A Probabilistic Logic for Resource-Bounded Multi-Agent Systems

Hoang Nga Nguyen[1] and **Abdur Rakib**[2]

[1]Institute for Future Transport and Cities, Coventry University, UK
[2]Computer Science Research Centre, University of the West of England, UK
Hoang.Nguyen@coventry.ac.uk, Rakib.Abdur@uwe.ac.uk

Abstract

Resource-bounded alternating-time temporal logic (RB-ATL) [Alechina *et al.*, 2010], an extension of Coalition Logic (CL) and Alternating-time Temporal Logic (ATL), allows reasoning about resource requirements of coalitions in concurrent systems. However, many real-world systems are inherently probabilistic as well as resource-bounded, and there is no straightforward way of reasoning about their unpredictable behaviours. In this paper, we propose a logic for reasoning about coalitional power under resource constraints in the probabilistic setting. We extend RB-ATL with probabilistic reasoning and provide a standard algorithm for the model-checking problem of the resulting logic Probabilistic Resource-Bounded ATL (pRB-ATL).

1 Introduction

The Alternating-time temporal logic (ATL) [Alur *et al.*, 2002] and Coalition logic (CL) [Pauly, 2002] were introduced as logical formalisms for analysing the strategic abilities of coalitions with temporal winning conditions. In the literature, several variants of ATL and CL have been proposed (see e.g., [Goranko, 2001; Ågotnes *et al.*, 2009; Wooldridge and Dunne, 2004]). These logics allow the specification of system behaviour in terms of logical formulae, and they can be used to express many interesting properties of coalitions and strategies. However, there is no straightforward way of expressing resource requirements for systems of reasoning agents in these logics. Recently, there has been growing interest in formal models of resource-bounded agents [Bulling and Farwer, 2010; Alechina *et al.*, 2010; Monica *et al.*, 2011]. In works on reasoning about resource-bounded agents, the main emphasis is on the behaviour of agents constrained by fixed resource bounds. For example, RB-ATL [Alechina *et al.*, 2010] was developed for reasoning about coalitional ability under resource bounds, which allows expressing various resource-bounded properties, including *"coalition A has a strategy to reach a state satisfying φ under the resource bound b, but they cannot enforce φ under a tighter resource bound $b'(< b)$"*.

Many real-world distributed systems, such as the Internet of Things (IoT) and Cyber-Physical Systems (CPS), are deeply rooted in activities of our daily living [Calvaresi *et al.*, 2017]. The Multi-Agent Systems (MAS) paradigm offers an ideal method to model and implement such systems [Leitão *et al.*, 2016]. These systems often operate in unpredictable and/or uncertain environments [Faza *et al.*, 2009; Zhang *et al.*, 2016]. Their applications encompass many safety-critical domains, and many such applications run in resource-constrained devices and environments [Abbas *et al.*, 2015; Laszka *et al.*, 2015]. They require not only the reasoning about the coalitional behaviour of agents but also that the agents and/or the environment may have random or unreliable behaviours. A rigorous analysis and verification approach can ensure the correctness of their designs [Kwiatkowska, 2016]. There has recently been increasing interest in developing logics with a probabilistic component in order to link logical and probabilistic reasoning [Forejt *et al.*, 2011; Huang *et al.*, 2012; Chen *et al.*, 2013]. These logics are essentially extensions of CTL or ATL which allow for probabilistic quantification of described properties. Research in this area includes developing a model-based verification tool PRISM by Kwiatkowska et. al [Kwiatkowska *et al.*, 2002]. Among other PRISM property specification languages, the logic rPATL [Chen *et al.*, 2013] allows reasoning quantitatively about a system's use of resources and emphasises on expected reward-related measures. In rPATL, we can express that a coalition of agents has a strategy which can ensure that either the probability of an event's occurrence or an expected reward measure meets some threshold. However, probabilistic resource-bounded properties such as

- *"can coalition A have a strategy so that the probability of reaching a state satisfying φ under the resource bound b is at least v?"*,
- *"a coalition of agents A has a strategy to achieve a property φ with probability v provided they have resources b, but they cannot enforce φ under a tighter resource bound $b'(< b)$ with probability v"*, and
- *"a coalition of agents A can maintain φ until ψ becomes true with probability v provided they have resources b"*

can neither be expressed in rPATL nor in any other probabilistic temporal logics mentioned above. The problem discussed in this paper can be identified as a probabilistic resource-constrained multi-agent coalition structure formation problem, where the coalitions perform actions towards specific goals, provided they have enough resources. We propose a

logic, Probabilistic Resource-Bounded ATL (pRB-ATL), for reasoning about coalitional power under resource constraints in the probabilistic setting. pRB-ATL allows us to express various interesting properties of probabilistic systems where abilities of agents are constrained by available resources. The logic rPATL is one of the approaches most closely related to the work presented in this paper. However, the reasoning problem considered in our work is different in two important aspects. First, and most importantly, properties in rPATL related to rewards are of a statistical nature. They are expressed and computed as constraints on expected values for rewards. In contrast, resource-bounded properties in our pRB-ATL logic lie within the realm of crisp values and constraints; actions and strategies are allowed if and only if they satisfy the resource-bounded constraints. Second, the semantics of rPATL is based on turn-based systems while ours is based on concurrent systems.

The rest of this paper is structured as follows. In Section 2, we discuss the basic notions of probability distributions and the underlying probabilistic formalisms of our logic. In Section 3, we present the syntax and semantics of pRB-ATL. In Section 4, we provide a model-checking algorithm for pRB-ATL, and conclude the paper in Section 5.

2 Background and Preliminaries

In this section, we discuss the basic notions that are used in the technical part of the proposed logic. Let Q be a finite set and $\mu : Q \to [0, 1]$ be a *probability distribution* function over Q such that $\sum_{q \in Q} \mu(q) = 1$. We denote by $\mathcal{D}(Q)$ the set of all such distributions over Q. For a given $\mu \in \mathcal{D}(Q)$, $\mathsf{supp}(\mu) = \{q \in Q \mid \mu(q) > 0\}$ is called the *support* of μ. A *probability space* is a measure space with total measure 1. The standard notation of a *probability space* is a triple $(\Omega, \mathcal{F}, Pr)$, where Ω is a sample space which represents all possible outcomes, $\mathcal{F} \subseteq \mathcal{P}(\Omega)$ is a σ-algebra over Ω which includes the empty subset and it is closed under countable unions and complement, and $Pr : \mathcal{F} \to [0, 1]$ is a probability measure over $(\Omega, \mathcal{F})$. The interested reader is referred to [Billingsley, 2012] for a complete description relating to probability distributions and measures. We also denote the set of all finite, non-empty finite and infinite sequences of elements of Q by Q^*, Q^+ and Q^ω, respectively.

2.1 DTMC and MDP

Discrete-time Markov chains (DTMCs) are the simplest probabilistic models in which the systems evolve through discrete time steps.

Definition 1. *A DTMC is a tuple $M_c = (Q, q_0, \Pi, \pi, \delta)$, where Q is a set of states, $q_0 \in Q$ is the initial state, Π is a finite set of propositional variables, $\pi : Q \to \wp(\Pi)$ is a labelling function, and $\delta : Q \times Q \to [0, 1]$ is a probability transition matrix such that $\sum_{q' \in S} \delta(q, q') = 1$ for all $q \in Q$. Here, $\delta(q, q')$ denotes the probability that the chain, whenever in state q, moves into next state q'.*

Markov decision processes (MDPs), an extension to ordinary DTMCs, are a widely used formalism for modelling systems that exhibit both probabilistic and non-deterministic behaviour [Forejt *et al.*, 2011].

Definition 2. *An MDP is a tuple $M_d = (Q, q_0, \Pi, \pi, \mathcal{A}, \delta)$, where $\mathcal{A}$ is a set of actions, $\delta : Q \times \mathcal{A} \to \mathcal{D}(Q)$ is a (partial) probabilistic transition function, and all the other components are the same as their counterparts in a DTMC.*

The set of available actions at a state q is defined by $\mathcal{A}(q) = \{\alpha \in \mathcal{A} \mid \exists q' \cdot \delta(q, \alpha)(q') > 0\}$. Unlike DTMCs, in MDPs, the transitions between states occur in two steps. Firstly, an action α is selected from a set of actions $\mathcal{A}(q)$ available at a given state q. Secondly, a successor state q' is chosen randomly, according to the probability distribution $\delta(q, \alpha)$. Thus, for a given state q and $\alpha \in \mathcal{A}(q)$, $\delta(q, \alpha) : Q \to [0, 1]$ is a function such that $\sum_{q' \in Q} \delta(q, \alpha)(q') = 1$.

Definition 3. *A path λ in an MDP M_d is an (infinite) alternating sequence of states and actions $\lambda = q_0 \xrightarrow{\alpha_0} q_1 \xrightarrow{\alpha_1} \ldots \in (Q \times \mathcal{A})^\omega$ where $\alpha_i \in \mathcal{A}(q_i)$ and $\delta(q_i, \alpha_i)(q_{i+1}) > 0$ for all $i \geq 0$. A finite path $\lambda = q_0 \xrightarrow{\alpha_0} q_1 \xrightarrow{\alpha_1} \ldots \xrightarrow{\alpha_{n-1}} q_n$ is defined as usual as a prefix of an infinite path ending at a state q_n. The set of finite paths is denoted by $(Q \times \mathcal{A})^* Q$.*

Definition 4. *A strategy in an MDP M_d is a function $f : (Q \times \mathcal{A})^* Q \to \mathcal{D}(\mathcal{A})$ that assigns each finite path $\lambda = q_0 \xrightarrow{\alpha_0} q_1 \xrightarrow{\alpha_1} \ldots \xrightarrow{\alpha_{n-1}} q_n$ a probability distribution over the enabled actions $\mathcal{A}(q_n)$ such that $f(\lambda)(\alpha) > 0$ if $\alpha \in \mathcal{A}(q_n)$.*

An MDP reduces to a DTMC if the player's strategy is fixed. That is, given an MDP M_d, any strategy induces a Markov chain M_c by resolving non-determinism.

3 Syntax and Semantics of pRB-ATL

In this section, we provide the syntax and semantics of pRB-ATL. Let us consider a multi-agent system consisting of a set $N = \{1, 2, \ldots n\}$ of $n(\geq 1)$ concurrently executing agents. In order to reason about resources, we assume that the actions performed by the agents have costs. Let R be a finite set of resources (such as money, energy, or anything else which may be required by an agent for performing an action). Without loss of generality, we assume that the cost of an action, for each of the resources, is a natural number. The set of resource bounds $\mathbb{B}$ over R is defined as $\mathbb{B} = (\mathbb{N} \cup \{\infty\})^r$, where $r = |R|$. We denote by $\vec{0}$ the smallest resource bound $(0, \ldots, 0)$ and by $\vec{\infty}$ the greatest resource bound $(\infty, \ldots, \infty)$.

3.1 Syntax of pRB-ATL

Let Π be a finite set of atomic propositions, N be the set of agents, A be a subset of N, and $b \in \mathbb{B}$. The syntax of $pRB-ATL$ is defined as follows:

$$\varphi := \top \mid p \mid \neg \varphi \mid \varphi \vee \varphi \mid \langle\!\langle A^b \rangle\!\rangle \mathsf{P}_{\bowtie v}[\psi]$$

$$\psi := \bigcirc \varphi \mid \varphi \, \mathcal{U}^k \, \varphi \mid \neg \psi$$

where $p \in \Pi, \bowtie \in \{<, \leq, \geq, >\}, v \in \mathbb{Q} \cap [0, 1], k \in \mathbb{N} \cup \{\infty\}$.

The two temporal operators have a standard meaning, $\bigcirc$ for "next" and $\mathcal{U}^{\leq k}$ for "bounded until" if $k < \infty$ or "until" otherwise. When $k = \infty$, we shall simply write $\mathcal{U}$ instead of $\mathcal{U}^\infty$. Here, $\langle\!\langle A^b \rangle\!\rangle \mathsf{P}_{\bowtie v}[\bigcirc \varphi]$ means that a coalition A has a strategy to make sure that the next state satisfies φ under resource bound b with a probability in relation $\bowtie$ with constant v, regardless of the strategies of other players. The formula

$\langle\langle A^b \rangle\rangle \mathsf{P}_{\bowtie v}[\varphi_1 \mathcal{U} \varphi_2]$ means that A has a strategy to enforce φ_2 while maintaining the truth of φ_1, and the cost of this strategy is at most b with a probability in relation $\bowtie$ with constant v, regardless of the strategies of other players. Other temporal operators are defined as abbreviations in a standard way. Particularly, "eventually" is defined as $\Diamond\varphi \equiv \top \mathcal{U} \varphi$, and "always" as $\Box \equiv \neg\Diamond\neg\varphi$. Notice that these operators when $b = \vec{\infty}$ mean the same as their counterparts in ATL, i.e., the ATL operator $\langle\langle A \rangle\rangle$ corresponds to $\langle\langle A^{\vec{\infty}} \rangle\rangle$. Other classical abbreviations for $\perp$, $\vee$, $\rightarrow$ and $\leftrightarrow$ are defined as usual.

3.2 Semantics of pRB-ATL

To interpret this language, we extend the definition of resource-bounded concurrent game structures (RB-CGS)) [Alechina *et al.*, 2010] with probabilistic behaviours of agents. For consistency with [Alur *et al.*, 2002], in what follows the terms 'agents' and 'players' and the terms 'actions' and 'moves' have been used interchangeably.

Definition 5. *A Probabilistic Resource Concurrent Game Structure (pRCGS) is a tuple $S = (n, r, Q, \Pi, \pi, d, c, \delta)$ where:*

- *$n \geq 1$ is the number of players (agents);*
- *$r \geq 0$ is the number of resources;*
- *Q is a non-empty finite set of states;*
- *Π is a finite set of propositional variables;*
- *$\pi : \Pi \to \wp(Q)$ is a function which assigns a subset of Q to each variable in Π;*
- *$d : Q \times N \to \mathbb{N}_+$ is a function which indicates the number of moves (actions) available at a state for each agent, where $\mathbb{N}_+ = \mathbb{N} \setminus \{0\}$;*
- *$c : Q \times N \times \mathbb{N}_+ \to \mathbb{B}$ is a partial function which indicates a resource cost for each move by an agent at a state; and $c(q, a, 1) = \vec{0}$ for any $q \in Q$ and $a \in N$;*
- *$\delta : Q \times (N \to \mathbb{N}_+) \to \mathcal{D}(Q)$ is a partial probabilistic transition function.*

Given a pRCGS $S = (n, r, Q, \Pi, \pi, d, c, \delta)$, we identify available moves at a state $q \in Q$ of an agent $i \in N$ by $1, \ldots, d(q, i)$; then $D_i(q) = \{1, \ldots, d(q, i)\}$ denotes the set of available moves; move 1 specifies idling which is always available with cost $\vec{0}$ by definition. Similar to ATL and RB-ATL, the zero-cost move 1 is required to avoid deadlock and, therefore, maintain totality.

A pRCGS is closely related to MDPs (Definition 2, Section 2.1), where abilities of individual agents and coalitions of agents are constrained by available resources in a nontrivial way. Given $A \subseteq N$, a joint move m of A is a function $m : A \to \mathbb{N}_+$. Let D denote the set of all joint actions for N, i.e., $D = N \to \mathbb{N}_+$. Given $q \in Q$, the set of available joint moves of A at q is denoted by $D_A(q) = \{m : A \to \mathbb{N}_+ \mid \forall a \in A : m(a) \in D_a(q)\}$. Given $q, q' \in Q$ and $m \in D(q)$, $\delta(q, m)(q')$ is the conditional probability of a transition from q to q' if every agent $i \in N$ performs $m(i)$. Then, q' is called a successor of q if $\exists m \in D(q)$ such that $q' \in \text{supp}(\delta(q, m))$. To this end, pRCGS is different from RB-CGS in defining the transition function δ. While the δ of a RB-CGS is deterministic, that of a pRCGS is a mapping to a distribution function over states and, hence, specifies non-determinism.
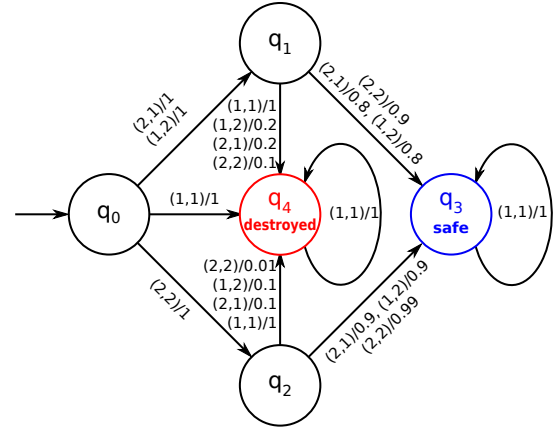

Figure 1: pRCGS S_{ff} of the two firefighters.

Example 1. *Let us consider the design of a system consisting of two firefighter agents $N = \{1, 2\}$ in a building. Each agent is equipped with two resources:* electricity *and* water*. Agents can perform three possible actions, namely,* sense*, pump water, and* idle*. They can sense to detect if there is a fire in the building and pump the water to stop the fire. Sensing the fire requires one unit of electricity, pumping water requires one unit of electricity and one unit of water, and idle costs nothing. This scenario is formalised by a pRCGS S_{ff} as depicted in Figure 1. Here, $n = 2$, $r = 2$, $Q = \{q_0, q_1, q_2, q_3, q_4\}$, and $\Pi = \{\text{safe}, \text{destroyed}\}$. For convenience, the transition function δ is written in terms of labels on transitions. Each transition from a state q_i to a state q_j is annotated with one or more labels of the form $(x, y)/z$ where (x, y) denotes the joint move, x by agent 1's move and y by agent 2's move performed at state q_i, and z denotes the probability of arriving at the next state q_j.*

At the initial state q_0, each agent can either stay idle 1 or perform sense *action 2. Therefore, the possible joint moves at q_0 are $(1, 1)$, $(1, 2)$, $(2, 1)$, and $(2, 2)$. At the states q_1 (only one agent detected the fire) and q_2 (both detected the fire), each agent can either stay idle 1 or pump water 2. Thus, possible joint moves at q_1 and q_2 are (1,1), (2,1), (1,2), and (2,2). At the "blue" state q_3, the building is saved from the fire and is labelled with a proposition "safe", while at the "red" state q_4, the building is destroyed and is labelled with a proposition "destroyed". If a fire does occur in the building, each agent autonomously decides to detect it or stay idle. When only one of the agents detects the fire, the chance of stopping it is 80% if only one of them acts. However, the chance of stopping it increases to 90% if both of them act. The effectiveness of stopping the fire could be improved if both the agents detect the fire jointly. The chance of stopping it then would be 90% if only one of them acts, while it would be 99% if both of them act. If both the agents stay idle (i.e., neither of them detects the fire nor pumps the water to extinguish the fire), the building will be destroyed.*

To compare costs and resource bounds, we use the usual point wise vector comparison, that is, $(b_1, \ldots, b_r) \leq (d_1, \ldots, d_r)$ iff $b_i \leq d_i$ for $i \in \{1, \ldots, r\}$ where $n \leq \infty$ for all $n \in \mathbb{N}$. We also use pointwise vector addition:

$(b_1, \ldots, b_r) + (d_1, \ldots, d_r) = (b_1 + d_1, \ldots, b_r + d_r)$ where $n + \infty = \infty$ for all $n \in \mathbb{N} \cup \{\infty\}$. Given $b \in \mathbb{B}$, $\mathbb{B}_{\leq b}$ denotes the set of bounds less or equal to b without taking into account ∞ components, i.e., $\mathbb{B}_{\leq b} = \{b' \in \mathbb{B} \mid \forall i \in \{1, \ldots, r\} : b'_i = b_i = \infty \vee b'_i \leq b_i < \infty\}$. Note that $|\mathbb{B}_{\leq b}| = \prod_{i \in \{1, \ldots, r\}, b_i \neq \infty} (b_i + 1)$.

Given a joint move $m \in D_A(q)$, the cost of m is defined as $cost(q, m) = \sum_{a \in A} c(q, a, m(a))$, i.e., the total cost of the actions performed by the agents in the coalition A.

Given a pRCGS S, we adopt the Definition 3 to define runs (computations). A infinite run is an infinite sequence $\lambda = q_0 \xrightarrow{m_0} q_1 \xrightarrow{m_1} \ldots \in (Q \times D)^\omega$ where $m_i \in D(q_i)$ and q_{i+1} is as successor of q_i by m_i, i.e., $(q_{i+1}) \in \text{supp}(\delta(q_i, m_i))$ for all $i \geq 0$. We denote the set of all infinite computations by $\Omega_S \subseteq (Q \times D)^\omega$. A finite computation is a finite prefix $\lambda = q_0 \xrightarrow{m_0} q_1 \xrightarrow{m_1} q_2 \ldots \xrightarrow{m_{n-1}} q_n \in (Q \times D)^* Q$ of some infinite sequence in Ω_S. We denote the set of all finite computations by Ω_S^+. For convenience, we denote $(Q \times D)^\omega$ and $(Q \times D)^* Q$ by $(QD)^\omega$ and $(QD)^* Q$, respectively. The length of a computation λ, denoted by $|\lambda|$, is defined as the number of transitions in λ. For a finite computation $\lambda = q_0 \xrightarrow{m_0} q_1 \xrightarrow{m_1} q_2 \ldots \xrightarrow{m_{n-1}} q_n \in \Omega_S^+$, $|\lambda| = n$; for an infinite computation $\lambda \in q_0 \xrightarrow{m_0} q_1 \xrightarrow{m_1} \ldots \Omega_S$, $|\lambda| = \infty$. Given a computation $\lambda \in \Omega_S^+$, $\lambda(i) = q_i$ for all $i \in \{0, \ldots, |\lambda|\}$; $\lambda(i, j) = q_i \ldots q_j$ for all $i, j \in \{0, \ldots, |\lambda|\}$ and $i \leq j$; $m_\lambda = m_0 m_1 \ldots$ as the projection of moves in λ and $m_\lambda(i) = m_i$ for $i \in \{0, \ldots, |\lambda| - 1\}$. Note that $\lambda(|\lambda|)$ is the last state in λ. Finally, $\Omega_{S,q}^+ = \{\lambda \in \Omega_S^+ \mid \lambda(0) = q\}$ denotes the set of (finite) computations starting from $q \in Q$. Given a finite computation $\lambda \in \Omega_S^+$ and a coalition A, the cost of joint actions by A is defined as $cost_A(\lambda) = \sum_{i=0}^{|\lambda|-1} cost(\lambda(i), m_\lambda(i))$. We adopt Definition 4 (Section 2.1) to define strategies:

Definition 6. *Given a pRCGS S, a strategy of a player $a \in N$ is a mapping $f_a : (QD)^* Q \to \mathcal{D}(\mathbb{N}_+)$ which associates each sequence $\lambda \in (QD)^* Q$ to a distribution $\mu_a \in \mathcal{D}(\{1, \ldots, d(\lambda(|\lambda|), a)\})$.*

Definition 7. *A strategy is called* memoryless *(or Markovian) if its choice of moves depends only on the current state, i.e., $f_a(\lambda) = f_a(\lambda(|\lambda|))$ for all $\lambda \in (QD)^* Q$. It is called* deterministic *if it always selects a move with probability 1, i.e., $f_a(\lambda)$ is a Dirac distribution.*

Definition 8. *Given a pRCGS S, a coalition strategy $F_A : A \to ((QD)^* Q \to \mathcal{D}(\mathbb{N}_+))$ is a function which associates each player a in A with a strategy.*

Given a coalition strategy F_A, we show that each finite sequence $\lambda \in (QD)^* Q$ gives rise to a distribution $\mu_\lambda^{F_A} \in \mathcal{D}(D_A(\lambda(|\lambda|)))$ over joint actions $m \in D_A(\lambda(|\lambda|))$ where $\mu_\lambda^{F_A}(m) = \prod_{a \in A} f_a(\lambda)(m(a))$ and $f_a = F_A(a)$ for all $a \in A$.

Lemma 1. *Given a sequence $\lambda \in (QD)^* Q$ and a coalition strategy F_A, $\mu_\lambda^{F_A}$ is a distribution over $D_A(\lambda(|\lambda|))$.*

Given two coalition strategies F_A and F_B of two disjoint coalitions A and B, i.e., $A \cap B = \emptyset$, their union is also a coalition strategy, denoted by $F_A \cup F_B$, for $A \cup B$.

Definition 9. *Given a bound $b \in \mathbb{B}$, and a strategy F_A, F_A is b-bounded iff for all $\lambda \in \Omega_S^+$ such that $cost_A(\lambda) \leq b$, it holds that $\text{supp}(\mu_\lambda^{F_A}) \subseteq \{m \in D_A(\lambda(|\lambda|)) \mid cost(\lambda(|\lambda|), m) \leq b - cost_A(\lambda)\}$.*

In other words, all executions of a b-bounded strategy cost at most b resources. In order to reason about probabilistic behaviour of S, we need to determine the probability that certain computations are taken. To do this, we construct for each state $q \in Q$, a *probability space* over the set of infinite computations $\Omega_{S,q}$ starting from q. The basis of the construction is the probability of individual finite computations induced by the transition probability function δ. Given a state $q_0 \in Q$, we can determine the probability of every finite computation $\lambda = q_0 \xrightarrow{m_0} q_1 \xrightarrow{m_1} q_2 \ldots \xrightarrow{m_{n-1}} q_n \in \Omega_{S,q}^+$ consistent with F_A as follows:

$$\Pr_{S,q_0}^{F_N}(\lambda) = \prod_{i=0}^{n-1} \mu_{\lambda(0,i)}^{F_N}(m_i) \cdot \delta(s_i, m_i)(s_{i+1}).$$

If $|\lambda| = 1$, $\Pr_{q_0}^{F_N}(\lambda) = 1$ as the above product is empty.

For each finite computation $\lambda \in \Omega_S^+$, we can then define a cylinder set C_λ that consists of all infinite computations prefixed by λ. Given an initial state $q \in Q$, it is then standard [Kwiatkowska *et al.*, 2007; Billingsley, 2012] to define a measurable space over $\Omega_{S,q}$, infinite runs of S from q, as $(\Omega_{S,q}, \mathcal{F}_{S,q})$ where $\mathcal{F}_{S,q} \subseteq \wp(\Omega_{S,q})$ is the least σ-algebra on $\Omega_{S,q}$ generated by the family of all cylinder sets C_λ where $\lambda \in \Omega_S^+$ that starts from q, i.e., $\lambda(0) = q$. Given a strategy F_N, a strategy for all players in the game, the behaviour of S is fully probabilistic. It then gives rise to a probability measure $(\Omega_{S,q}, \mathcal{F}_{S,q}, \Pr_{S,q}^{F_N})$ where $\Pr_{S,q}^{F_N} : \mathcal{F}_{S,q} \to [0, 1]$ uniquely extends $\Pr_{S,q}^{F_N} : \Omega_{S,q}^+ \to [0, 1]$ such that $\Pr_{S,q}^{F_N}(C_\lambda) = \Pr_{S,q}^{F_N}(\lambda)$ for all finite computations $\lambda \in \Omega_{S,q}^+$ starting from q.

3.3 Truth Definition for pRB-ATL

Given a pRCGS $S = (n, r, Q, \Pi, \pi, d, c, \delta)$, the truth definition for pRB-ATL is given inductively as follows:

- $S, q \models \top$;
- $S, q \models p$ iff $q \in \pi(p)$;
- $S, q \models \neg\varphi$ iff $S, q \not\models \varphi$;
- $S, q \models \varphi_1 \vee \varphi_2$ iff $S, q \models \varphi_1$ or $S, q \models \varphi_2$;
- $S, q \models \langle\!\langle A^b \rangle\!\rangle \mathsf{P}_{\bowtie v}[\psi]$ iff $\exists$ b-bounded F_A such that $\forall F_{\bar{A}}$, $\Pr_{S,q}^{F_A \cup F_{\bar{A}}}(\{\lambda \in \Omega_{S,q} \mid S, \lambda \models \psi\}) \bowtie v$;
- $S, \lambda \models \bigcirc\varphi$ iff $S, \lambda(1) \models \varphi$;
- $S, \lambda \models \varphi_1 \mathcal{U}^k \varphi_2$ iff $\exists i \in \mathbb{N}$ such that $i \leq k$, $\forall j < i$: $S, \lambda(j) \models \varphi_1$, and $S, \lambda(i) \models \varphi_2$;
- $S, \lambda \models \neg\psi$ iff $S, \lambda \not\models \psi$.

This definition is a combination of pATL and RB-ATL. In particular, the case of $\langle\!\langle A^b \rangle\!\rangle \mathsf{P}_{\bowtie z} \psi$ requires the existence of a strategy F_A which must be b-bounded while there is no restriction on the strategies of the remaining players $\bar{A} = N \backslash A$.

From the truth definition, the following result directly inherits the complement rule for probability where $\geq^{-1} \equiv \leq$, $>^{-1} \equiv <$, $\leq^{-1} \equiv \geq$ and $<^{-1} \equiv >$:

Lemma 2. $\forall S, q : S, q \models \langle\!\langle A^b \rangle\!\rangle \mathsf{P}_{\bowtie v} \psi \Leftrightarrow S, q \models \langle\!\langle A^b \rangle\!\rangle \mathsf{P}_{\bowtie^{-1} 1-v} \neg\psi$.

Example 2. *Let us continue with the running Example 1. Consider a question (property):* "Can agent 1 from q_0, equipped with 2 units of electricity and 1 unit of water, make sure that the building is at least 90% safe?". *This means to check if* $\varphi_{\{1\}} = \langle\langle\{1\}^{(2,1)}\rangle\rangle P_{\geq 0.9}\Diamond safe$ *is true at q_0. Unfortunately, there is no such strategy for agent 1, i.e., $S_{ff}, q_0 \not\models \varphi_{\{1\}}$. Consider another question (property):* "Can agents 1 and 2 jointly from q_0, equipped with 4 units of electricity and 2 unit of water, make sure that the building is at least 90% safe?". *Similar to the previous question, we need to check if* $\varphi_{\{1,2\}} = \langle\langle\{1,2\}^{(4,2)}\rangle\rangle P_{\geq 0.9}\Diamond safe$ *is true at q_0. This is true, for example, when employing a strategy where both the agents perform sensing at q_0 and pumping the water at q_2. In fact, this strategy can guarantee the safety of the building by up to 99%. Hence, $S_{ff}, q_0 \models \varphi_{\{1,2\}}$.*

4 Model Checking

In this section, we present an algorithm for the model checking problem of pRB-ATL. In particular, given a pRCGS $S = (n, r, Q, \Pi, \pi, d, c, \delta)$ and a pRB-ATL formula φ, the algorithm produces the set of states $Sat(\varphi)$ of S that satisfy φ, i.e., $Sat(\varphi) = \{q \in Q \mid S, q \models \varphi\}$. Similar to ATL and its descendants, the algorithm generally processes φ recursively by computing the set of states satisfying sub-formulae of φ before combining them to produce $Sat(\varphi)$. For the propositional cases, the algorithm can be summarised as follows:

$$Sat(\top) = S, \qquad Sat(\varphi_1 \vee \varphi_2) = Sat(\varphi_1) \cup Sat(\varphi_2),$$
$$Sat(\neg\varphi) = S \setminus Sat(\varphi), \qquad Sat(p) = \{q \in Q \mid q \in \pi(p)\}.$$

Let us focus on the last cases $Sat(\langle\langle A^b\rangle\rangle P_{\bowtie v}[\psi])$ where $\psi = \bigcirc\varphi_1$ and $\psi = \varphi_1 \mathcal{U}^k \varphi_2$ with $k \in \mathbb{N} \cup \{\infty\}$. Notice that the case $Sat(\langle\langle A^b\rangle\rangle P_{\bowtie v}[\neg\psi])$ can be reduced to $Sat(\langle\langle A^b\rangle\rangle P_{\bowtie^{-1} 1-v}[\psi])$ due to Lemma 2. Instead of following the semantics definition, i.e., determining the existence of a b-strategy for A to achieve a certain probability v from a state s, we compute the min and max values over all possible b-strategies for A. In particular:

$$Pr_{S,q}^{\max}(A^b, \psi) = \sup_{b\text{-bounded } F_A} \inf_{F_{\bar{A}}} Pr_{S,q}^{F_A \cup F_{\bar{A}}}(\psi)$$
$$Pr_{S,q}^{\min}(A^b, \psi) = \inf_{b\text{-bounded } F_A} \sup_{F_{\bar{A}}} Pr_{S,q}^{F_A \cup F_{\bar{A}}}(\psi),$$

where $Pr_{S,q}^{F_A \cup F_{\bar{A}}}(\psi)$ is a shorthand for

$$Pr_{S,q}^{F_A \cup F_{\bar{A}}}(\{\lambda \in \Omega_{S,q} \mid S, \lambda \models \psi\}).$$

Then, computing states satisfying $\langle\langle A^b\rangle\rangle P_{\bowtie v}[\psi]$ is reduced to comparing these min/max values with v as follows:

$$Sat(\langle\langle A^b\rangle\rangle P_{\rhd v}[\psi]) = \{q \in Q \mid Pr_{S,q}^{\max}(A^b, \psi) \rhd v\} \quad (1)$$
$$Sat(\langle\langle A^b\rangle\rangle P_{\lhd v}[\psi]) = \{q \in Q \mid Pr_{S,q}^{\min}(A^b, \psi) \lhd v\}, \quad (2)$$

where $\lhd \in \{<, \leq\}$ and $\rhd \in \{>, \geq\}$. In the following, we show how to compute $Pr_{S,q}^{\max}(A^b, \psi)$ and $Pr_{S,q}^{\min}(A^b, \psi)$ recursively on the structure of ψ.

Case (4a): $\psi = \bigcirc\varphi_1$. Assume that $Sat(\varphi_1)$ has been computed. Then the maximal probability to arrive at a state in $Sat(\varphi_1)$ is obtained by players in A selecting an allowed move (costing at most b) to maximise the probability while

players outside A, i.e., in $\bar{A}$, select an arbitrary move to minimise it. Conversely, the minimal probability is obtained by players in A selecting an allowed move to minimise it while those outside select one to maximise it. Therefore, we have:

$$Pr_{S,q}^{\max}(A^b, \psi) = \max_{\substack{m \in D_A(q) \\ cost(q,m) \leq b}} \min_{m' \in D_{\bar{A}}(q)} \sum_{t \in Sat(\varphi_1)} \delta(q, m \cup m')(t)$$

$$Pr_{S,q}^{\min}(A^b, \psi) = \min_{\substack{m \in D_A(q) \\ cost(q,m) \leq b}} \max_{m' \in D_{\bar{A}}(q)} \sum_{t \in Sat(\varphi_1)} \delta(q, m \cup m')(t).$$

Case (4b): $\psi = \varphi_1 \mathcal{U}^{\leq k} \varphi_2$. Assume that $Sat(\varphi_1)$ and $Sat(\varphi_2)$ are computed. For convenience, we denote $Pr_{S,q}^{\max}(A^b, \varphi_1 \mathcal{U}^k \varphi_2)$ and $Pr_{S,q}^{\min}(A^b, \varphi_1 \mathcal{U}^k \varphi_2)$ by $X_{q,k}^b$ and $Y_{q,k}^b$, respectively. Then, there are three trivial sub-cases:

- $q \in Sat(\varphi_2)$: any computation from q satisfies ψ, hence $X_{q,k}^b = Y_{q,k}^b = 1$.
- $q \notin Sat(\varphi_1) \cup Sat(\varphi_2)$: any computation from q does not satisfy ψ, hence $X_{q,k}^b = Y_{q,k}^b = 0$.
- $q \notin Sat(\varphi_2)$ & $k = 0$: any computation from q does not satisfy ψ before 0 transition, hence $X_{q,k}^b = Y_{q,k}^b = 0$.

Otherwise, players in A try to choose an allowed move m from q with cost at most b that maximises the probability to arrive at a state that can satisfy ψ with the remaining resource $b' = b - cost(q, m)$ and within $k' = k - 1$ transitions. Formally, this can be defined as follows:

$$X_{q,k}^b = \max_{\substack{m \in D_A(q) \\ cost(q,m) \leq b}} \min_{m' \in D_{\bar{A}}(q)} \sum_{t \in Q} \delta(q, m \cup m')(t) \cdot X_{t,k-1}^{b-cost(q,m)}$$

$$Y_{q,k}^b = \min_{\substack{m \in D_A(q) \\ cost(q,m) \leq b}} \max_{m' \in D_{\bar{A}}(q)} \sum_{t \in Q} \delta(q, m \cup m')(t) \cdot Y_{t,k-1}^{b-cost(q,m)}.$$

Overall, one can form two linear equation systems with variables $X_{q,k}^b$ and $Y_{q,k}^b$, respectively. They can be solved by direct methods such as Gaussian elimination or iterative methods such as Jacobi and Gauss-Seidel [Kwiatkowska *et al.*, 2007]. In general, iterative methods suit the two linear equation systems best. It should not iterate more than $k + 1$ times as $X_{q,0}^b$ and $Y_{q,0}^b$ saturate to either 0 or 1 regardless of b and q by definition.

Case (4c): $\psi = \varphi_1 \mathcal{U} \varphi_2$. Assume that $Sat(\varphi_1)$ and $Sat(\varphi_2)$ are computed. Again for convenience, we denote $Pr_{S,q}^{\max}(A^b, \varphi_1 \mathcal{U} \varphi_2)$ and $Pr_{S,q}^{\min}(A^b, \varphi_1 \mathcal{U}' \varphi_2)$ by X_q^b and Y_q^b, respectively. We now have two trivial cases:

- If $q \in Sat(\varphi_2)$, any computation from q satisfies ψ, hence $X_q^b = Y_q^b = 1$.
- If $q \notin Sat(\varphi_1) \cup Sat(\varphi_2)$, any computation from q does not satisfy ψ, hence $X_q^b = Y_q^b = 0$.

Otherwise, players in A try to choose an allowed move m from q with cost at most b that maximises the probability to arrive at a state that can satisfy ψ with the remaining resource $b' = b - cost(q, m)$. Formally, this can be defined as follows:

$$X_q^b = \max_{\substack{m \in D_A(q) \\ cost(q,m) \leq b}} \min_{m' \in D_{\bar{A}}(q)} \sum_{t \in Q} \delta(q, m \cup m')(t) \cdot X_t^{b-cost(q,m)}$$

$$Y_q^b = \min_{\substack{m \in D_A(q) \\ cost(q,m) \leq b}} \max_{m' \in D_{\bar{A}}(q)} \sum_{t \in Q} \delta(q, m \cup m')(t) \cdot Y_t^{b-cost(q,m)}.$$

Then, one can form two linear equation systems with variables X_q^b and Y_q^b, respectively. Once again, they can be solved via existing methods.

Lemma 3. *Given a pRCGS S and a pRB-ATL formula φ, (i) $Sat(\varphi)$ terminates and (ii) $q \in Sat(\varphi)$ iff $S, q \models \varphi$.*

Sketch of proof. Since recursive calls within $Sat(\varphi)$ are always applied to strictly sub-formulas of φ, termination of (i) is straightforward. Correctness of (ii) can be proved by induction on the structure of φ. Consider $\varphi = \neg\varphi_1$, $q \in Sat(\varphi) \Leftrightarrow q \in S \setminus Sat(\varphi_1) \Leftrightarrow q \notin Sat(\varphi_1) \Leftrightarrow S, q \not\models \varphi_1$ (by I.H.) $\Leftrightarrow S, q \models \neg\varphi_1$. Other propositional cases ($\varphi = \top$; p; $\varphi_1 \vee \varphi_2$) are similar. Finally, from (1) and (2), the correctness of $Sat(\langle\!\langle A^b \rangle\!\rangle P_{\bowtie v}[\psi])$ follows from the correctness of computing $\mathsf{Pr}^{\max}_{S,q}(A^b, \psi)$ and $\mathsf{Pr}^{\min}_{S,q}(A^b, \psi)$. This is immediate from the arguments of the cases $(4a)$, $(4b)$ and $(4c)$. $\square$

Assuming that the natural numbers occurring in a pRB-ATL formula φ are encoded in unary, we have the following result.

Lemma 4. *The upper bound of the time complexity for $Sat(\varphi)$ is $O(|\varphi|^{3r+4} \cdot |S|^3)$.*

Sketch of proof. φ has at most $O(|\varphi|)$ sub-formulae. The cases $(4b)$ and $(4c)$ are the most computationally complex. In case $(4b)$, the numbers of b and k are bounded by $O(|\varphi|^r)$ and $O(|\varphi|)$, respectively. Therefore, the number of variables, also that of equations, in each corresponding linear equation system is bounded by $O(|\varphi|^r \cdot |S| \cdot |\varphi|) = O(|\varphi|^{r+1} \cdot |S|)$. Similarly, in case $(4c)$, the number of variables and also equations in each corresponding linear equation system is bounded by $O(|\varphi|^r \cdot |S|)$. It is well-known that the time complexity of solving such a linear equation system is at most $O(n^3)$ [Golub and Van Loan, 1996], where n is the number of equations. Therefore, the complexity of computing $Sat(\varphi)$ is $O(|\varphi| \cdot (|\varphi|^{r+1} \cdot |S|)^3) = O(|\varphi|^{3r+4} \cdot |S|^3)$. $\square$

Furthermore, the lower bound is given by that of the ATL, i.e., linear to the size of the input model and the input formula.

Example 3. *Let us illustrate the use of the pRB-ATL and quantitatively verify the system presented in Example 1 via model checking. We generalise the properties described in Example 2 as follows "Can a coalition from q_0, equipped with e units of electricity and w units of water, make sure that the building is at least v safe?". This is formalised in pRB-ATL by $\varphi_A = \langle\!\langle A^{(e,w)} \rangle\!\rangle P_{\geq v} \diamond safe$. To this end, we need to check whether $S_{ff}, q_0 \models \varphi_A$. As mentioned in the previous section, this can be reduced to determine the maximal probability:*

$$\mathsf{Pr}^{\max}_{S_{ff}, q_0}(A^{(e,w)}, \diamond safe) = X^{(e,w)}_{q_0}.$$

Intuitively, the best strategy for one agent to help the building to be safe is to sense and then to pump the water. In total, this costs two units of electricity and one unit of water. Similarly, while cooperating, the best strategy for both the agents would be to choose their best strategies concurrently, which will cost together four units of electricity and two units of water. Therefore, for $A = \{1\}$, we consider the resources bounded by $(e,w) \leq (2,1)$ and for $A = \{1,2\}$ those are bounded by $(e,w) \leq (4,2)$. For each case of A and (e,w), the corresponding equation system from case $(4c)$ can be solved by, for example, the iterative method as applied in [Kwiatkowska et al., 2007]. The solutions of these linear

w/e	0	1	2
0	0.0	0.0	0.0
1	0.0	0.0	0.8

Table 1: $\mathsf{Pr}^{\max}_{S_{ff}, q_0}(\{1\}^{(e,w)}, \diamond safe)$

w/e	0	1	2	3	4
0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.8	0.9	0.9
2	0.0	0.0	0.8	0.9	0.99

Table 2: $\mathsf{Pr}^{\max}_{S_{ff}, q_0}(\{1,2\}^{(e,w)}, \diamond safe)$

equation systems are summarised in Table 1 for $A = \{1\}$ and in Table 2 for $A = \{1,2\}$. In particular, Table 1 shows that any resource bound less than $(2,1)$ is not helpful for agent 1 as it has no strategy to make sure that the building is safe. In the best case, with resource bound $(2,1)$, the only vital strategy is to sense the fire and then pump the water. In this case, since agent 2 is not required to cooperate, the worst case is to end up in q_1 from q_0 where the chance to arrive at q_3, the safe state, is at least 80%. That is, choosing the following actions: $(2,1)$ in q_0, $(2,1)$ in q_1 and $(1,1)$ in q_3. Similarly, Table 2 shows that any resource bound less than $(2,1)$ will not be enough for both the agents while cooperating. However, the chance of making the building safe increases to 90% as more and more resources are given. This is because from q_0 both the agents can force the arrival at q_2 instead of q_1, where there are at least three units of electricity are available at q_0. Eventually, the maximal chance of making the building safe reaches 99% as both the agents have enough resources to follow the same best strategy. That is, choosing the following actions: $(2,2)$ in q_0, $(2,2)$ in q_2 and $(1,1)$ in q_3. Due to the small size of the illustrative example system, the time taken for the verification was less than 1 second.

5 Conclusions and Future Work

In this paper, we proposed a logic pRB-ATL for reasoning about coalitional abilities of systems under resource constraints that exhibit both probabilistic and non-deterministic behaviour, and provided a standard model-checking algorithm for it. pRB-ATL allows us to express various interesting properties of coalitions of agents involving resource limitations and probabilistic behaviour. For example, it can express that "*a coalition of agents A has a strategy to achieve a property φ with probability $\bowtie v$ ($\bowtie \in \{<, \leq, \geq, >\}$) provided they have resources b, but they cannot enforce φ under a tighter resource bound b' with probability $\bowtie v$*".

In future work, we would like to explore the real world applicability of the proposed logic as an underlying solution framework for more complex analyses of resource-bounded systems. The example scenario presented in this paper is a very simple case. If we model a more realistic scenario and increase the problem size, the analysis and verification task would be hard to do by hand. Therefore, it would be more convenient to use an automatic method to encode and verify them, for example using a model checking tool [Kwiatkowska *et al.*, 2002].

References

[Abbas *et al.*, 2015] Waseem Abbas, Aron Laszka, Yevgeniy Vorobeychik, and Xenofon D. Koutsoukos. Scheduling Intrusion Detection Systems in Resource-Bounded Cyber-Physical Systems. In *Proceedings of the 1st CPS-SPC*, pages 55–66, Denver, Colorado, USA, 2015. ACM Press.

[Ågotnes *et al.*, 2009] Thomas Ågotnes, Wiebe van der Hoek, and Michael J. Wooldridge. Reasoning about coalitional games. *Artificial Intelligence*, 173(1):45–79, 2009.

[Alechina *et al.*, 2010] Natasha Alechina, Brian Logan, Hoang Nga Nguyen, and Abdur Rakib. Resource-bounded Alternating-time Temporal Logic. In *Proceedings of the 9th AAMAS: Volume 1*, pages 481–488, Toronto, Canada, 2010.

[Alur *et al.*, 2002] Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49(5):672–713, September 2002.

[Billingsley, 2012] Patrick Billingsley. *Probability and Measure*. Wiley, 2012.

[Bulling and Farwer, 2010] Nils Bulling and Berndt Farwer. Expressing Properties of Resource-Bounded Systems: RTL* and RTL. In J. Dix, M. Fisher, and P. Novák, editors, *CLIMA*, volume 6214, pages 22–45. Springer, 2010.

[Calvaresi *et al.*, 2017] Davide Calvaresi, Mauro Marinoni, Arnon Sturm, Michael Schumacher, and Giorgio C. Buttazzo. The challenge of real-time multi-agent systems for enabling IoT and CPS. In *Proceedings of WI '17*, pages 356–364, Leipzig, Germany, 2017. ACM Press.

[Chen *et al.*, 2013] Taolue Chen, Vojtech Forejt, Marta Z. Kwiatkowska, David Parker, and Aistis Simaitis. Automatic verification of competitive stochastic systems. *Formal Methods in System Design*, 43(1):61–92, August 2013.

[Faza *et al.*, 2009] Ayman Z. Faza, Sahra Sedigh, and Bruce M. McMillin. Reliability Analysis for the Advanced Electric Power Grid: From Cyber Control and Communication to Physical Manifestations of Failure. In B. Buth, G. Rabe, and T. Seyfarth, editors, *Computer Safety, Reliability, and Security*, volume 5775, pages 257–269. Springer, 2009.

[Forejt *et al.*, 2011] Vojtech Forejt, Marta Z. Kwiatkowska, Gethin Norman, and David Parker. Automated Verification Techniques for Probabilistic Systems. In M. Bernardo and V. Issarny, editors, *Formal Methods for Eternal Networked Software Systems*, volume 6659, pages 53–113. Springer, 2011.

[Golub and Van Loan, 1996] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, 3rd ed edition, 1996.

[Goranko, 2001] Valentin Goranko. Coalition Games and Alternating Temporal Logics. In *Proceedings of the 8th TARK*, pages 259–272, Siena, Italy, 2001. Morgan Kaufmann.

[Huang *et al.*, 2012] Xiaowei Huang, Kaile Su, and Chenyi Zhang. Probabilistic Alternating-Time Temporal Logic of Incomplete Information and Synchronous Perfect Recall. In *Proceedings of the 26th AAAI*, Toronto, Canada, 2012.

[Kwiatkowska *et al.*, 2002] Marta Z. Kwiatkowska, Gethin Norman, and David Parker. PRISM: Probabilistic Symbolic Model Checker. In G. Goos, J. Hartmanis, J. van Leeuwen, T. Field, P. G. Harrison, J. Bradley, and U. Harder, editors, *Comp. Performance Evaluation: Modelling Techniques & Tools*, volume 2324, pages 200–204. Springer, 2002.

[Kwiatkowska *et al.*, 2007] Marta Z. Kwiatkowska, Gethin Norman, and David Parker. Stochastic Model Checking. In M. Bernardo and J. Hillston, editors, *Formal Methods for Performance Evaluation*, volume 4486, pages 220–270. Springer, 2007.

[Kwiatkowska, 2016] Marta Kwiatkowska. Advances and challenges of quantitative verification and synthesis for cyber-physical systems. In *SOSCYPS*, pages 1–5, Vienna, Austria, April 2016. IEEE.

[Laszka *et al.*, 2015] Aron Laszka, Yevgeniy Vorobeychik, and Xenofon D. Koutsoukos. Integrity assurance in resource-bounded systems through stochastic message authentication. In *Proceedings of HotSoS*, pages 1–12, Urbana, Illinois, 2015. ACM Press.

[Leitão *et al.*, 2016] Paulo Leitão, Stamatis Karnouskos, Luis Ribeiro, Jay Lee, Thomas I. Strasser, and Armando W. Colombo. Smart Agents in Industrial Cyber–Physical Systems. *Proceedings of the IEEE*, 104(5):1086–1101, May 2016.

[Monica *et al.*, 2011] Dario Della Monica, Margherita Napoli, and Mimmo Parente. On a Logic for Coalitional Games with Priced-Resource Agents. *ENTCS*, 278:215–228, 2011.

[Pauly, 2002] Marc Pauly. A Modal Logic for Coalitional Power in Games. *Journal of Logic and Computation*, 12(1):149–166, February 2002.

[Wooldridge and Dunne, 2004] Michael J. Wooldridge and Paul E. Dunne. On the computational complexity of qualitative coalitional games. *Arti. Intelligence*, 158(1):27–73, 2004.

[Zhang *et al.*, 2016] Man Zhang, Bran Selic, Shaukat Ali, Tao Yue, Oscar Okariz, and Roland Norgren. Understanding Uncertainty in Cyber-Physical Systems: A Conceptual Model. In A. Wasowski and H. Lönn, editors, *Modelling Foundations and Applications*, volume 9764, pages 247–264. Springer International Publishing, 2016.