Resource Action-based Bounded ATL: a New Logic for MAS to Express a Cost Over the Actions

Davide Catta[1][0000-0001-8656-3274], Angelo Ferrando[2][0000-0002-8711-4670]*, and Vadim Malvone[3][0000-0001-6138-4229]

[1] LIPN, CNRS UMR 7030, Université Sorbonne Paris Nord, France
catta@lipn.univ-paris13.fr
[2] University of Modena and Reggio Emilia, Italy
angelo.ferrando@unimore.it
[3] Télécom Paris, Institut Polytechnique de Paris, France
vadim.malvone@telecom-paris.fr

Abstract. In both human society and Multi-Agent Systems (MAS), actions entail costs due to resource limitations such as energy consumption and communication bandwidth. Consideration of these constraints is crucial during MAS design and implementation, especially regarding agents' ability to achieve temporal objectives. Resource Bounded ATL (RB-ATL) extends ATL to accommodate resource limitations but struggles to isolate costs to individual actions. Agents' actions can be influenced by others, affecting cooperation or competition. To address these complexities, we introduce Resource Action-based Bounded ATL (RAB-ATL), which considers actions' costs in relation to other agents' actions within the same state. RAB-ATL enhances understanding and introduces strategic considerations at the resource handling level, offering a more comprehensive approach to agent interaction. Additionally, we analyse the model checking complexity for RAB-ATL and show that it remains consistent with that of RB-ATL. Finally, we present a resulting implementation of the technique and its application to an existing case study.

Keywords: Strategic Verification · Resource Bounded ATL · Model Checking

1 Introduction

The proverb "nothing comes free in life" holds true whether we consider human beings in society or software agents in MAS. Actions incur costs, including the pursuit of personal goals. An agent, particularly when situated within an environment, may encounter resource limitations such as the energy required to act on an object, the bandwidth needed for communication with other agents, or the computational time necessary for internal processes. Therefore, the notion of an ideal system where agents exist without cost is merely an illusion and does not align with reality in today's world.

This intuition must be considered at both the design and implementation stages of a MAS. Specifically, when verifying whether the agents in the MAS can achieve their goals, the concept of available resources for the agents to utilise is of paramount importance. In this context, works have been carried out to pursue a resource-bound verification of MAS, particularly leveraging Alternating-time Temporal Logic (ATL) [5] as a formalism to denote the agents' temporal objectives. This has led to the proposition of Resource Bounded ATL (RB-ATL) [2], an extension of ATL where resource limitations are taken into consideration. Such logic allows for constraining the actions performed by the agents in terms of their own costs. However, RB-ATL presents a limitation in terms of how such costs are assigned to the agents' actions. In RB-ATL, the cost of an action is deterministically determined by only the agent performing the action and the state in which the action has been performed. This limitation restricts the possible uses of RB-ATL to only scenarios where the cost of actions is considered in complete isolation (*i.e.*, each action is assumed to not be influenced by other actions performed concurrently). However, in general, an agent's action can be influenced by another agent's action. For example, consider the action of pulling a lever. To perform such an action, it is reasonable to assume the agent needs to consume a certain amount of energy. Nonetheless, if another agent were also pulling the lever along with the agent, then the energy required to complete the action would not necessarily be the same. Indeed, with two agents pulling the lever, the energy required by each agent would hypothetically decrease (perhaps by half). Note that this aspect is not limited solely to agents' cooperation but also extends to agents' competition. For example, revisiting the lever scenario, if an enemy agent were to push the lever while the agent is pulling it, this action would result in the agent consuming more energy. Such scenarios cannot be natively supported in RB-ATL.

To overcome this limitation, we present Resource Action-based Bounded ATL (RAB-ATL), a variant of RB-ATL where the actions' cost is not associated only with the agent performing the action and the state where the action is performed, but also with the other actions performed by other agents in the same state. It is worth noting that the difference between these logics is not only conceptual but also foundational. This distinction arises from the fact that RAB-ATL introduces the strategic aspect at the resource handling level. Unlike RB-ATL, in RAB-ATL, the cost of actions is not considered in isolation, allowing for the full exploitation of cooperation (and competition) amongst agents. From this perspective, we consider RAB-ATL as the natural extension of ATL towards resource handling.

Last, but not least, we emphasise that this is the first work to address both theoretical and practical aspects in the formal verification of MAS with resource bounds. Not only we propose and study RAB-ATL (particularly in comparison with RB-ATL), but we also present an implementation that tackles the resulting model checking problem as a component of the `VITAMIN` framework [17,18]. `VITAMIN` is an open-source model checker designed for verifying MAS and supports a variety of specifications, including Alternating-time Temporal Logic (ATL) [5], Natural ATL (NatATL) [19], Natural SL [8], ATL with Fuzzy functions (ATLF) [16], Resource-Bounded ATL (RB-ATL) [21], Capacity ATL (CapATL) [6], Obstruction Logic (OL) [11], and Obstruction ATL (OATL) [12].

The paper is structured as follows. Section 2 provides the theoretical basis necessary for understanding the contribution. Section 3 revisits RB-ATL syntax and semantics, and shows that the memoryful and the memoryless semantics do not coincide for this logic. Section 4 introduces the new logic RAB-ATL and its model checking problem. Section 5 outlines the implementation approach and reports on the experimental evaluation. Finally, Section 6 positions the paper with respect to the state of the art, while Section 7 concludes the paper and highlights future directions.

2 Preliminaries

Let us fix some notation and terminology that will be used in the following. If X is a set and $Y \subseteq X$, we denote by $\overline{Y}$ the complementary set $X \setminus Y$ of Y in X. If π is a sequence, we denote by $|\pi|$ its length and, given $i \leq |\pi|$, we let π_i denote the i-th element of π, $\pi_{\leq i}$ the prefix $\pi_1, \ldots, \pi_i$ of π and $\pi_{\geq i}$ the suffix of π starting at π_i. If π is finite, then $last(\pi)$ denote its last element $\pi_{|\pi|}$. If $\alpha = \langle x_1, \ldots, x_n \rangle$ is a tuple, then $\alpha[i]$ denotes its i-th component x_i.

Definition 1. *A Concurrent Game Structure (CGS for short) is a tuple* $\mathfrak{G} = \langle Ap, Ag, S, s_I, \{act_i\}_{i \in Ag}, P, t, L \rangle$ *such that:*

- Ap *is a non-empty set of atomic propositions;*
- $Ag = \{1, \ldots, n\}$ *is a finite set of agents;*
- S *is a non-empty set of states and* $s_I \in S$ *is the initial state;*
- *for any* $i \in Ag$*,* act_i *is a set of actions,* $ACT = \Pi_{i \in Ag} act_i$ *is the set of tuples of actions, and* $act = \bigcup_{i \in Ag} act_i$ *the set of all actions;*
- $P : Ag \times S \rightarrow (2^{act} \setminus \emptyset)$ *is the protocol function that associates to any agent i and state s a non-empty subset of* act_i *representing the actions that are available for i at s. We impose that the idle action* $\star$ *always belong to* $P(i, s)$ *for any i;*
- $t : S \times ACT \rightarrow S$ *is the transition function, that is given a state s and a tuple of actions* $\mathbf{a}$ *(where* $\forall i$*,* $\mathbf{a}[i] \in P(i, s)$*) such function outputs a state* s'*;*
- $L : S \rightarrow 2^{Ap}$ *is the labeling function associating to any state s a set of atomic propositions; such set can be empty and represents the set of proposition that are true at s.*

If C is a coalition and s is a state a **C-action** available at s is a tuple α whose length is $|Ag|$ and such that $\alpha[i] \in P(s, i)$ for each $i \in C$ and for each $j \in \overline{C}$, $\alpha[j] = \#_j$, where $\#_j$ is a fixed symbol used as placeholder for an arbitrary action of player j. We denote by $Act(C, s)$ the set of all C-actions at s. If $\alpha \in Act(C, s)$ and $\beta \in Act(\overline{C}, s)$ then $\alpha \cdot \beta$ denotes the unique joint action $\mathbf{a} \in Act(Ag, s)$ such that $\mathbf{a}[i] = \alpha[i]$ for each $i \in C$ and $\mathbf{a}[j] = \beta[j]$ for each $j \in \overline{C}$. We denote by $Act(C, S)$ the set $\bigcup_{s \in S} Act(C, s)$. If $\alpha \in Act(C, s)$ then we denote by $\alpha_s^{\preceq}$ the set of joint actions extending α at s, that is $\alpha_s^{\preceq} = \{\mathbf{a} \in Act(Ag, s) \mid \mathbf{a} = \alpha \cdot \beta \text{ for some } \beta \in Act(\overline{C}, s)\}$. Given $\alpha \in Act(C, s)$, we denote by $Post(s, \alpha)$ the set of states that the coalition C can reach by playing the action α, that is: $Post(s, \alpha) = \{s' \in S \mid \exists\, \mathbf{a} \in ACT \text{ s.t. } t(s, \mathbf{a}) = s' \text{ and } \mathbf{a} = \alpha \cdot \beta \text{ for } \beta \in Act(\overline{C}, s)\}$.

A **path** ρ is an infinite alternated sequence $s_1, \mathbf{a}_1, s_2, \ldots$ of states and tuples in ACT such that for all $i \geq 1$, $t(s_i, \mathbf{a}_i) = s_{i+1}$. If ρ is a path, we denote by ρ^S the sub-sequence of ρ only containing states. If $h \in S^+$ is a finite sequence of states, we say that h is a **history** iff there is a path ρ such that $h = \rho^S_{\leq i}$ for some $i \in \mathbb{N}$. We use H to denote the set of all histories.

3 Resource Bounded ATL

In many MAS, agents are resource-bounded, in the sense that they require resources in order to act. To formalise such notion, in [2] the authors introduced Resource Bounded ATL (RB-ATL for short). RB-ATL is a variant of ATL in which strategic formulae are decorated with bound, *i.e.*, natural numbers vectors of finite size. The intended meaning of a formula $\langle\langle C^{\mathbf{b}} \rangle\rangle \psi$ of RB-ATL can be expressed as *the coalition of agents C has a strategy to achieve the objective ψ whose cost does not surpass* $\mathbf{b}$. We now recall the syntax and semantics of RB-ATL.

Definition 2. *Formulae of RB-ATL are defined by the following grammar:*

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle\langle C^{\mathbf{b}} \rangle\rangle \mathsf{X}\, \varphi \mid \langle\langle C^{\mathbf{b}} \rangle\rangle \mathsf{G}\, \varphi \mid \langle\langle C^{\mathbf{b}} \rangle\rangle \varphi \,\mathsf{U}\, \varphi$$

where $p \in Ap$, $C \subseteq Ag$, and $\mathbf{b}$ is any bound. We can derive the boolean connectives $\top$, $\bot$, $\vee$, and $\rightarrow$ as usual. We define $\langle\langle C^{\mathbf{b}} \rangle\rangle\ \mathsf{F}\, \varphi$ as $\langle\langle C^{\mathbf{b}} \rangle\rangle \top \mathsf{U}\, \varphi$. We will use φ, ψ, θ, etc., to denote arbitrary formulae.

Formulae of RB-ATL are interpreted over RB-CGSs. These are CGSs in which a cost (a natural number vector) is associated to any agent action at any state. The formal definition follows.

Definition 3. *A Resource Bounded CGS (RB-CGS for short) is a triple $\mathfrak{M} = \langle \mathfrak{G}, r, \mathsf{C} \rangle$ where:*

- *$\mathfrak{G}$ is a CGS as in Definition 1;*
- *$r \geq 1$ is a natural number (the number of resources types);*
- *$\mathsf{C} : S \times act \rightarrow \mathbb{N}^r$ is a function associating to any state s and action a, a cost $\mathsf{C}(s, a)$, that is a vector in $\mathbb{N}^r$.*

As in [2] we impose that any agent at any state has at its disposal the idle action $\star$ and that the cost of such action is always $\mathbf{0}$.

Example 1. To better present our contribution, we report a motivating example, initially introduced in [22]. This example concerns a sensor network depicted in Figure 1 and comprising two agents, labelled as 1 and 2, tasked with monitoring movement. Should either agent detect movement, they can notify their counterpart sensor. Upon receiving such a communication, an agent can store it. If an agent accumulates multiple movement records, it can relay this information to the central base station. Notably, we assume that agent 2 is situated closer to the base station than agent 1. In this setup, two vital resources come into play: energy and memory. The act of message transmission consumes energy, contingent upon the distance to the recipient, while storing communications requires memory. Specifically, sending messages from 1 to 2 ($send_{12}$) and from 2 to 1 ($send_{21}$) both necessitate 2 units of energy and 0 memory. Saving a record demands 0 units of energy but incurs a cost of 1 unit of memory. Transmitting data from 1 to the base station ($send_{1b}$) requires 3 units of energy, whereas sending from 2 to the base ($send_{2b}$) demands 1 unit of energy. Alternatively, the option to remain

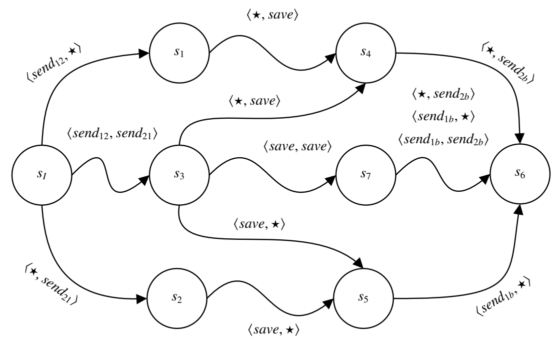


Fig. 1. The RB-CGS $\mathfrak{M}$ for the sensor network example as in [22].

idle is perpetually available at no cost. Initially, each agent holds a record of detecting movement.

Figure 1 depicts the system, with transitions between states annotated with action tuples, denoting the actions of the agents 1 and 2, respectively. For clarity, we exclude self-loops in each state, represented by the joint action $\langle \star, \star \rangle$.

Strategies are defined in the standard way as follows.

Definition 4. *A memoryful strategy for a coalition of agents C is a function $\sigma_C : H \rightarrow Act(C, S)$ mapping a history h to a joint action $\alpha \in Act(C, last(h))$. A strategy for a coalition C is said to be memoryless whenever $\sigma_C(h) = \sigma_C(h')$ if $last(h) = last(h')$.*

A path $\rho = s_1, \mathbf{a_1}, s_2 \ldots$ is **compatible** with a joint strategy σ_C if for every $i \geq 1$ and every $k \in C$ it holds that $\sigma_C(\rho^S_{\leq i})[k] = \mathbf{a}_i[k]$. We denote with $out(s, \sigma_C)$ the set of all σ_C-compatible paths whose first element is s.

Let s be a state and $\alpha \in Act(C, s)$. The cost of α at s is given by:

$$cost(s, \alpha) = \sum_{i \in C} C(s, \alpha[i])$$

Let σ_C be a strategy for the coalition C, $\rho = s_1, \mathbf{a}_1, s_2, \ldots$ be a path in $out(s, \sigma_C)$, and $\mathbf{b} \in \mathbb{N}^r$. We say that ρ is $\mathbf{b}$-consistent when for each natural number $n \geq 1$, we have that:

$$\sum_{k=1}^{n} cost(s_k, \sigma_C(\rho^S_{\leq k})) \leq \mathbf{b}$$

A strategy σ_C for a coalition C is $\mathbf{b}$-consistent whenever, for every state s, given any $\rho \in out(s, \sigma_C)$, ρ is $\mathbf{b}$-consistent.

We now define the semantic interpretation of RB-ATL formulae.

Definition 5. *Given a RB-CGS $\mathfrak{M}$, a state s of $\mathfrak{M}$, and a formula φ, the satisfaction relation $\mathfrak{M}, s \models \varphi$ is inductively defined on the structure of φ as follows:*

- $\mathfrak{M}, s \models \top$ *always;*

- *$\mathfrak{M}, s \models p$ iff $p \in L(s)$;*
- *$\mathfrak{M}, s \models \neg\psi$ iff it is not the case that $\mathfrak{M}, s \models \psi$ (denoted $\mathfrak{M}, s \not\models \psi$);*
- *$\mathfrak{M}, s \models \psi \wedge \theta$ iff $\mathfrak{M}, s \models \psi$ and $\mathfrak{M}, s \models \theta$;*
- *$\mathfrak{M}, s \models \langle\langle C^{\mathbf{b}} \rangle\rangle \mathsf{X}\, \psi$ iff there is a **b**-consistent strategy σ_C for the coalition C such that for every path $\rho \in out(s, \sigma_C)$ we have that $\mathfrak{M}, \rho_2^S \models \psi$;*
- *$\mathfrak{M}, s \models \langle\langle C^{\mathbf{b}} \rangle\rangle \mathsf{G}\, \psi$ iff there is a **b**-consistent strategy σ_C for the coalition C such that for every path $\rho \in out(s, \sigma_C)$ and for every $i \geq 1$, we have that $\mathfrak{M}, \rho_i^S \models \psi$;*
- *$\mathfrak{M}, s \models \langle\langle C^{\mathbf{b}} \rangle\rangle \psi \,\mathsf{U}\, \theta$ iff there is a **b**-consistent strategy σ_C for the coalition C such that for every path $\rho \in out(s, \sigma_C)$ there exists a $j \geq 1$ such that $\mathfrak{M}, \rho_j^S \models \theta$ and for every $1 \leq i < j$ we have that $\mathfrak{M}, \rho_i^S \models \psi$.*

We write $\mathfrak{M} \models \varphi$ and we say that $\mathfrak{M}$ satisfies φ iff $\mathfrak{M}, s_I \models \varphi$.

The memoryless satisfaction relation $\models_r$ is obtained by substituting, above, every occurrence of strategy with memoryless strategy.

If $\mathfrak{M}$ is a RB-CGS and φ is a formula, we denote by $[\![\varphi]\!]^{\mathfrak{M}}$ the set of states of $\mathfrak{M}$ that satisfies φ, that is $[\![\varphi]\!]^{\mathfrak{M}} = \{s \in S \mid \mathfrak{M}, s \models \varphi\}$. In what follows, we may omit the superscript $\mathfrak{M}$ from $[\![\varphi]\!]^{\mathfrak{M}}$ when $\mathfrak{M}$ is contextually given. In [2] the authors claim that the model-checking problem for RB-ATL can be solved in time that is linear on the size of the model and exponential in the number r of resource bounds. We report the exact statement of the Theorem.

Theorem 1 ([2]). *Given a finite RB-CGS $\mathfrak{M}$ and a formula φ, there is an algorithm that computes $[\![\varphi]\!]^{\mathfrak{M}}$ which runs in $\mathbf{O}(|\mathfrak{M}| \times |\varphi|^{2r+1})$ where $|\mathfrak{M}|$ is the size of the RB-CGS and r is the number of resources.*

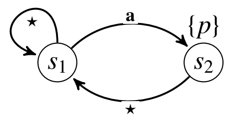
Example 2. Considering the motivating example whose RB-CGS is reported in Figure 1, both agents collaboratively achieve the desired outcome s_6, indicating that the base station has been informed. This is expressed in RB-ATL as $\varphi = \langle\langle \{1,2\}^{\langle \mathbf{b}_1, \mathbf{b}_2 \rangle} \rangle\rangle \top \,\mathsf{U}\, s_6$, where $\mathbf{b}_1$ and $\mathbf{b}_2$ are the chosen bounds for the energy and memory resources, respectively. Note that, we use the atomic proposition s_6 by assuming that in each state s_i the atomic proposition s_i holds. This property can be satisfied by assuming a cost of 3 units of energy and 1 unit of memory, that is $\mathbf{b}_1 = 3$ and $\mathbf{b}_2 = 1$. For example, through the sequence of actions: $\langle send_{12}, \star \rangle$ in s_I, $\langle \star, save \rangle$ in s_1, and $\langle \star, send_{2b} \rangle$ in s_4. Notice that, the same property but assuming cost of 2 units of energy and 1 unit of memory, that is $\mathbf{b}_1 = 2$ and $\mathbf{b}_2 = 1$, is not instead satisfied in $\mathfrak{M}$.

3.1 Comparison between memoryless and memoryful semantics for RB-ATL

We here show that, contrarily to what happens in ATL, for RB-ATL the set of formulae that can be enforced by a coalition C in a given RB-CGS using memoryless strategies and the set of formulae that can be enforced by C using memoryful strategies do not coincide. Later on the paper, we will use this result to show that the same property is enjoyed by our logic RAB-ATL.

Theorem 2. *The memoryless $\models_r$ and memoryful $\models$ satisfaction relations do not coincide on RB-CGSs: there is a RB-CGS $\mathfrak{M}$, a RB-ATL formula φ and a state s of $\mathfrak{M}$ such that $\mathfrak{M}, s \models \varphi$ and $\mathfrak{M}, s \not\models_r \varphi$.*

Proof. Consider the 1-agent RB-CGS depicted below, where $C(s_1, \mathbf{a}) = 1$ and the cost of the idle action is 0 as specified in Definition 6.



Consider the formula $\varphi = \langle\langle a^1 \rangle\rangle \, \mathsf{X} \, p$ where a is the only agent of the RB-CGS. The formula φ is satisfied at s_1 if a uses memoryful strategies. In fact, consider the strategy σ_a choosing $\mathbf{a}$ on h if $h = s_1$ and the idle action $\star$ otherwise. The only path in $out(s_1, \sigma_a)$ is $s_1 \cdot s_2 \cdot s_1^{\omega}$, this path is 1-compatible and s_2 satisfies p. On the contrary, it is easy to see that there is no memoryless strategies that realize the same formula starting at s_1 that is 1-compatible.

4 Resource Action-based Bounded ATL

As we have previously said, in many scenarios, agents are resource-bounded: they dispose of a limited amount of resources in order to perform their actions. We saw that to reason about the strategic objectives of resource bounded agents, one can use RB-ATL. However, in RB-ATL it is not possible to model, at least naturally, situations where the cost of an action in a particular state depends on the actions of other agents in the same state. To clarify this concept, let us propose a very simple example in which we have three actors Alice, Bob, and Carl. We assume that Carl's car has broken down, and to restart it, some of them need to push it for a while. Alice and Bob take charge of pushing the car. The cost, in terms of physical effort, of this coordinated action for both depends on both the action of the other and the action taken by Carl: if Carl forgets to release the handbrake, the pushing action will have a much higher cost.

To reason about these types of scenarios, we introduce a variant of RB-ATL which we call, for lack of wit, Resource Action-Based Bounded ATL. The syntax of this logic is exactly the same as that of RB-ATL: strategic formulae are decorated with a bound, and such formulae have the same intuitive meaning as the strategic formulae of RB-ATL: $\langle\langle C^{\mathbf{b}} \rangle\rangle \, \psi$ means that the coalition C possesses a strategy to achieve ψ while spending at most $\mathbf{b}$ resources. However, the way the cost of an action for an agent is computed in RAB-ATL is different. Models of RAB-ATL will be CGS where a cost is associated with triples $\langle s, a, \mathbf{a} \rangle$ where s is a state, a is an action, and $\mathbf{a}$ is a collective action of which a is one of the components. This formalizes the intuition that the cost of an action in a state depends on what other agents do in the same state. We now formally define models of RAB-ATL.

Definition 6. *A Resource Action-Based Bounded CGS (RAB-CGS) is a tuple* $\mathfrak{A} = \langle \mathfrak{G}, r, \$ \rangle$ *where:*

- $\mathfrak{G}$ *is a CGS as in Definition 1;*
- $r \geq 1$ *is a natural number (the number of resources types);*
- $\$: S \times act \times ACT \to \mathbb{N}^r$ *is a function mapping a state s, an action a, and a tuple of actions* $\mathbf{a} = \langle a_1, \ldots, a_n \rangle$ *to a natural number vector of length r. We impose that a is one of the* a_i *composing* $\mathbf{a}$*, and impose that* $\$(s, \star, \mathbf{a}) = \mathbf{0}$ *for any* $\mathbf{a}$.

Example 3. We can revise the model $\mathfrak{M}$ of Figure 1 as a RAB-CGS $\mathfrak{A}$. In particular, we can adjust the costs associated with the actions by indicating that in state s_I, if both agents 1 and 2 exchange data by selecting the tuple $\langle send_{12}, send_{21}\rangle$, the cost of each action is only 0.5 energy (resulting in a total cost of 1). This adjustment reflects the use of the TCP protocol, which allows for the addition of an acknowledgement message along with a standard packet to be sent (a concept known as Piggybacking). In this scenario, the act of duplicating the sharing of data ensures a more reliable and robust communication amongst the agents.

In what follows, a bound $\mathbf{b}$ will be any element of $\mathbb{N}^r$. Let s be a state, $\alpha \in Act(C, s)$ and $\mathbf{a} = \langle a_1, \dots, a_{\#(Ag)}\rangle \in Act(Ag, s)$ be a tuple of actions extending α. The cost of α at s with respect to $\mathbf{a}$ is given by:

$$cost(s, \alpha, \mathbf{a}) = \sum_{i \in C} \$(s, \alpha[i], \mathbf{a})$$

Let σ_C be a strategy for the coalition C, $\rho = s_1, \mathbf{a}_1, s_2, \dots$ be a path in $out(s, \sigma_C)$, and $\mathbf{b} \in \mathbb{N}^r$. We say that ρ is action $\mathbf{b}$-consistent (action-bound consistent) when for each natural number $n \geq 1$, we have that:

$$\sum_{k=1}^{n} cost(s_k, \sigma_C(\rho^S_{\leq k}), \mathbf{a}_k) \leq \mathbf{b}$$

A strategy σ_C for a coalition C is action $\mathbf{b}$-consistent whenever, for every state s, given any $\rho \in out(s, \sigma_C)$, ρ is $\mathbf{b}$-consistent.

Definition 7. *Formulae of RAB-ATL are defined exactly as in Definition 2. Given a RAB-CGS $\mathfrak{A}$, a state s and a formula φ, the satisfaction relation $\mathfrak{A}, s \models^A \varphi$ is defined exactly as in Definition 5, the only difference being that strategies must be action-bound consistent, and likewise for the memoryless satisfaction relation $\mathfrak{A}, s \models^A_r \varphi$. n*

If $\mathfrak{A}$ is a RAB-CGS and φ is a formula, we denote by $[\![\varphi]\!]^{\mathfrak{A}}$ the set of states of $\mathfrak{A}$ that satisfies φ, that is $[\![\varphi]\!]^{\mathfrak{A}} = \{s \in S \mid \mathfrak{A}, s \models^A \varphi\}$. In what follows, we may omit the superscript $\mathfrak{A}$ from $[\![\varphi]\!]^{\mathfrak{A}}$ when $\mathfrak{A}$ is contextually given.

We now show that RAB-ATL can simulate RB-ATL. More precisely, given a RB-CGS $\mathfrak{M} = \langle \mathfrak{G}, r, C\rangle$ one can construct a RAB-CGS $\mathfrak{A} = \langle \mathfrak{G}, r, \$\rangle$ such that for any formula φ we have that $[\![\varphi]\!]^{\mathfrak{M}} = [\![\varphi]\!]^{\mathfrak{A}}$. The proof of this result is particularly straightforward.

Proposition 1. *If $\mathfrak{M}$ is a RB-CGS then there is a RAB-CGS $\mathfrak{A}$ whose size is equal to those of $\mathfrak{M}$ and such that, for any formula φ we have that $\mathfrak{M} \models \varphi$ iff $\mathfrak{A} \models^A \varphi$.*

Proof. Given $\mathfrak{M} = \langle \mathfrak{G}, r, C\rangle$ consider the RAB-CGS $\mathfrak{A}^{\mathfrak{M}} = \langle \mathfrak{G}, r, \$^C\rangle$ where for any state s, action a, and joint action $\mathbf{a}$ we have that $\$^C(s, a, \mathbf{a}) = C(s, a)$. It is immediate to show that $\mathfrak{M} \models \varphi$ iff $\mathfrak{A} \models^A \varphi$ by induction on the structure of φ.

Because of the above proposition and by Theorem 2, we obtain the following.

Corollary 1. *The memoryless $\models_r^A$ and memoryful $\models^A$ satisfaction relations do not coincide on RAB-CGS: there is a RAB-CGS $\mathfrak{A}$, a RAB-ATL formula φ and a state s of $\mathfrak{A}$ such that $\mathfrak{A}, s \models^A \varphi$ and $\mathfrak{A}, s \not\models_r^A \varphi$.*

Example 4. Given the above relation and the RB-CGS $\mathfrak{M}$ of Example 1, we can construct an equivalent RAB-CGS $\mathfrak{A}^{\mathfrak{M}}$. In the latter model, as consequence, the formula φ defined in Example 2 holds for $\mathbf{b}_1 = 3$ and $\mathbf{b}_2 = 1$ but not for $\mathbf{b}_1 = 2$ and $\mathbf{b}_2 = 1$. However, by assuming the RAB-CGS $\mathfrak{A}$ of the Example 3 the formula φ still holds for $\mathbf{b}_1 = 3$ and $\mathbf{b}_2 = 1$ but also for $\mathbf{b}_1 = 2$ and $\mathbf{b}_2 = 1$.

4.1 Model checking

We now prove that the model-checking problem for RAB-ATL is decidable, and it is in the same complexity class of RB-ATL.

If C is a coalition, X is a set of states and $\mathbf{b}$ is a bound, we denote by $Pre(C, X, \mathbf{b})$ the set of states from which there is a C-action α such that C can reach, by executing α, only states in X with cost at most $\mathbf{b}$, that is:

$$Pre(C, X, \mathbf{b}) = \{s \in S \mid \exists \alpha \in Act(C, s), Post(s, \alpha) \subseteq X,$$

$$cost(s, \alpha, \mathbf{c}) \leq \mathbf{b} \text{ for each } \mathbf{c} \in \alpha_s^{\preceq}\}$$

We can prove that, given a formula φ, the set of states $Pre(C, [\![\varphi]\!], \mathbf{b})$ corresponds to the set of states that satisfies $\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X}\, \varphi$, and that $Pre(C, X, \mathbf{b})$ can be computed in quadratic time.

Proposition 2. *If $\mathfrak{A}$ is a RAB-CGS and φ is a formula, then $[\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X}\, \varphi]\!] = Pre(C, [\![\varphi]\!], \mathbf{b})$. Moreover calculating $Pre(C, [\![\varphi]\!], \mathbf{b})$ takes polynomial time in the size of $\mathfrak{A}$.*

Proof. We first prove that $[\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X}\, \varphi]\!] = Pre(C, [\![\varphi]\!], \mathbf{b})$. For the left-to-right direction, suppose that $s \in [\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X}\, \varphi]\!]$, this means that there is an action $\mathbf{b}$-consistent strategy σ_C such that for every path $\rho \in out(s, \sigma_C)$ we have that $\rho_2^S \in [\![\varphi]\!]$. Let α be $\sigma_C(s)$. Since σ_C is $\mathbf{b}$-consistent, we must have that $cost(s, \alpha, \mathbf{a}) \leq \mathbf{b}$ for any $\mathbf{a}$ such that $t(s, \mathbf{a}) = s' = \rho_2^S$ for some $\rho \in out(s, \sigma_C)$. From this, and the fact that $\rho_2^S \in [\![\varphi]\!]$ the result follows. For the other direction, suppose that $s \in Pre(C, [\![\varphi]\!], \mathbf{b})$, thus there is a joint action $\alpha \in Act(C, s)$ such that $Post(\alpha, s) \subseteq [\![\varphi]\!]$ and $cost(s, \alpha, \mathbf{c}) \leq \mathbf{b}$ for any $\mathbf{c}$ extending α. Let σ_C the joint strategy for C such that $\sigma_C(s) = \alpha$ and $\sigma_C(h) = \beta$ for any $h \neq s$, where β is the C joint action such that $\beta[i] = \star$ for any $i \in C$. By hypothesis, $cost(s, \sigma_C(s), \mathbf{c}) \leq \mathbf{b}$ for any $\mathbf{c}$ extending α, $Post(s, \sigma_C(s)) \subseteq [\![\varphi]\!]$ and, by Definition 6, $cost(\pi_i^S, \sigma_C(\pi_i^S), \mathbf{c}) = 0$ for any $i \geq 2$ and $\pi \in out(s, \sigma_C)$. We thus conclude that σ_C is $\mathbf{b}$-consistent and, since $\pi_2^S \in [\![\varphi]\!]$, we have that $\mathfrak{A}, s \models \langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \varphi$.

Algorithm 1 gives a procedure to calculate $Pre(C, X, \mathbf{b})$ given a model $\mathfrak{A}$. To obtain the desired complexity result, it is sufficient to remark that the three algorithm loops are bound by the cardinality of the transition function of $\mathfrak{A}$.

Note that one can calculate the *Pre* function of RB-ATL by omitting the loop at line 6 of our algorithm and using the cost function of RB-ATL. This implies that, while staying in the same complexity class, RAB-ATL *Pre* function takes quadratic time in

Algorithm 1 $Pre(C, X, \mathbf{b})$

```
Y = ∅
for s ∈ S do
    for α ∈ Act(C, s) do
        if Post(α, s) ⊆ X then
            bool = true
            for c ∈ Act(Ag, s) do
                if Cost(s, α, c) > b then
                    bool = false
                    Break
            if bool = true then
                Y = Y ∪ {s}
Return Y
```

the cardinality of the transition function, while the one of RB-ATL is linear w.r.t. the same parameter [1,2,3,4].

We now prove that the set semantics of strategic operators of RAB-ATL is the fix-point of particular monotone functions over the powerset of the set of states of a RAB-CGS. The proof will follow the analogous one in [2]. Let $\mathfrak{A}$ be a RAB-CGS, C a coalition, $\mathbf{b}$ a bound, and $\mathbf{0}$ the vector in $\mathbb{N}^r$ in which each component is 0. Consider the two monotone functions from 2^S to 2^S defined as follows:

$$\mathsf{G}^{\mathbf{b}}_{C,\varphi}(X) = [\![\varphi]\!] \cap ([\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X} \langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{G}\, \varphi]\!] \cup Pre(C, X, \mathbf{0}))$$

$$\mathsf{U}^{\mathbf{b}}_{C,\varphi,\psi}(X) = [\![\psi]\!] \cup (([\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X} \langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \varphi \,\mathsf{U}\, \psi]\!]) \cup Pre(C, X, \mathbf{0}))$$

One can show the following.

Lemma 1. *For every model $\mathfrak{M}$, coalition C, bound $\mathbf{b}$, and pair of formulae φ and ψ:*

1. $[\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{G}\, \varphi]\!]$ *is the least fix-point of the function* $\mathsf{G}^{\mathbf{b}}_{C,\varphi}$*;*
2. $[\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \varphi \,\mathsf{U}\, \psi]\!]$ *is the greatest fix-point of the function* $\mathsf{U}^{\mathbf{b}}_{C,\varphi,\psi}$*.*

Proof. We give a detailed proof of 1. The proof follows the one presented in [2]. We first show that $Y \subseteq \mathsf{G}^{\mathbf{b}}_{C,\varphi}(Y)$ when $Y = [\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{G}\, \varphi]\!]$. Let $s \in Y$, by the definition of satisfaction, there is an action $\mathbf{b}$-consistent strategy σ_C such that $\pi^S_i \in [\![\varphi]\!]$ for any path $\pi \in out(s, \sigma_C)$. In particular, this means that $s \in [\![\varphi]\!]$. Let $\mathbf{b}' = max\{cost(s, \sigma_C(s), \mathbf{c}) \mid \mathbf{c} \in \sigma_C(s)^{\preceq}_s\}$. We define a $\mathbf{b} - \mathbf{b}'$ strategy σ'_C as follows: $\sigma'_C(h) = \sigma_C(s \cdot h)$ for any history h that is a prefix of some path $\pi_{\geq 2}$ such that $\pi \in out(s, \sigma_C)$, and $\sigma'_C(h) = \sigma_C(h)$ for any history h that does not have the above-mentioned characteristics. Clearly, for any path $\rho \in out(\pi_2, \sigma'_C)$ we have that $\rho^S_i \in [\![\varphi]\!]$. Moreover, any of those paths is action $\mathbf{b} - \mathbf{b}'$-consistent. It follows that for any $\pi \in out(s, \sigma_C)$, $\mathfrak{M}, \pi_2 \models^A \langle\!\langle C^{\mathbf{b}-\mathbf{b}'} \rangle\!\rangle \mathsf{G}\, \varphi$. Thus, if $\mathbf{b}' \neq 0$, we obtain that $s \in [\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X} \langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{G}\, \varphi]\!]$, otherwise $s \in Pre(C, [\![\langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \varphi]\!], \mathbf{0})$ as desired.

For the other direction, let Z be a post-fixed point of $\mathsf{G}^{\mathbf{b}}_{C,\varphi}(X)$; we show that $Z \subseteq Y$. The proof proceeds by induction on the bound $\mathbf{b}$.

Base case $\mathbf{b} = \mathbf{0}$. We have that $\mathsf{G}^{\mathbf{b}}_{C,\varphi}(X) = [\![\varphi]\!] \cap Pre(C, X, \mathbf{0})$. Assume that $s \in Z$, thus $s \in \mathsf{G}^{\mathbf{b}}_{C,\varphi}(Z)$ as Z is a post-fixed point of $\mathsf{G}^{\mathbf{b}}_{C,\varphi}(X)$. We define a strategy σ_C that is action $\mathbf{0}$-consistent and in which each element of each path belongs to Z. Formally, for any

$n \in \mathbb{N}$, and for any history $h \in H^n$ of length n, we choose a joint action α that can be played at $last(h)$. By choosing the appropriate α, we make sure that each successor of $last(h)$ selected according to α is in Z and thus in $\mathsf{G}^{\mathbf{b}}_{C,\varphi}(Z)$.

1. if $n = 1$, then by hypothesis $H^n_s = \{s\}$ and $s \in [\![\varphi]\!]$. Since $s \in Z$, there is a joint action α such that $cost(s, \alpha, \mathbf{c}) = \mathbf{0}$ for any $\mathbf{c}$ extending α, and $s' \in Z$ for any $s' \in Post(\alpha, s) \subseteq Z \subseteq \mathsf{G}^{\mathbf{b}}_{C,\varphi}(Z)$.
2. For $n > 1$, we have that for any member of $h \in H^n$, and for any $i \leq n$, $h_i \in Z$. Since Z is a fixed-point of $\mathsf{G}^{\mathbf{b}}_{C,\varphi}$, we can choose an appropriate joint action as in the previous case and obtain the wanted result.

Induction step $\mathbf{b} > \mathbf{0}$. We have that $\mathsf{G}^{\mathbf{b}}_{C,\varphi}(Z) = [\![\varphi]\!] \cap ([\![\langle\!\langle C^{\mathbf{b}}\rangle\!\rangle \mathsf{X} \langle\!\langle C^{\mathbf{b}}\rangle\!\rangle \mathsf{G}\, \varphi]\!] \cup Pre(C, Z, \mathbf{0}))$. Assume that $s \in Z$. Then $s \in [\![\varphi]\!]$ and either $s \in [\![\langle\!\langle C^{\mathbf{b}}\rangle\!\rangle \mathsf{X} \langle\!\langle C^{\mathbf{b}}\rangle\!\rangle \mathsf{G}\, \varphi]\!]$ or $s \in Pre(C, Z, \mathbf{0})$. We proceed as in the base case in defining a strategy. Assume that the strategy is defined for any history $h \in H^n$ and any $\mathbf{b}' \leq \mathbf{b}$.

1. If for any $h \in H^n$ and any $i \leq n$ we have that $h_i \in [\![\langle\!\langle C^{\mathbf{b}_2}\rangle\!\rangle \mathsf{G}\, \varphi]\!]$ then the strategy σ_C has already been defined and $\mathbf{b}_2 < \mathbf{b}$. Let $H^{n+1} = \{h \cdot q \mid q \in out(h_n, \sigma_C)\}$. Let $\mathbf{b}' = max\{cost(q, \sigma_C(last(h \cdot q)), \mathbf{c})\}$. Since $\mathbf{b}_2 < \mathbf{b}$ we have that each of the q are in $[\![\langle\!\langle C\rangle\!\rangle^{\mathbf{b}_2 - \mathbf{b}'} \mathsf{G}\, \varphi]\!]$ as wanted.
2. if $last(h) \in Z \subseteq \mathsf{G}^{\mathbf{b}}_{C,\varphi}(Z)$, then $h_n \in [\![\varphi]\!]$ and either (a) $h_n \in [\![\langle\!\langle C^{\mathbf{b}}\rangle\!\rangle \mathsf{X} \langle\!\langle C^{\mathbf{b}}\rangle\!\rangle \mathsf{G}\, \varphi]\!]$ or (b) $h_n \in pre(C, [\![\varphi]\!], \mathbf{0})$. We only detail the second case. A proof of the former one can be retrieved from the one in [2]. Suppose that $h_n \in pre(C, [\![\varphi]\!], \mathbf{0})$, this means that there is a move for the coalition C that has cost zero w.r.t. any possible Ag move. Let α be this move, we have that $Post(h_n, \alpha) \subseteq Z$. Define σ_C to be exactly α when the history h is given as input.

From the definition of σ_C we have that any path that is compatible with this strategy is action $\mathbf{b}$-consistent and satisfy φ. We thus obtain the wanted result.

Theorem 3. *The model-checking problem for RAB-ATL is decidable: given a finite RAB-CGS $\mathfrak{A}$ and a formula φ, we can compute $[\![\varphi]\!]$. Moreover, the model-checking problem for RAB-ATL is in the same complexity class as the one of RB-ATL and can be computed in* $\mathbf{O}(|\mathfrak{A}| \times |\mathfrak{A}| \times |\varphi|^{2r+1})$.

Proof. Algorithm 2 computes, given a formula φ and a RAB-CGS $\mathfrak{A}$, the set of states satisfying φ. Soundness and completeness of the Algorithm follows by Proposition 2 and Lemma 1. The complexity result follows from the fact that Algorithm 2 is exactly the same as the one provided for model checking RB-ATL in [2] the only difference being how the function $Pre(C, X, \mathbf{b})$ is computed. We have already established that this function is quadratic in the cardinality of the transition function of $\mathfrak{A}$ (Proposition 2).

5 Implementation

Both RB-ATL and RAB-ATL model checking algorithms have been implemented[4] in Python as a component of the `VITAMIN` framework [18]. Furthermore, the RB-CGS and RAB-CGS models have been added in `VITAMIN` as well to allow the definition of resources in the CGSs.

[4] `https://anonymous.4open.science/r/RABATL-43F4`

Algorithm 2 *ModelChecking* $(\mathfrak{M}, \varphi)$

for all $\varphi \in Sub(\varphi)$ **do**
  switch φ **do**
    case $\varphi = \top$
      $[\![\varphi]\!] \leftarrow S$
    case $\varphi = p$
      $[\![\varphi]\!] \leftarrow \{s \in S \ : \ p \in L(s)\}$
    case $\varphi = \neg\varphi_1$
      $[\![\varphi]\!] \leftarrow S \setminus [\![\varphi_1]\!]$
    case $\varphi = \varphi_1 \wedge \varphi_2$
      $[\![\varphi]\!] \leftarrow [\![\varphi_1]\!] \cap [\![\varphi_2]\!]$
    case $\varphi = \langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{X} \varphi_1$
      $[\![\varphi]\!] \leftarrow Pre(C, [\![\varphi_1]\!], \mathbf{b})$
    case $\varphi = \langle\!\langle C^{\mathbf{0}} \rangle\!\rangle \mathsf{G} \varphi_1$
      $X \leftarrow [\![\top]\!]$; $Y \leftarrow [\![\varphi_1]\!]$
      while $Y \neq X$ **do**
        $X \leftarrow Y$
        $Y \leftarrow [\![\varphi_1]\!] \cap Pre(C, [\![\varphi_1]\!], \mathbf{0})$
      $[\![\varphi]\!] \leftarrow Y$
    case $\varphi = \langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \mathsf{G} \varphi_1$ for $\mathbf{b} > \mathbf{0}$
      $X \leftarrow \emptyset$; $Y \leftarrow \emptyset$
      for all $\mathbf{b}' < \mathbf{b}$ **do**
        $Y \leftarrow Pre(C, [\![\langle\!\langle C^{\mathbf{b}'} \varphi_1 \rangle\!\rangle]\!], \mathbf{b} - \mathbf{b}') \cap [\![\varphi_1]\!]$
        while $X \neq Y$ **do**
          $X \leftarrow X \cup Y$; $Y \leftarrow Pre(C, Y, \mathbf{0}) \cap [\![\varphi_1]\!]$
      $[\![\varphi]\!] \leftarrow X$
    case $\varphi = \langle\!\langle C^{\mathbf{0}} \rangle\!\rangle \varphi_1 \mathsf{U} \varphi_2$
      $X \leftarrow \emptyset$; $Y \leftarrow [\![\varphi_2]\!]$
      while $X \neq Y$ **do**
        $X \leftarrow X \cup Y$; $Y \leftarrow Pre(C, Y, \mathbf{0}) \cap [\![\varphi_1]\!]$
    $[\![\varphi]\!] \leftarrow X$
    case $\varphi = \langle\!\langle C^{\mathbf{b}} \rangle\!\rangle \varphi_1 \mathsf{U} \varphi_2$ for $\mathbf{b} > \mathbf{0}$
      $X \leftarrow \emptyset$; $Y \leftarrow \emptyset$
      for all $\mathbf{b}' < \mathbf{b}$ **do**
        $Y \leftarrow Pre(C, [\![\langle\!\langle C^{\mathbf{b}'} \rangle\!\rangle \varphi_1 \mathsf{U} \varphi_2]\!], \mathbf{b} - \mathbf{b}') \cap [\![\varphi_1]\!]$
        while $X \neq Y$ **do**
          $X \leftarrow X \cup Y$; $Y \leftarrow Pre(C, X, \mathbf{0}) \cap [\![\varphi_1]\!]$
      $[\![\varphi]\!] \leftarrow X$

5.1 Experiments

We tested our implementation on a machine with the following specifications: Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz, 4 cores 8 threads, 16 GB RAM DDR4.

We experimented our implementation on the motivating example presented along the paper. Specifically, considering the RB-CGS $\mathfrak{M}$ presented in Example 1 and the RB-ATL formula $\varphi = \langle\!\langle \{1,2\}^{\langle \mathbf{b}_1, \mathbf{b}_2 \rangle} \rangle\!\rangle \top \mathsf{U}\, s_6$ reported in Example 2, we carried out experiments to validate our approach. First, we considered the case when $\mathbf{b}_1 = 3$ and $\mathbf{b}_2 = 1$, where, as expected, the tool confirmed the satisfaction of φ on $\mathfrak{M}$ within 0.005 seconds. After that, we considered the case when $\mathbf{b}_1 = 2$ and $\mathbf{b}_2 = 1$ confirming the violation of φ on $\mathfrak{M}$ within 0.019 seconds.

Subsequently, we experimented our implementation on the RAB-CGS $\mathfrak{A}^{\mathfrak{M}}$ equivalent to $\mathfrak{M}$, which can be obtained as reported in Example 4. As expected, our implementation confirmed the satisfaction of φ on $\mathfrak{A}^{\mathfrak{M}}$ when $\mathbf{b}_1 = 3$ and $\mathbf{b}_2 = 1$ within 0.011 seconds, as well as the violation of φ on $\mathfrak{A}^{\mathfrak{M}}$ when $\mathbf{b}_1 = 2$ and $\mathbf{b}_2 = 1$ within 0.026 seconds.

We then conducted additional experiments on $\mathfrak{A}$, the revised version of the $\mathfrak{A}^{\mathfrak{M}}$ RAB-CGS, reported in Example 3, where in state s_I, if both agents 1 and 2 exchange data by selecting $\langle send_{12}, send_{21}\rangle$, the cost of each action is only 0.5 energy (resulting in a total cost of 1). As expected, φ with $\mathbf{b}_1 = 3$ and $\mathbf{b}_2 = 1$ was still considered satisfied in $\mathfrak{A}$ within 0.015 seconds. However, φ with $\mathbf{b}_1 = 2$ and $\mathbf{b}_2 = 1$, that was violated in $\mathfrak{A}^{\mathfrak{M}}$, was considered satisfied in $\mathfrak{A}$ within 0.008 seconds. Indeed, because RAB-ATL enables consideration of an agent's action costs relative to the actions performed by other agents, the verification process was able to determine the satisfaction of the property as the resulting strategy traversed through state s_3.

To better evaluate the implementations' performance, we carried out additional experiments. Figure 2 presents the results of our instantiation when applied to the model depicted in Figure 1, while varying the number of resources. In this plot, on the x-axis, we report the number of resources used in the model. With respect to Figure 1, we replicate the number of energy and memory resources to simulate more complex variations of the running example. For instance, in the x-axis with value 2, we have 2 memory and 2 energy resources to consider, with value 4, we have 4 memory and 4 energy resources, and so on. The times taken for the verification are reported in logarithmic scale to improve visualisation. As anticipated, the behaviour exhibits exponential growth concerning the number of resources. This empirically confirms our theoretical results on RAB-ATL and the theoretical results on RB-ATL presented in [2] as well.

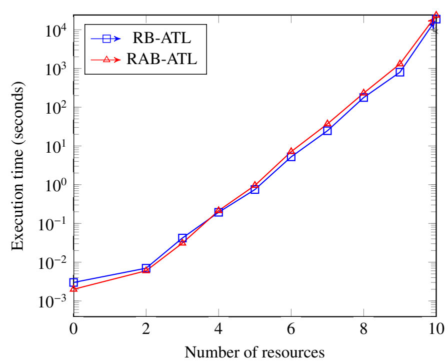


Fig. 2. Time taken by RB-ATL and RAB-ATL for different number of resources when applied to model of Figure 1. The times are reported in logarithmic scale.

Figure 3 presents the experimental results obtained by varying the system model. In these synthetic experiments, the models differ from the one presented in Example 1 and shown in Figure 1; they are automatically generated. Each model has a specific size, determined by the sum of the states and transitions within it, and consists of two agents and two resources. Note that, for each RB-CGS so randomly generated, a corresponding equivalent RAB-CGS is obtained as well to perform both RB-ATL and RAB-ATL model checking. Figure 3 displays the execution time required for model checking each of these models against a fixed strategic formula with two resources, bound set to 5 for each, and a temporal objective. The distribution of the atomic propositions among the states is randomly generated for each model. As observed in Figure 3, the results exhibit the expected polynomial behaviour relative to the size of the model under analysis.

It is important to note that, for each reported model size, over 10,000 models have been randomly generated and verified against the strategic formula. The plot depicts the average execution time.

Furthermore, it is important to observe that both RB-ATL and RAB-ATL model checking exhibit polynomial behaviour relative to the size of the model, although RAB-ATL shows a steeper slope. This difference is attributed to the more complex evaluation of costs, which, instead of being statically defined for each action in every state, depend on all the other actions performed by all the other agents in each state, necessitating consequent evaluation.

We also want to emphasise that the size of the models used in these experiments is not negligible. In fact, the largest models subjected to verification contain more than 100 states and over 10,000 transitions. Nonetheless, despite their size, both algorithms manage to complete the verification process in less than 25 seconds (even in 5 seconds in the case of RB-ATL).

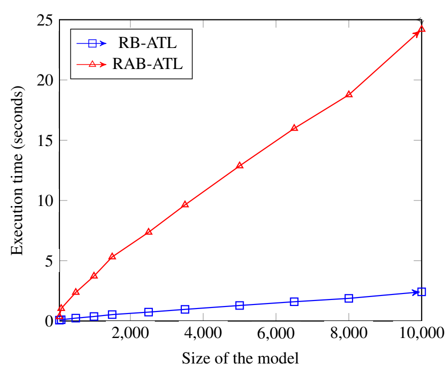


Fig. 3. Time taken by RB-ATL and RAB-ATL for different numbers of states (with 2 resources, bound set to 5 for each, and 2 agents).

6 Related Work

Formal models of resource-bounded agents have garnered increasing attention in recent years [1,2,3,7,9,20,22]. In this line of research, the focus lies on understanding the behaviour of agents operating under fixed resource constraints. In this area, the work presented in [2] is the most closely related to ours. It introduces RB-ATL along with its syntax, semantics, and resulting model checking problem. The distinctions from RAB-ATL have been discussed along the paper. In summary, RAB-ATL is designed to acknowledge that agents can influence each other, particularly concerning the costs associated with the actions they execute. Another relevant work is [15], which diverges from our approach by emphasising the internal mental processes of individual agents within a group, rather than designing a logic to reason about coalitions and strategies. In this sense, this work is complementary rather than competitive. The Price Resource-Bounded ATL (PRB-ATL) logic, introduced in [20], addresses resource endowment within the system when evaluating formulae pertaining to agent coalitions. In this model, resources are convertible to money and are bounded. The model-checking problem for PRB-ATL is decidable, with complexity similar to RB-ATL. Unlike our approach, PRB-ATL considers bounds for both agents inside and outside the coalition under analysis. However, similar to RB-ATL, the cost of actions is determined solely by the state, not influenced by actions performed by other agents. Hence, PRB-ATL stands as orthogonal to our work. Nonetheless, it is conceivable to imagine an extension of [20], referred to as PRAB-ATL, to integrate a revised pricing mechanism for actions, aligning with our approach in this paper (this exploration is out of the scope of this contribution). Comparing [10] to our approach reveals a close technical and conceptual relationship. However, unlike our approach, which focus on resource-bound verification, the framework in [10] considers a wider range of consequences from collective actions, including gains, incentives, and rewards. Such a distinction leads to differences in the model checking problem, as well as in the logical languages to be used and their formal semantics. Other works on the synthesis of resource-aware controllers in MAS can be found in [13,14].

Note that, at the time of submission of this work and to the best of our knowledge, except for [4], whose implementation could not be located by the authors of this work, there is no existing implementation of resource-bound model checking for ATL. Naturally, this pertains to standard RB-ATL model checking. As RAB-ATL is introduced for the first time in this work, its implementation is entirely novel.

7 Conclusions and Future Work

We have introduced a novel variant of RB-ATL, called Resource Action-based Bounded ATL (RAB-ATL), to address its limitations. Our investigation revealed the foundational differences between RB-ATL and RAB-ATL, particularly in their treatment of actions' costs and their implications for agent interaction within MAS. The introduction of RAB-ATL provided a richer approach to resource handling, allowing for a better exploitation of cooperation and competition amongst agents. For example, considering the well-known prisoner's dilemma, the same action taken by one prisoner has a different cost (or reward) depending on the action taken by the other prisoner. Furthermore, our implementation approach and experimental evaluation demonstrated the feasibility and effectiveness of RAB-ATL.

We anticipate that the insights gained from this study will pave the way for future research directions in the field of MAS and formal verification when resource limitations are considered. For instance, our logic might be ideal for specifying properties in the context of distributed systems where agents can share data, but sharing has an impact on agents' resources.

In summary, the introduction of RAB-ATL represents a significant advancement in the domain of resource-bound verification in MAS, offering a more holistic approach to modeling and analysing agents' behaviour.

References

1. Alechina, N., Logan, B., Nga, N.H., Rakib, A.: Expressing properties of coalitional ability under resource bounds. In: LORI 2009. vol. 5834, pp. 1–14 (2009)
2. Alechina, N., Logan, B., Nga, N.H., Rakib, A.: Resource-bounded alternating-time temporal logic. In: AAMAS 2010. pp. 481–488 (2010)
3. Alechina, N., Logan, B., Nga, N.H., Rakib, A.: Logic for coalitions with bounded resources. J. Log. Comput. **21**(6), 907–937 (2011)
4. Alechina, N., Logan, B., Nguyen, H.N., Raimondi, F., Mostarda, L.: Symbolic model-checking for resource-bounded ATL. In: AAMAS 2015. pp. 1809–1810 (2015)
5. Alur, R., Henzinger, T.A., Kupferman, O.: Alternating-time temporal logic. J. ACM **49**(5), 672–713 (2002)
6. Ballot, G., Malvone, V., Leneutre, J., Laarouchi, Y.: Strategic reasoning under capacity-constrained agents. In: AAMAS 2024. pp. 123–131 (2024)
7. Belardinelli, F., Demri, S.: Strategic reasoning with a bounded number of resources: The quest for tractability. Artif. Intell. **300**, 103557 (2021)
8. Belardinelli, F., Jamroga, W., Malvone, V., Mittelmann, M., Murano, A., Perrussel, L.: Reasoning about human-friendly strategies in repeated keyword auctions. In: AAMAS 2022. pp. 62–71 (2022)
9. Bulling, N., Farwer, B.: Expressing properties of resource-bounded systems: The logics rtl* and RTL. In: CLIMA X. pp. 22–45 (2009)
10. Bulling, N., Goranko, V.: Combining quantitative and qualitative reasoning in concurrent multi-player games. Auton. Agents Multi Agent Syst. **36**(1), 2 (2022)
11. Catta, D., Leneutre, J., Malvone, V.: Obstruction logic: A strategic temporal logic to reason about dynamic game models. In: ECAI 2023. pp. 365–372 (2023)
12. Catta, D., Leneutre, J., Malvone, V., Murano, A.: Obstruction alternating-time temporal logic: A strategic logic to reason about dynamic models. In: AAMAS 2024. pp. 271–280 (2024)
13. Condurache, R., Dima, C., Oualhadj, Y., Troquard, N.: Rational synthesis in the commons with careless and careful agents. In: AAMAS '21. pp. 368–376 (2021)
14. Condurache, R., Dima, C., Oualhadj, Y., Troquard, N.: Synthesis of resource-aware controllers against rational agents. In: AAMAS 2023. pp. 775–783 (2023)
15. Costantini, S., Formisano, A., Pitoni, V.: An epistemic logic for multi-agent systems with budget and costs. In: JELIA 2021. pp. 101–115 (2021)
16. Ferrando, A., Luongo, G., Malvone, V., Murano, A.: Theory and practice of quantitative atl. In: PRIMA 2024. vol. to appear (2024)
17. Ferrando, A., Malvone, V.: Hands-on VITAMIN: A compositional tool for model checking of multi-agent systems. In: WOA 2024. pp. 148–160 (2024)
18. Ferrando, A., Malvone, V.: VITAMIN: A compositional framework for model checking of multi-agent systems. CoRR **abs/2403.02170** (2024)
19. Jamroga, W., Malvone, V., Murano, A.: Natural strategic ability. Artif. Intell. **277** (2019)
20. Monica, D.D., Napoli, M., Parente, M.: On a logic for coalitional games with priced-resource agents. In: LAMAS 2011. pp. 215–228 (2011)
21. Nguyen, H.N., Alechina, N., Logan, B., Rakib, A.: Alternating-time temporal logic with resource bounds. J. Log. Comput. **28**(4), 631–663 (2018)
22. Nguyen, H.N., Alechina, N., Logan, B., Rakib, A.: Alternating-time temporal logic with resource bounds. J. Log. Comput. **28**(4), 631–663 (2018)