots, x_k^r\}$; p propositional variables z^r and p Boolean formulae φ_r in positive normal form (i.e., negation is allowed only on the level of literals) for $r = 1, \dots, p$. Each φ_r involves only variables from $X^r \cup \{z^1, \dots, z^{r-1}\}$, with the following requirement: $z^r \equiv \exists Y_r(\varphi_r(z^1, \dots, z^{r-1}, X^r))$ where $Y_r \subseteq X^r$. (The notation is understood as follows: there is a truth assignment assigning 1 (resp. 0) to the variables in Y_r (resp. $X^r \setminus Y_r$) such that $\varphi_r(z^1, \dots, z^{r-1}, X^r)$ is true under this assignment.)

Output: The value of z^p .

We use $I = (\varphi_1(X^1), \dots, \varphi_p(X^p))$ or just $I = (\varphi_1, \dots, \varphi_p)$ to denote an instance of SNSAT₁ and set $Z = \{z^1, \dots, z^p\}$.

We often need to analyze a solution of an SNSAT₁-instance. In the following we show how a solution can formally be stated.

Definition 31 (Witness and solution of an SNSAT₁-instance) Let $I = (\varphi_1, \dots, \varphi_p)$ be an SNSAT₁-instance. A tuple (T_1, \dots, T_p) is an I -witness if it satisfies the following properties:

1. $T_i \subseteq \{z^1, \dots, z^i\} \cup X^i$;
2. If φ_i is satisfiable under the partial assignment $\{z^j \in T_j \mid j < i\}$ then T_i is a satisfying truth assignment of φ_i and $z^i \in T_i$; else $z^i \notin T_i$ and $T_i \cap X^i = \emptyset$;
3. $z^i \in T_i \Rightarrow z^i \in T_j$ for all $j \geq i$,

An I -witness T is a solution of I iff $z^p \in T_p$.

In the next proposition we state some properties about solutions and witnesses.

Proposition 20 Let $I = (\varphi_1, \dots, \varphi_p)$ be an SNSAT₁ instance and let $T = (T_1, \dots, T_p)$ be an I -witness.

- (a) Let $T' = (T'_1, \dots, T'_p)$ be another I -witness; then, $T'_i \cap Z = T_i \cap Z$ for all $i = 1, \dots, p$. (That is, among all witnesses the values for the z^i 's are uniquely determined.)
- (b) If $I^i = (\varphi_1, \dots, \varphi_i)$ with $i < p$ has a solution, then $T = (T_1, \dots, T_i)$ is a solution of I^i .
- (c) Let $I' = (\varphi_1, \dots, \varphi_p, \varphi_{p+1})$ be such that I' has no solution; then, $T' = (T_1, \dots, T_p, \{z \in T_p\})$ is an I' -witness.

Proof.

- (a) Suppose that i is the minimal index for which both sets $T_i \cap Z$ and $T'_i \cap Z$ differ; that is, wlog, $z_i \in T_i$ and $z_i \notin T'_i$. This is a contradiction to property (2) of the definition of a witness; as the satisfiability of φ_i under z_1, \dots, z_{i-1} is uniquely determined.
- (b) Follows from (a).
- (c) We show that $R := \{z \in T_p\}$ satisfies the conditions (1-3) in the Definition of a witness. Clearly, $R \subseteq \{z^1, \dots, z^{p+1}\}$. Since I' has no solution, there is no way to make z_{p+1} true. Clearly, $z_{p+1} \notin R$ and $R \cap X^{p+1} = \emptyset$. Finally, condition (3) is satisfied by definition of R . ■

Our reduction of SNSAT₁ is a modification of the reduction of SNSAT₂ presented in [15, 28] and extends the NP/coNP-hardness construction of the previous section. Consider an SNSAT₁ instance $I = (\varphi_1, \dots, \varphi_p)$. Essentially, we construct models $\mathfrak{M}(\varphi_r)$ for $r = 1, \dots, p$ as shown above but we label each state of $\mathfrak{M}(\varphi_r)$ and each agent name by an additional superindex r (that is, states are denoted by q_w^r and agents by a_i^r and \bar{a}_i^r). The main difference is how the literal states corresponding to literals z_i and $\neg z_i$ are treated. We connect such states of model $\mathfrak{M}(\varphi_r)$ with $r > 1$ with the initial state q_0^{r-1} of model $\mathfrak{M}(\varphi_{r-1})$. Additionally, states referring to negated variables z_i are labeled with a proposition **neg**. Finally, the full model $\mathfrak{M}(I)$ wrt an SNSAT₁ instance I is given by the combination of these model as just explained. In particular, the set of agents is given by $\{\mathbf{v}, \mathbf{r}\} \cup \{a_i^r, \bar{a}_i^r \mid x_i^r \in \bigcup_{j=1}^p X^j\}$. An example of

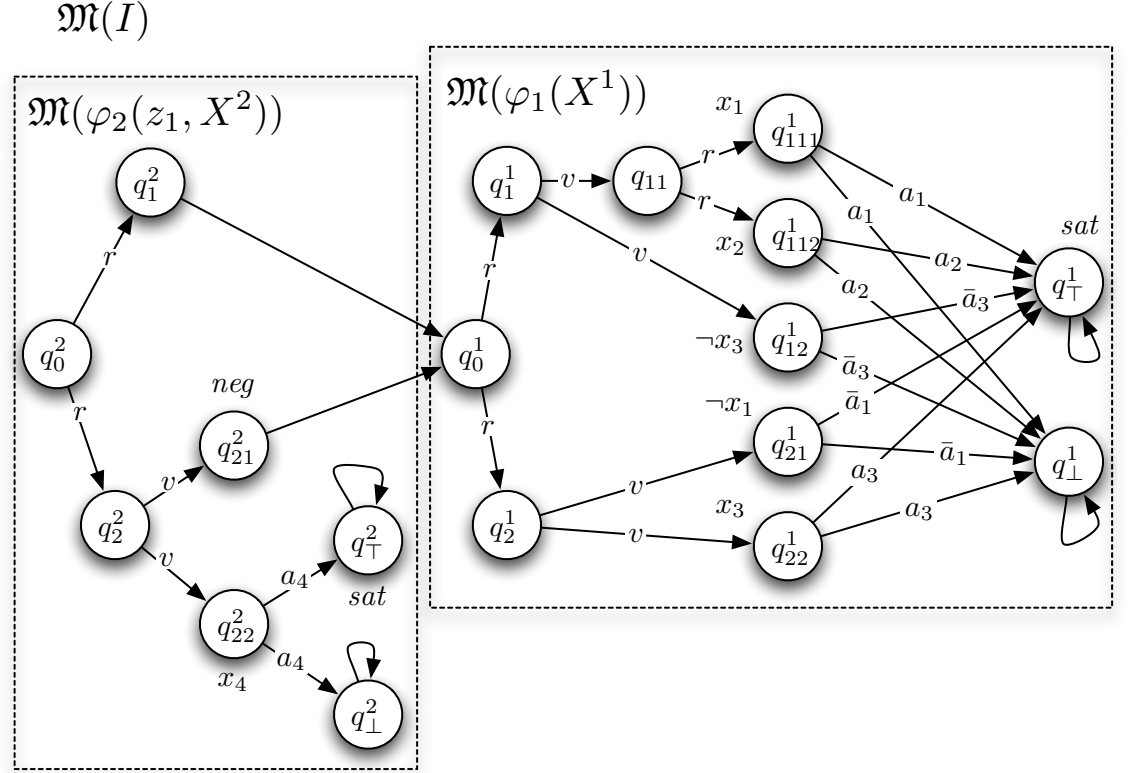


Figure 6: The construction of $\mathfrak{M}(I)$ for $\varphi_1 = ((x_1 \wedge x_2) \vee \neg x_3) \wedge (\neg x_1 \vee x_3)$ and $\varphi_2 = z_1 \wedge (\neg z_1 \vee x_4)$. Transitions labeled with an agent indicate that this agent can force the transition disregarding the behavior of the other agents.

the construction is shown in Figure 6. Given the model $\mathfrak{M}(I)$ we use $\mathfrak{M}(\varphi_i)$ to refer the restriction of $\mathfrak{M}(I)$ to the states with superindex i . In Figure 6 the two submodels $\mathfrak{M}(\varphi_1)$ and $\mathfrak{M}(\varphi_2)$ are framed.

The formulae used in this reduction are more sophisticated as they have to account for the nested structure of an SNSAT_1 instance. We define

$$\phi_0 \equiv \top$$

and

$$\phi_r \equiv \langle \mathbf{v} \rangle (\neg \text{neg} \mathcal{U} (\text{sat} \vee (\text{neg} \wedge \langle \emptyset \rangle) \bigcirc \neg \phi_{r-1}))$$

for $r = 1, \dots, p$.

Before we come to the theorem proving the reduction we state a fundamental lemma which can be seen as counterpart of Lemma 17.

Lemma 21 *Let $I = (\varphi_1, \dots, \varphi_p)$ be an SNSAT_1 instance.*

- (a) *Let $T = (T_1, \dots, T_p)$ be a solution for I . For all $r = 1, \dots, p$, if $z_r \in T_r$ then there is a \mathbf{v} -choice of $\mathfrak{M}(\varphi_r)$ such that for the set L of all literal states reachable from q_0^r and which belong to $\mathfrak{M}(\varphi_r)$ it holds that $\{x \in X^r \cup Z \mid q_w \in L \text{ and } \chi_w^{\varphi_r} = x\} \subseteq T_r$ and $\{x \in X^r \cup Z \mid q_w \in L \text{ and } \chi_w^{\varphi_r} = \neg x\} \cap T_r = \emptyset$.*
- (b) *Let $I^{p-1} = (\varphi_1, \dots, \varphi_{p-1})$ and let $T^{p-1} = (T_1^{p-1}, \dots, T_{p-1}^{p-1})$ be an I^{p-1} -witness. Then, if there is a \mathbf{v} -choice of $\mathfrak{M}(\varphi_p)$ such that for the set L of all literal states reachable from q_0^p that belong to $\mathfrak{M}(\varphi_p)$ we have that*

- (i) for any $q_w, q_v \in L$ the literals $\chi_w^{\varphi_p}$ and $\chi_v^{\varphi_p}$ are non-complementary; and
- (ii) if $q_v \in L$ with $\chi_v^{\varphi_p} = z_i$ (resp. $\chi_v^{\varphi_p} = \neg z_i$) then $z_i \in T_i$ (resp. $z_i \notin T_i$);

then, $T = (T_1^{p-1}, \dots, T_{p-1}^{p-1}, T_p)$ is a solution for I where

$$T_p = \{x \in X^r \mid q_w \in L \text{ and } \chi_w = x\} \cup \{z^i \mid z^i \in T_i^{p-1}, i < p\} \cup \{z^p\}.$$

- (c) $\mathfrak{M}(I), q_0^i \models \phi_i$ if, and only if, $\mathfrak{M}(I), q_0^i \models \phi_j$; and $\mathfrak{M}(I), q_0^i \models \neg \phi_i$ if, and only if, $\mathfrak{M}(I), q_0^i \models \neg \phi_j$ for all $j \geq i$.

Proof.

- (a) Let $z_r \in T_r$. As in the proof of Lemma 17(a) we proceed with the following labeling: We consider a state q_w such that q_{w1} and q_{w2} are literal states in $\mathfrak{M}(\varphi_p)$. Firstly, let us consider the case that q_w is a **v**-state. Then, we label it with \top if for some i with $\chi_{w_i} = x \in X^r \cup Z$ (resp. $\chi_{w_i} = \neg x$, $x \in X^r \cup Z$) we have that $x \in T_r$ (resp. $x \notin T_r$); otherwise we label the state with \perp . Secondly, if q_w is a **r**-state we label it with \top if the above condition holds for *both* i ; otherwise by \perp . In the second step, (labeling non-literal states) we proceed exactly as before.

In the second step, we label each **v**-state with two labelled successor states with \top if at least one of its successors has the label \top ; otherwise we assign to it the label \perp . On the other hand, if it is a **r**-state, if both successors have the label \top we label it \top ; otherwise we assign to it the label \perp . We proceed like that until all states q_w with a relevant w are labelled.

The **v**-choice is constructed in the very same way as in Lemma 17(a) and also the verification that the **v**-choice has the same properties is done in the very same way.

- (b) Let us consider a **v**-choice with the stated properties (i) and (ii). We adopt the notation from the proof of Lemma 17(b). By definition of T_p it is obvious that it satisfies conditions 1. and 3. of Definition 31.

Suppose that T is not a solution of I ; that is, condition 2. of Definition 31 is violated. Since T^{p-1} is a witness of I^{p-1} there is a reachable subformula χ_w of φ_p with lexicographically minimal w that is not satisfiable given the valuation of the z^i 's according to T^{p-1} (note that the z_i 's are uniquely determined in T^{p-1}). Note, that the condition (ii) guarantees that the “right” value is chosen for $z_i \in T_i$ and $z_i \notin T_i$. The rest of the proof is done analogously to the one given in Lemma 17(b) by considering the literal states corresponding to variables $X^p \cup Z$. This proves that T is a solution.

- (c) Along each path from q_0^i to a state labelled **sat** there are at most $i - 1$ states labelled **neg**. Hence, the truth of a formula ϕ_j with $j \geq i$ is equivalent to the truth of ϕ_i . ■

Similar to [15, 28], we have the following result which shows that the construction above is a polynomial-time reduction of SNSAT_1 to model checking CoALATL .

Theorem 22 *The size of $\mathfrak{M}(I)$ and of the formulae ϕ_p is polynomially in the size of the SNSAT_1 instance $I = (\varphi_1, \dots, \varphi_p)$ and we have the following:*

$$\text{There is a solution } T = (T_1, \dots, T_r) \text{ of } I^r = (\varphi_1, \dots, \varphi_r) \text{ if, and only if, } \mathfrak{M}(I^r), q_0^r \models \phi_l$$

for $l \geq r$ and $r \leq p$.

Proof. That $\mathfrak{M}(I)$ is polynomially wrt φ follows from Proposition 18 and by the way it is constructed. The size of ϕ_p also is polynomially in $\varphi(X)$.

The proof is done by induction on r .

Induction start. The induction starts with $r = 1$. An SNSAT_1 -instance with $r = 1$ is given by $\exists Y_1 \varphi(X^1)$. This corresponds to the satisfiability problem. So, this case is proven in Proposition 18 and Lemma 21(c).

Induction step. Suppose the claim holds up to $r < p$. We show that there is a solution $T = (T_1, \dots, T_{r+1})$ of I^{r+1} iff $\mathfrak{M}(I^{r+1}), q_0^{r+1} \models \phi_{l+1}$.

“ \Rightarrow ”: Let $T = (T_1, \dots, T_{r+1})$ be a solution for I^{r+1} .

From the reduction-suitableness we have that there is a coalition $C \in \mathbf{sem}(\zeta(\{\mathbf{v}\})(q_0^p))$ with

$$C = \{\mathbf{v}\} \cup \bigcup_{i=1, \dots, r+1} \{a_j^i \mid a_j^i \in T_i\} \cup \bigcup_{i=1, \dots, r+1} \{\bar{a}_j^i \mid x_j^i \notin T_i\}$$

Note that, if $z_i \in T_i$ then $T_i = \{x_j^i \mid a_j^i \in C, j = 1, \dots, s\} \cup \{z_j \mid \mathfrak{M}(I^{r+1}), q_0^j \models \phi_j, j = 1, \dots, i\}$.

We show that $\mathfrak{M}(I^{r+1}), q_0^{r+1} \models \langle\langle C \rangle\rangle(\neg \mathbf{neg} \mathcal{U}(\mathbf{sat} \vee (\mathbf{neg} \wedge \langle\langle \emptyset \rangle\rangle \bigcirc \neg \phi_{i+1})))$. Let $s_{\mathbf{v}}^{r+1}$ be the partial strategy corresponding to the \mathbf{v} -choice according to Lemma 21(a). The set of reachable literal states L^{r+1} in $\mathfrak{M}(\varphi_{r+1})$ corresponds to finite sequences of states starting in q_0^{r+1} . In the following, we call such literal states referring to some z^i (resp. $\neg z^i$), z (resp. $\neg z$)-*literal states* and the others x (resp. $\neg x$)-*literal states*. Let λ be a *finite* sequence of states starting in q_0^{r+1} and ending in one of the states in L^{r+1} . Then this sequence has one of the following properties: (1) The last state is an x , z , $\neg x$ -literal state and there is no $\neg z$ -literal state on it; or (2) the last state is a $\neg z$ -literal state and there is no other $\neg z$ -literal state on it. Note, that due to Lemma 21(a) it is not possible to reach two complementary literal states. We show, that we can extend $s_{\mathbf{v}}$ to a strategy s_C that witnesses the truth of ϕ_{i+1} .

Case 1. For a path ending with an x -literal state $\chi_w = x_i^{r+1}$ we have that $x_i^{r+1} \in T_{r+1}$ and thus $a_i^{r+1} \in C$. If this agent executes \top the next state is q_{\top} .

Case 2. For a path ending with an $\neg x$ -literal state $\chi_w = \neg x_i^{r+1}$ we have that $x_i^{r+1} \notin T_{r+1}$ and thus $\bar{a}_i^{r+1} \in C$. If this agent executes \perp the next state is q_{\perp} .

Case 3. For a path ending with an z -literal state $q_w, \chi_w = z^i$, we have that $z^i \in T_{r+1}$ hence also $z^i \in T_i$. By induction hypothesis we have that $\mathfrak{M}(I), q_0^i \models \phi_r$. Now, it is easily seen by another induction that the very same coalition C has a winning strategy s'_C witnessing ϕ_r . We combine the strategy $s_{\mathbf{v}}$ with s'_C .

Case 4. For a path ending with an $\neg z$ -literal state $q_w, \chi_w = \neg z^i$, we have that $z^i \notin T_{r+1}$ hence also $z^i \notin T_i$. By induction hypothesis we have that $\mathfrak{M}(I), q_0^i \not\models \phi_r$; hence, $\mathfrak{M}(I^{r+1}), q_w \models \mathbf{neg} \wedge \langle\langle \emptyset \rangle\rangle \bigcirc \neg \phi_i$.

These cases give us the desired strategy s_C witnessing ϕ_{l+1} for the chosen coalition C .

“ \Leftarrow ”: Suppose that $\mathfrak{M}(I^{r+1}), q_0^{r+1} \models \phi_{l+1}$.

Let z^i with $i < r+1$ and with an maximal index i be such that $\mathfrak{M}(I^i), q_0^i \models \phi_i$. Then, there is a solution $T^i = (T_1, \dots, T_i)$ of I^i by induction hypothesis. According to Proposition 20(b) and (c) we can extend this solution to an I^r witness $T^r = (T_1, \dots, T_r)$ of I^r .

Now, since $\mathfrak{M}(I^{r+1}), q_0^{r+1} \models \phi_{l+1}$, there is a coalition $C \in \mathbf{sem}(\zeta(\mathbf{v})(q_0^{r+1}))$ and a strategy s_C that witnesses the truth of the formula. Let L be the set of all reachable literal states in $\mathfrak{M}(\varphi_{r+1})$ under s_C (a \mathbf{v} -choice is contained implicitly). We show that the preconditions (i) and (ii) of Lemma 21(b) are satisfied by this \mathbf{v} -choice.

Condition (i). Firstly, for each reachable $x/\neg x$ -literal state $q_w^{r+1} \in L$ with $\chi_w^{\varphi_{r+1}} = x$ we must have that a_i^{r+1} (resp. \bar{a}_i^{r+1}) is in C if $x = x_i^{r+1}$ (resp. $x = \neg x_i^{r+1}$) (otherwise the formula $\neg \mathbf{neg} \mathcal{U} \mathbf{sat}$ would not be true). Hence, there cannot be any complementary x -literal states contained in L due to the reduction-suitable semantics (again, otherwise $\neg \mathbf{neg} \mathcal{U} \mathbf{sat}$ would not be true).

Secondly, observe that for any positive z -literal state $q_w^{r+1} \in L$ with $\chi_w^{\varphi_{r+1}} = z^i$ we have that $\mathfrak{M}(I^{r+1}), q_w^{r+1} \models \phi_{l+1}$ and by Lemma 21(c) also $\mathfrak{M}(I^{r+1}), q_0^i \models \phi_i$; hence, by induction hypothesis, $z^i \in T_i$.

For each negative z -literal state $q_w^{r+1} \in L$ with $\chi_w^{\varphi_{r+1}} = \neg z^i$ we have that $\mathfrak{M}(I^{r+1}), q_0^i \models \neg \phi_i$ and thus, by Lemma 21(c) $\mathfrak{M}(I^{r+1}), q_0^i \models \neg \phi_i$. By induction hypothesis, we have that for *any* I^i -witness (T_1', \dots, T_i') it holds that $z^i \notin T_i'$. Contradiction to $z^i \in T_i$.

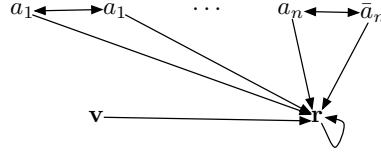


Figure 7: Proof of Corollary 25.

Condition (ii). Let $q_v \in L^p$ with $\chi_v^{\varphi_p} = z_i$. We show that $z_i \in T_i$. Suppose the contrary. Then, $\mathfrak{M}(I^{r+1}), q_0^i \models \neg\phi_i$ and by Lemma 21(c) also $\mathfrak{M}(I^{r+1}), q_0^i \models \neg\phi_{l+1}$. However, since along the path from q_0^{r+1} to q_0^i we always have $\neg\text{neg}$ the coalition C performing strategy s_C witnesses the formula ϕ_{l+1} in q_0^i . Contradiction!

Secondly, assume $q_v \in L$ with $\chi_v^{\varphi_{r+1}} = \neg z_i$ and again, by the sake of contradiction, that $z_i \in T_i$. Then $\mathfrak{M}(I^{r+1}), q_0^i \models \phi_i$ and thus also $\mathfrak{M}(I^{r+1}), q_0^i \models \phi_l$ by Lemma 21(c). However, on the path to $q_v \in L$ we have visited a state labeled neg ; hence, we must have $\mathfrak{M}(I^{r+1}), q_0^i \models \neg\phi_l$. Contradiction!

Now by Lemma 21(b), we can construct a solution for I^{r+1} from T^r . ■

The reduction gives us the following hardness result.

Theorem 23 *Model checking COALATL is Δ_2^P -hard for any reduction-suitable semantics.*

With Corollary 11 we obtain the following completeness result.

Theorem 24 *Model checking COALATL is Δ_2^P -complete for any reduction-suitable semantics sem with $\mathcal{VER}_{\text{sem}} \in \text{NP}$.*

Finally, we show completeness for the stable semantics.

Corollary 25 *Model checking COALATL is Δ_2^P -complete for the $\text{sem}_{\text{stable}}$ semantics.*

Proof. We have to show that the semantics is reduction-suitable; that is, we have to come up with a sem -witness over Agt .

Consider the coalitional framework shown in Figure 7. We show that this is a witness for the reduction suitability of the stable semantics. Let $X = \{x_1, \dots, x_n\}$ and $\bar{X} = \{\bar{x}_1, \dots, \bar{x}_n\}$ be given and $f(x_i) := \bar{x}_i$. We show that the conditions from Definition 27 are met.

“ \Rightarrow ”: Let $E \in \text{sem}_{\text{stable}}$. Clearly, $\mathbf{r} \notin E$ because $\{\mathbf{r}\}$ is not conflict-free. Moreover, $\{\mathbf{v}\} \in E$; as $\{\mathbf{v}\}$ cannot be attacked by E . We also have that $E \subseteq X \cup \bar{X}$. Suppose there is an i such that $\{a_i, \bar{a}_i\} \cap E = \emptyset$. Then, $\{a_i, \bar{a}_i\}$ is not attacked by E . Contradiction. Now suppose that $\{a_i, \bar{a}_i\} \subseteq E$. This is a contradiction since $\{a_i, \bar{a}_i\}$ is not conflict-free.

“ \Leftarrow ”: Suppose $E \cup \{\mathbf{v}\}$ satisfies condition (1), (2), and (3). Clearly, $E \cup \{\mathbf{v}\}$ is conflict-free. Because for each i either a_i or \bar{a}_i is in $E \cup \{\mathbf{v}\}$ every element outside is attacked by some element from $E \cup \{\mathbf{v}\}$. ■

7 Related and Future Work

Related Work. The main inspiration for our work was the powerful argument-based model for reasoning about coalition structures proposed by Amgoud [3]. Indeed, our notion of coalitional framework (Def. 4) is based on the notion of framework for generating coalition structures (FCS) presented in Amgoud’s paper. However, in contrast with Amgoud’s proposal, our work is concerned with extending ATL by argumentation in order to model coalition formation.

Types of Semantics	MC Complexity
$\mathcal{VER}_{\text{sem}} \in \mathcal{C}$	$\mathbf{P}^{\mathbf{NP}^c}$
$\text{sem}_{\text{preferred}}$	$\Delta_3^{\mathbf{P}}$
reduction-suitable	$\Delta_2^{\mathbf{P}}$ -hard
$\mathcal{VER}_{\text{sem}} \in \mathbf{NP}$, reduction-suitable	$\Delta_2^{\mathbf{P}}$ -complete
$\text{sem}_{\text{stable}}$	$\Delta_2^{\mathbf{P}}$ -complete
$\mathcal{VER}_{\text{sem}} \in \mathbf{NP}$	$\Delta_2^{\mathbf{P}}$
$\text{sem}_{\text{admissible}}, \text{sem}_{\text{complete}}$	$\Delta_2^{\mathbf{P}}$
polynomially many coalitions, polynomially enumerable sem	\mathbf{P} -complete
$\text{sem}_{\text{grounded}}$	\mathbf{P} -complete

Figure 8: Overview of the model checking results modulo the complexity of the used argumentation semantics.

Previous research by Hattori *et al.* [24] has also addressed the problem of argument-based coalition formation, but from a different perspective than ours. In [24] the authors propose an argumentation-based negotiation method for coalition formation which combines a logical framework and an argument evaluation mechanism. The resulting system involves several user agents and a mediator agent. During the negotiation, the mediator agent encourages appropriate user agents to join in a coalition in order to facilitate reaching an agreement. User agents advance proposals using a part of the user's valuations in order to reflect the user's preferences in an agreement. This approach differs greatly from our proposal, as we are not concerned with the negotiation process among agents, and our focus is on modelling coalitions within an extension of a highly expressive temporal logic, where coalition formation is part of the logical language.

Modelling argument-based reasoning with bounded rationality has also been the focus of previous research. In [39] the authors propose the use of a framework for argument-based negotiation, which allows for a strategic and adaptive communication to achieve private goals within the limits of bounded rationality in open argumentation communities. In contrast with our approach, the focus here is not on extending a particular logic for reasoning about coalitions, as in our case. Recent research in formalizing coalition formation has been oriented towards adding more expressivity to Pauly's coalition logic [34]. E.g. in [1], the authors define *Quantified Coalition Logic*, extending coalition logic with a limited but useful form of quantification to express properties such as “*there exists a coalition C satisfying property P such that C can achieve ϕ* ”. In [9], a semantic translation from coalition logic to a fragment of an action logic is defined, connecting the notions of coalition power and the actions of the agents. However, in none of these cases argumentation is used to model the notion of coalition formation as done in this paper.

It must be noted that the adequate formalization of preferences has deserved considerable attention within the argumentation community. In preference-based argumentation theory, an argument may be preferred to another one when, for example, it is more specific, its beliefs have a higher probability or certainty, or it promotes a higher value. Recent work by Kaci *et al.* [29, 30] has provided interesting contributions in this direction, including default reasoning abilities about the preferences over the arguments, as well as an algorithm to derive the most likely preference order.

Future Work. Indeed, in the line with the previous paragraph, one of our future research lines is to extend our current formalization of CoALATL to capture more complex issues in preference handling and to consider more sophisticated semantics.

In the semantics presented in Definition 15 a valid coalition is initially formed and kept until the property is fulfilled. For instance, consider formula $\langle A \rangle \Box \varphi$. The formula is true in q if a valid coalition B in q can be formed such that it can ensure $\Box \varphi$. One might strengthen the scenario and require that B must be valid in *each* state on the path λ satisfying φ . Formally, the semantics could be given as follows: $q \models \langle A \rangle \Box \varphi$ if, and only if, $q \models \varphi$ and there is a coalition $B \in \text{vc}(q, A)$ and a common strategy $s_B \in \Sigma_B$ such that for all paths $\lambda \in \text{out}(q, s_B)$ and for all $i \in \mathbb{N}$ it holds that $\lambda[i] \models \varphi$ and $B \in \text{vc}(\lambda[i], A)$. The

last part specifies that B must be a valid coalition in each state $q_i = \lambda[i]$ of λ .

In the semantics just presented the formed coalition B must persist over time until φ is enforced. One can go one step further. Instead of keeping the same coalition B it can also be sensible to consider “new” valid coalitions in each time step (wrt. A), possibly distinct from B . This leads to some kind of fixed-point definition. At first, B must be a valid coalition in state q leading to a state in which φ is fulfilled and in which another valid coalition (wrt. A and the new state) exists which in turn can ensure to enter a state in which, again, there is another valid coalition and so on.

Finally, we also aim at an actual implementation of a subset of CoALATL in order to perform experiments to assess our proposal when modelling complex problems. We will consider a restriction to some particular argumentation semantics for which proof procedures can be deployed, such as assumption-based argumentation [23]. Research in this direction is currently being pursued. Also the model checking complexity of the goal-based semantics still has to be determined.

8 Conclusions

In this paper we have presented CoALATL , an extension of ATL which is able to model coalition formation through argumentation. Our formalism includes two different modalities, $\langle\langle A \rangle\rangle$ and $\langle\langle A \rangle\rangle$, which refer to different kinds of abilities agents may have. Note that the original operator $\langle\langle A \rangle\rangle$ is used to reason about the pure ability of the very group A . However, the question whether it is reasonable to assume that the members of A collaborate is not taken into account in ATL . With the new operator $\langle\langle A \rangle\rangle$ we try to close this gap, providing also a way to focus on *sensible coalition structures*. In this context, “sensible” refers to *acceptable coalitions with respect to some argumentative semantics* (as characterized in Def. 8).

Furthermore, we have defined the formal machinery required for characterizing argument-based coalition formation in terms of the proposed operator $\langle\langle A \rangle\rangle$. Coalitions can be computed in terms of a given argumentation semantics, which can be given as a parameter within our model, thus providing a modular way of analyzing the results associated with different alternative semantics. This allows us to compare the ability of agents to form particular coalitions and study emerging properties regarding different semantics. Additionally, as outlined in Section 6, the model checking algorithm used in ATL can be extended to CoALATL by integrating suitable proof procedures for argumentation semantics. We have shown that the model checking problem is Δ_2^P -complete/easy for the most natural cases.

Acknowledgements. This research was partially funded by the Projects DAAD-SeCyT (DA0609) and PGI-UNS (24/ZN10).

References

- [1] Thomas Ågotnes, Wiebe van der Hoek, and Michael Wooldridge. Quantified coalition logic. In *IJCAI*, pages 1181–1186, 2007.
- [2] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time Temporal Logic. *Journal of the ACM*, 49:672–713, 2002.
- [3] Leila Amgoud. An argumentation-based model for reasoning about coalition structures. In *ArgMAS*, pages 217–228, 2005.
- [4] Leila Amgoud. Towards a formal model for task allocation via coalition formation. In *AAMAS*, pages 1185–1186, 2005.
- [5] Fahiem Bacchus and Froduald Kabanza. Planning for temporally extended goals. *Ann. Math. Artif. Intell.*, 22(1-2):5–27, 1998.
- [6] Fahiem Bacchus and Froduald Kabanza. Using temporal logics to express search control knowledge for planning. *Artificial Intelligence*, 116(1-2):123–191, 2000.
- [7] Chitta Baral, Vladik Kreinovich, S. Sarkar, and Raúl Trejo. Computational complexity of planning with temporal goals. In *IJCAI*, pages 509–514, 2001.

- [8] Chitta Baral and Jicheng Zhao. Non-monotonic temporal logics for goal specification. In *IJCAI*, pages 236–242, 2007.
- [9] Stefano Borgo. Coalitions in action logic. In *IJCAI*, pages 1822–1827, 2007.
- [10] M.E. Bratman. *Intentions, Plans, and Practical Reason*. Harvard University Press, 1987.
- [11] Ramón F. Brena, José-Luis Aguirre, Carlos Iván Chesñevar, Eduardo H. Ramírez, and Leonardo Garrido. Knowledge and information distribution leveraged by intelligent agents. *Knowl. Inf. Syst.*, 12(2):203–227, 2007.
- [12] N. Bulling, C. Chesñevar, and J. Dix. Modelling coalitions: ATL+argumentation. In *Proceedings of AAMAS’08*, pages 681–688, Estoril, Portugal, May 2008. ACM Press.
- [13] N. Bulling and J. Dix. A finer grained modeling of rational coalitions using goals. In *Proceedings of the 14th Argentinean Conference on Computer Science CACIC’08*, 2008.
- [14] N. Bulling, J. Dix, and C. Chesñevar. An argumentative approach for modelling coalitions using atl. In I. Rahwan and P. Moraitis, editors, *Proc. of the 5rd. Intl. Workshop on Argumentation and Multiagent Systems ArgMAS 2008 (Selected Contributions and Invited Papers)*, Lecture Notes, pages 190–211, Estoril, Portugal, may 2009. Springer.
- [15] Nils Bulling, Wojtek Jamroga, and Jürgen Dix. Reasoning about temporal properties of rational play. *Annals of Mathematics and Artificial Intelligence*, 53(1–4):51–114, 2009.
- [16] M. Caminada. Semi-stable semantics. In *Intl. Conference on Computational Models of Argument (COMMA)*, pages 121–130, 2006.
- [17] Carlos Iván Chesñevar, Ana Gabriela Maguitman, and Ronald Prescott Loui. Logical models of argument. *ACM Comput. Surv.*, 32(4):337–383, 2000.
- [18] E. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 1999.
- [19] Yannis Dimopoulos and Alberto Torres. Graph theoretical structures in logic programs and default theories. *Theor. Comput. Sci.*, 170(1-2):209–244, 1996.
- [20] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.
- [21] Paul E. Dunne and Trevor J. M. Bench-Capon. Two party immediate response disputes: Properties and efficiency. *Artif. Intell.*, 149(2):221–250, 2003.
- [22] Paul E. Dunne and Martin Caminada. Computational complexity of semi-stable semantics in abstract argumentation frameworks. In *JELIA ’08: Proceedings of the 11th European conference on Logics in Artificial Intelligence*, pages 153–165, Berlin, Heidelberg, 2008. Springer-Verlag.
- [23] Dorian Gaertner and Francesca Toni. Computing arguments and attacks in assumption-based argumentation. *IEEE Intelligent Systems*, 22(6):24–33, 2007.
- [24] Hiromitsu Hattori, Takayuki Ito, Tadachika Ozono, and Toramatsu Shintani. An approach to coalition formation using argumentation-based negotiation in multi-agent systems. In *Proceedings of IEA/AIE*, pages 687–696, London, UK, 2001. Springer-Verlag.
- [25] J. Hintikka and G. Sandu. Game-theoretical semantics. In J. van Benthem and A. ter Meulen, editors, *Handbook of Logic and Language*, pages 361–410. Elsevier: Amsterdam, 1997.
- [26] W. Jamroga and N. Bulling. A general framework for reasoning about rational agents. In *Proceedings of AAMAS’07*, pages 592–594, Honolulu, Hawaii, USA, 2007. ACM Press.
- [27] W. Jamroga and N. Bulling. A logic for reasoning about rational agents. In F. Sadri and K. Satoh, editors, *Pre-Proceedings of CLIMA ’07*, pages 54–69, Porto, Portugal, September 2007. Universidade Do Porto.

- [28] W. Jamroga and J. Dix. Model checking abilities of agents: A closer look. *Theory of Computing Systems*, 42(3):366–410, 2008.
- [29] Souhila Kaci and Leendert van der Torre. Preference-based argumentation: Arguments supporting multiple values. *Int. J. Approx. Reasoning*, 48(3):730–751, 2008.
- [30] Souhila Kaci, Leendert W. N. van der Torre, and Emil Weydert. On the acceptability of incompatible arguments. In *Proc. of ECSQARU Conf.*, pages 247–258, 2007.
- [31] Nishan C. Karunatilake, Nicholas R. Jennings, Iyad Rahwan, and Sarvapali D. Ramchurn. Managing social influences through argumentation-based negotiation. In *Proc. of AAMAS*, pages 426–428, 2006.
- [32] F. Laroussinie, N. Markey, and Ph. Schnoebelen. Model checking CTL+ and FCTL is hard. In *Proceedings of FoSSaCS’01*, volume 2030 of *Lecture Notes in Computer Science*, pages 318–331. Springer, 2001.
- [33] C.H. Papadimitriou. *Computational Complexity*. Addison Wesley : Reading, 1994.
- [34] Marc Pauly. A modal logic for coalitional power in games. *J. Log. Comput.*, 12(1):149–166, 2002.
- [35] H. Prakken and G. Vreeswijk. Logical Systems for Defeasible Argumentation. In D. Gabbay and F. Guenther, editors, *Handbook of Phil. Logic*, pages 219–318. Kluwer, 2002.
- [36] Iyad Rahwan and Leila Amgoud. An argumentation based approach for practical reasoning. In *Proc. of AAMAS Conf.*, pages 347–354, 2006.
- [37] Iyad Rahwan, Philippe Pasquier, Liz Sonenberg, and Frank Dignum. On the benefits of exploiting underlying goals in argument-based negotiation. In *Proc. of AAAI Conf.*, pages 116–121, 2007.
- [38] A.S. Rao and M.P. Georgeff. Modeling rational agents within a BDI-architecture. In *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning*, pages 473–484, 1991.
- [39] Michael Rovatsos, Iyad Rahwan, Felix A. Fischer, and Gerhard Weiß. Practical strategic reasoning and adaptation in rational argument-based negotiation. In Simon Parsons, Nicolas Maudet, Pavlos Moraitis, and Iyad Rahwan, editors, *ArgMAS*, volume 4049 of *Lecture Notes in Computer Science*, pages 122–137. Springer, 2005.
- [40] Yuqing Tang and Simon Parsons. Argumentation-based dialogues for deliberation. In *Proc. of AAMAS*, pages 552–559, 2005.
- [41] M. Wooldridge. *An Introduction to Multi Agent Systems*. John Wiley & Sons, 2002.