Reasoning about Resource-bounded Agents

Natasha Alechina
joint work with Brian Logan, Hoang Nga Nguyen, Franco Raimondi, Nils Bulling

Agent Verification Workshop
Liverpool, 11 September 2015

Acknowledgement

This work is funded by the EPSRC project(s)

Verification of resource-bounded multiagent systems

(joint between the University of Nottingham and Middlesex University)

Plan of the talk

- motivation: why reason about resources?
- resource logics
- decidability and undecidability of the model-checking problem for resource logics
- decidable case (RB+-ATL)
- feasible cases (no production, or one resource)
- case study (sensor network protocol)

Motivating examples

- sensor networks: nodes can only send and receive messages if they have sufficient *energy* levels
- mobile agents, for example patrolling robots: also need energy to move
- agents may need other resources for performing actions, for example money, fuel, or water (for extinguishing fires), etc.

Resource Logics

- variants of Alternating-Time Temporal Logic (ATL) where transitions have costs (or rewards) and the syntax can express resource requirements of a strategy, e.g.:

  agents A can enforce outcome φ if they have at most b_1 units of resource r_1 and b_2 units of resource r_2

- various flavours of resource logics exist: RBCL (IJCAI 2009), RB-ATL (AAMAS 2010), RB$\pm$ATL (ECAI 2014), RAL (Bulling & Farwer), PRB-ATL (Della Monica et al.), QATL* (Bulling & Goranko)

Model-checking resource logics

- model-checking problem: given a structure, a state in the structure and a formula, does the state satisfy the formula?
- for most resource logics the model-checking problem is undecidable: in particular, various flavours of RAL, and QATL*

Resource Agent Logic (Bulling & Farwer 2010)

RAL formulae are defined by:

$$\phi ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid \langle\langle A \rangle\rangle_B^{\downarrow} \bigcirc \varphi \mid \langle\langle A \rangle\rangle_B^{\eta} \bigcirc \varphi \mid \langle\langle A \rangle\rangle_B^{\downarrow} \varphi \mathcal{U} \psi \mid \langle\langle A \rangle\rangle_B^{\eta} \varphi \mathcal{U} \psi \mid \langle\langle A \rangle\rangle_B^{\downarrow} \Box \varphi \mid \langle\langle A \rangle\rangle_B^{\eta} \Box \varphi$$

where p is a proposition, $A, B \subseteq Agt$ are sets of agents, and η is a resource endowment

$\langle\langle A \rangle\rangle_B^{\eta} \varphi$ means that *agents A have a strategy compatible with the endowment* η *to enforce* φ whatever the opponent agents do (opponents in B also act under resource bound η)

$\langle\langle A \rangle\rangle_B^{\downarrow} \varphi$ means that *agents A have a strategy compatible with the current resource endowment to enforce* φ whatever the opponent agents do (opponents in B also act under the current resource bound)

RAL fragments

rfRAL in *resource flat RAL*, each nested ATL operator has a fresh assignment of resources ($\langle\langle A\rangle\rangle_B^{\downarrow}\varphi$ is not allowed):

$$\langle\langle A\rangle\rangle_A^{\eta_0}(\mathit{safe}\,\mathcal{U}\,(\langle\langle A\rangle\rangle_A^{\eta_1}(\mathit{visual}\,\mathcal{U}\,\mathit{rescue})))$$

prRAL in *proponent restricted RAL*, only the strategy of the proponent agents is resource bounded — the opponent agents have no resource bound $\langle\langle A\rangle\rangle^{\eta}\varphi$, $\langle\langle A\rangle\rangle^{\downarrow}\varphi$

rfprRAL in *resource flat proponent restricted RAL* is the combination of rfRAL and prRAL

prRAL^{r} *positive proponent restricted RAL* is the same as prRAL except that no coalition modality is under the scope of a negation

Summary of known results (IJCAI 2015)

Models	RAL	rfRAL	prRAL	rfprRAL	$prRAL^r$
RBM	U [1]	U [1]	U [1]	U [1]	U [1]*
iRBM	U [1]*	U	U [1]*	D [2]*	D

RBM Resource Bounded Models (infinite semantics)
iRBM Resource Bounded Models with *idle* actions

[1] Bulling & Farwer 2010
[2] Alechina et al 2014 (* corollary)

Decidable case: RB±ATL

RB±ATL: syntax

- $Agt = \{a_1, \ldots, a_n\}$ a set of n agents
- $Res = \{res_1, \ldots, res_r\}$ a set of r resources,
- Π a set of propositions
- $B = \mathbb{N}^r_\infty$ a set of resource bounds, where $\mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$

RB±ATL: syntax

Formulas of RB±ATL are defined by the following syntax

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \psi \mid \langle\langle A^b \rangle\rangle \bigcirc \varphi \mid \langle\langle A^b \rangle\rangle \varphi \mathcal{U} \psi \mid \langle\langle A^b \rangle\rangle \Box \varphi$$

where $p \in \Pi$ is a proposition, $A \subseteq Agt$, and $b \in B$ is a resource bound.

RB±ATL: meaning of formulas

- $\langle\langle A^b \rangle\rangle \bigcirc \psi$ means that a coalition *A* can ensure that the next state satisfies φ under resource bound *b*
- $\langle\langle A^b \rangle\rangle \psi_1 \, \mathcal{U} \, \psi_2$ means that *A* has a strategy to enforce ψ while maintaining the truth of φ, and the cost of this strategy is at most *b*
- $\langle\langle A^b \rangle\rangle \Box \psi$ means that *A* has a strategy to make sure that φ is always true, and the cost of this strategy is at most *b*

Resource-bounded concurrent game structure

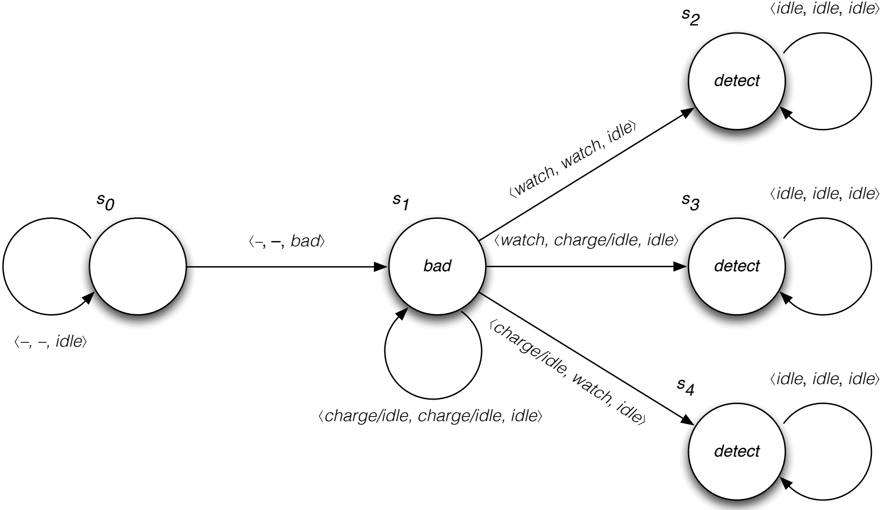
A RB-CGS is a tuple $M = (Agt, Res, S, \Pi, \pi, Act, d, c, \delta)$ where:

- Agt is a non-empty set of n agents, Res is a non-empty set of r resources and S is a non-empty set of states;
- Π is a finite set of propositional variables and $\pi : \Pi \to \wp(S)$ is a truth assignment
- Act is a non-empty set of actions which includes *idle*, and $d : S \times Agt \to \wp(Act) \setminus \{\emptyset\}$ is a function which assigns to each $s \in S$ a non-empty set of actions available to each agent $a \in Agt$
- $c : S \times Agt \times Act \to \mathbb{Z}^r$ (the integer in position i indicates consumption or production of resource res_i by the action a)
- $\delta : (s, \sigma) \mapsto S$ for every $s \in S$ and joint action $\sigma \in D(s)$ gives the state resulting from executing σ in s.

Additional assumptions and notation

- for every $s \in S$ and $a \in Agt$, $idle \in d(s, a)$
- $c(s, a, idle) = \bar{0}$ for all $s \in S$ and $a \in Agt$ where $\bar{0} = 0^r$
- we denote joint actions by all agents in Agt available at s by $D(s) = d(s, a_1) \times \cdots \times d(s, a_n)$
- for a coalition A, $D_A(s)$ is the set of all joint actions by agents in A
- $out(s, \sigma) = \{s' \in S \mid \exists \sigma' \in D(s) : \sigma = \sigma'_A \wedge s' = \delta(s, \sigma')\}$
- $cost(s, \sigma) = \sum_{a \in A} c(s, a, \sigma_a)$
- if one agent consumes 10 units of resource and another agent produces 10 units of resource, the cost of their joint action is 0

Example: c(-,-,idle)=0, c(-,-,watch)=1, c(-,-,charge)=-1



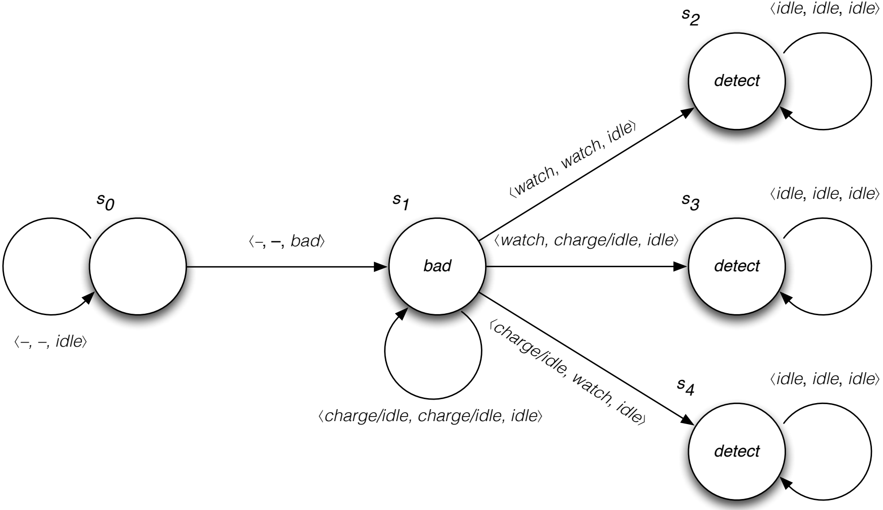
Strategies and their costs

- a *strategy for a coalition* $A \subseteq Agt$ is a mapping $F_A : S^+ \to Act$ such that, for every $\lambda s \in S^+$, $F_A(\lambda s) \in D_A(s)$
- a computation $\lambda \in S^\omega$ is consistent with a strategy F_A iff, for all $i \geq 0$, $\lambda[i+1] \in out(\lambda[i], F_A(\lambda[0,i]))$
- $out(s, F_A)$ the set of all consistent computations λ of F_A that start from s
- given a bound $b \in B$, a computation $\lambda \in out(s, F_A)$ is b-consistent with F_A iff, for every $i \geq 0$, $\sum_{j=0}^{i} cost(\lambda[j], F_A(\lambda[0,j])) \leq b$
- F_A is a b-strategy if all $\lambda \in out(s, F_A)$ are b-consistent

Truth definition

- $M, s \models \langle\langle A^b \rangle\rangle \bigcirc \phi$ iff $\exists$ b-strategy F_A such that for all $\lambda \in out(s, F_A)$: $M, \lambda[1] \models \phi$
- $M, s \models \langle\langle A^b \rangle\rangle \phi \mathcal{U} \psi$ iff $\exists$ b-strategy F_A such that for all $\lambda \in out(s, F_A)$, $\exists i \geq 0$: $M, \lambda[i] \models \psi$ and $M, \lambda[j] \models \phi$ for all $j \in \{0, \ldots, i-1\}$
- $M, s \models \langle\langle A^b \rangle\rangle \Box \phi$ iff $\exists$ b-strategy F_A such that for all $\lambda \in out(s, F_A)$ and $i \geq 0$: $M, \lambda[i] \models \phi$

Example: $\langle\langle\{1,2\}^0\rangle\rangle\Box(bad \rightarrow \langle\langle\{1,2\}^0\rangle\rangle\bigcirc detect)$



Infinite bound versions

Since the infinite resource bound version of RB-ATL modalities correspond to the standard ATL modalities, we write

- $\langle\langle A^{\bar{\infty}} \rangle\rangle \bigcirc \phi$ as $\langle\langle A \rangle\rangle \bigcirc \phi$
- $\langle\langle A^{\bar{\infty}} \rangle\rangle \phi \mathcal{U} \psi$ as $\langle\langle A \rangle\rangle \phi \mathcal{U} \psi$
- $\langle\langle A^{\bar{\infty}} \rangle\rangle \Box \phi$ as $\langle\langle A \rangle\rangle \Box \phi$

Model-checking RB±ATL

The model-checking problem for RB±ATL is the question whether, for a given RB-CGS structure M, a state s in M and an RB±ATL formula ϕ, $M, s \models \phi$.

Theorem (Alechina, Logan, Nguyen, Raimondi 2014):

The model-checking problem for RB±ATL is decidable

Complexity

- the model-checking problem for RB$\pm$ATL is EXPSPACE-hard
- model-checking problem for RB$\pm$ATL with one resource type is in PSPACE
- no production (RB-ATL): exponential in resources, but polynomial in the model and the formula

Feasible Cases

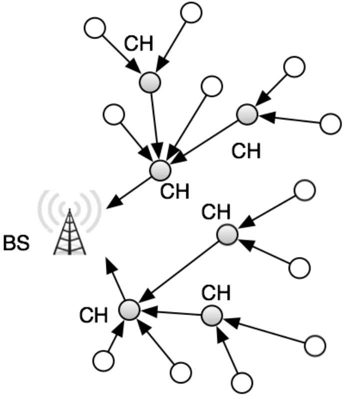
Feasible cases

- model-checking problem for RB$\pm$ATL with one resource type is in PSPACE
- symbolic model-checking for 1-RB$\pm$ATL is implemented in MCMAS (IJCAI 2015)
- no production (RB-ATL): exponential in resources, but polynomial in the model and the formula
- symbolic model-checking for RB-ATL implemented in MCMAS (AAMAS 2015 poster)

Case study

- energy consumption in a sensor network running LEACH protocol (we collaborated with Leonardo Mostarda from SENSOLAB at Middlesex University)
- model-checking uses RB-ATL with one resource (energy)
- can verify how long the network can function with a given amount of energy per node before at least one node dies

LEACH protocol



LEACH study results

Verification of $\langle\langle A1 \rangle\rangle^{80}$ *true* $\mathcal{U}$ *Completed*
(agent $A1$, closest to the base, can complete all rounds of the protocol in a given network configuration within an energy bound of 80).

Degree	Depth	Cluster size	Iterations	Net. Life (days)	Result
2	2	3	5	15	True
2	2	3	7	21	True
2	2	3	9	27	True
2	2	3	11	33	True
2	2	3	13	39	False
2	2	3	15	45	False

Future work

- using MCMAS with resources for more case studies
  Suggestions of case studies welcome!
- implement more variants of resource logics:
  - explicit flag for whether agents can pool resources (assumed in RB-ATL and RB$\pm$ATL, and but not natural for sensor networks)
  - different combination rules for resources (we use addition, but for example time is different)
  - add shared resources