Towards a Tool for LTL Synthesis with Bounded-Energy Constraints

Bastien Maubert[1], Aniello Murano[1], Pasquale Perillo[1], Sasha Rubin[2], and Alessio Spasiano[1]

[1] University of Naples Federico II, Italy
[2] University of Sydney, Australia

Abstract. MCMAS is an open-source model checker specialised in logics for multi-agent systems (MAS), in particular logics for strategic reasoning, but which at the moment can only handle qualitative objectives. We propose a first step towards extending MCMAS to certain quantitative strategic logics where the quantitative aspects have the form of energy constraints. We show that the existence of strategies for a coalition of agents for objectives given as a combination of LTL formula and bounded energy constraint can be reduced to the same problem for pure LTL objectives. Using this reduction, we run some experiments with MCMAS to solve quantitative strategic problems.

1 Introduction

Multi-agent systems (MAS) are systems that involve several autonomous entities, called agents, acting in some environment [6] and trying to achieve some individual and/or common goals against adversarial entities, possibly by cooperating. One fundamental problem in the study of MAS is to decide the existence of and synthesize *strategies* for coalitions of agents to achieve their goals. Several logical formalisms have been developed to express such strategic abilities in MAS, extending the Linear-time Temporal Logic LTL, and several tools have been developed. Among them, MCMAS (Model Checker for Multi-agent Systems) [7] can handle a number of such formalisms, based on ATL (Alternating-time Temporal Logic) [1] or SL (Strategy Logic) [4, 8, 9]. MCMAS can thus solve, in particular, the strategy existence problem for LTL objectives, when we restrict attention to *memoryless agents*, i.e., agents whose strategies depend only on the current state of the system, and not on the past. LTL, though very successful as a language to specify properties of systems' behaviours, is purely qualitative. But in increasingly widespread contexts such as resource-constrained environments, it is desirable to also consider quantitative objectives. In this paper we consider strategy synthesis for objectives combining a qualitative component expressed in LTL, and a quantitative *energy* condition [3, 5], given by integer weights assigned to agents' actions and whose sum through time should always remain in a given range. MCMAS can not handle such quantitative objectives, but we show how when the interval is bounded, the strategy problem for LTL + energy condition can be reduced to the pure LTL case, and we can thus use MCMAS to solve it.

2 Formal setting

In this section we recall interpreted systems, LTL, energy conditions and we finally define the strategic problem we study.

Interpreted Systems. *Interpreted Systems* (IS) are a classic class of models for MAS, and are the models used in MCMAS [7]. Let $Agt = \{0, 1, \cdots, n\}$ be a finite, non-empty set of agent names where agent 0 is called the *environment agent*, and AP a set of atomic propositions.

Definition 1. *An Interpreted System $\mathcal{I}$ is a tuple $\langle (L_i, ACT_i, P_i, \tau_i)_{i \in Agt}, \lambda \rangle$ where: L_i is a finite set of* local states *for agent $i \in Agt$, ACT_i is a finite, non-empty set of* actions *that agent i can perform, $P_i \colon L_i \to 2^{ACT_i} \setminus \{\emptyset\}$ is the* protocol *for agent i that denotes the set of actions available in each local state; $\tau_i : L_i \times ACT_0 \times ACT_1 \times \ldots \times ACT_n \to L_i$ is the* transition function *of agent i, and $\lambda : L_0 \times L_1 \times \ldots \times L_n \to 2^{AP}$ is a labelling function encoding for each state which atomic propositions are true.*

$S = L_0 \times L_1 \times \ldots \times L_n$ is the set of *global states* and $Dc = ACT_0 \times \ldots \times ACT_n$ is the set of possible *joint actions*, representing a choice of action for each agent. Each agent's transition function τ_i maps joint actions and local states to resulting local states after execution of the joint action. We let $l_i(s)$ be the local state of agent i in the global state $s \in S$ and we define the *global transition function* $\tau : S \times Dc \to S$ as $\tau(s, \alpha) = s'$ where $l_i(s') = \tau_i(l_i(s), \alpha)$ for each $i \in Agt$.

A *computation* is an infinite sequence $s_1 \alpha_1 s_2 \alpha_2 \ldots$ that alternates between global states and joint actions such that $\tau(s_k, \alpha_k) = s_{k+1}$ for every $k \geq 1$. The k-th state of a computation c with $k \in \mathbb{N}^*$ is denoted by c_k and the suffix of a computation starting from $k \in \mathbb{N}^*$ is denoted by $c_{\geq k}$.

Linear-time Temporal Logic. We briefly recall the syntax and semantics of LTL. Formulas are built according to the grammar $\varphi ::= p \mid \varphi \vee \varphi \mid \neg \varphi \mid X \varphi \mid \varphi \, U \, \varphi$, where $p \in AP$. Other classic operators can be defined in the usual way, e.g., *true* $\top$ (defined as $p \vee \neg p$), *finally* F ($F\varphi$ stands for $\top U \varphi$) and *globally* G ($G\varphi$ stands for $\neg F \neg \varphi$). The semantics of a LTL formula φ on an infinite sequence $u = u_1 u_2 \ldots \in (2^{AP})^\omega$ of valuations over AP is defined as follows: boolean cases are as usual, $u \models X\varphi$ if $u_{\geq 2} \models \varphi$, and $u \models \varphi_1 \, U \, \varphi_2$ if there exists $k \geq 1$ such that $u_{\geq k} \models \varphi_2$ and $u_{\geq j} \models \varphi_1$ for all $1 \leq j < k$.

For an Interpreted System $\mathcal{I}$, a computation $c = s_1 \alpha_1 s_2 \alpha_2 \ldots$ over $\mathcal{I}$ and a LTL formula φ, we write $\mathcal{I}, c \models \varphi$ to mean that $\lambda(s_1)\lambda(s_2) \ldots \models \varphi$.

Energy Constraints. An *energy condition* (over an *IS* $\mathcal{I}$) is a triple $e = \langle w, \varepsilon^{init}, [a, b] \rangle$ where $w : Dc \to \mathbb{Z}$ is a *weight function*, $\varepsilon^{init} \in [a, b]$ is the *initial energy level* and $[a, b]$ is its *energy bound*, with $a, b \in \mathbb{Z}$ and $a \leq b$. Given a computation $c = s_1 \alpha_1 s_2 \alpha_2 \cdots$, the *energy values* ε induced by c is the following infinite sequence of integers: $\varepsilon_1 = \varepsilon^{init}$, and for $k > 1$, $\varepsilon_k = \varepsilon_{k-1} + w(\alpha_{k-1})$.

A computation c over an *IS* $\mathcal{I}$ *satisfies* an energy condition $e = \langle w, \varepsilon^{init}, [a,b] \rangle$ if the energy values ε induced by c with weight assignment w and $\varepsilon_1 = \varepsilon^{init}$ are such that $\varepsilon_k \in [a,b]$ for $k > 1$. In this case, we write $\mathcal{I}, c \models e$.

Remark 1. In [5], which we take inspiration from, the weight function depends on global states and joint actions. Here it only depends on joint actions, which is also reasonable. We make this choice to make it possible to simulate energy thanks to an *energy agent* (see next section). Indeed, in interpreted systems agents can see joint actions but not other agents's local states.

Problem definition. We want to determine if a team of agents has a way to choose actions so that the resulting behaviour of the system satisfies a given LTL formula and energy constraint. In other words we want to know if the agents have a *winning strategy*. We now define this notion.

A *local strategy* for agent i is a function $\sigma_i : L_i \to ACT_i$ such that $\sigma_i(l) \in P_i(l)$ for every $l \in L_i$. Note that strategies are uniform (they only depend on the information available to the agent) and memoryless (they do not depend on the history, only on the current state). A computation $c = s_1\alpha_1 s_2\alpha_2 \ldots$ is *consistent* with a strategy σ_i for agent i if for every $k \geq 1$ it is true that $\sigma_i(l_i(s_k)) = ac_i(\alpha_k)$, where $ac_i(\alpha_k)$ is the action of agent i in α_k. A *strategy profile* $\sigma_A = \{\sigma_i\}_{i \in A}$ for a set $A \subseteq Agt$ of agents is a set of strategies, one for each agent in A. Given a global state s, a LTL formula φ and an energy condition e, we say that a strategy profile σ_A for A is *winning* if for every computation c that starts from s and that is consistent with each σ_i for $i \in A$, it holds that $\mathcal{I}, c \models \phi$ and $\mathcal{I}, c \models e$.

Problem. *Given an Interpreted System $\mathcal{I}$, a team of agents $A \subseteq Agt$, a global state $s \in S$, a LTL formula φ and an energy condition $e = \langle w, \varepsilon^{init}, [a,b] \rangle$, determine whether there exists a winning strategy profile for the agents A.*

3 Problem Reduction

MCMAS can deal with LTL objectives, but not energy constraints. In this section we show how to reduce our problem to the same problem with only a LTL objective, so that we can use MCMAS to solve it.

Fix an IS $\mathcal{I}$, a global state s, a LTL formula φ and an energy constraint $e = \langle w, \varepsilon^{init}, [a,b] \rangle$. Intuitively, the transformation consists in adding in $\mathcal{I}$ a new *energy agent*, whose only task is to track the current energy in the evolution of the system. We introduce a fresh atomic proposition e-ok that is meant to indicate that the current energy level is in the authorized range $[a,b]$. We then can express that the energy condition is satisfied with the LTL formula $\phi_e = G\,\text{e-ok}$, and it remains to take as new objective the LTL formula $\phi \wedge \phi_e$.

We define the energy agent $(L_e, ACT_e, P_e, \tau_e)$ as follows. First $L_e = \{a, a+1, \ldots, b-1, b, O\}$: we have one local state for each energy level in $[a,b]$, plus one local state O denoting that the energy is out of range. Second, the agent has only one dummy action available in every local state: $ACT_e = \{\text{idle}\}$, and $P_e(l) = ACT_e$ for each $l \in L_e$. The transition function τ_e updates the local state according to the weight assignment of the energy constraint:

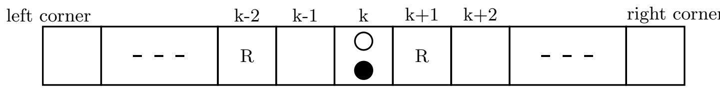

Fig. 1. Representation of the corridor system with two agents and two recharge cells

– $\tau_e(l_e, \alpha) = l_e + w(\alpha)$ if $a \leq l_e + w(\alpha) \leq b$ and $l_e \neq O$;
– $\tau_e(l_e, \alpha) = O$ if $l_e + w(\alpha) < a$ or $l_e + w(\alpha) > b$ or $l_e = O$.

Writing $\mathcal{I} = \langle (L_i, ACT_i, P_i, \tau_i)_{i \in Agt}, \lambda \rangle$, we define the new interpreted system $\mathcal{I}' = \langle (L_i, ACT_i, P_i, \tau_i)_{i \in Agt \cup \{e\}}, \lambda' \rangle$ over agents $Agt' = Agt \cup \{e\}$ and atomic propositions $AP' = AP \cup \{\text{e-ok}\}$ as follows. $(L_e, ACT_e, P_e, \tau_e)$ is as defined above, and $\lambda' : S \times L_e \to 2^{AP'}$ is such that $\lambda'(s, l_e) = \lambda(s) \cup \{\text{e-ok}\}$ if $l_e \neq O$, and $\lambda(s)$ otherwise. This means that e-ok holds if the current state of the energy agent correspondents to an energy level in the acceptable range.

The LTL formula $\varphi_e = G\,\text{e-ok}$ means that the energy is always in the range $[a, b]$. Clearly, for every computation c, $\mathcal{I}, c \models e$ if, and only if, $\mathcal{I}', c \models \varphi_e$.

It follows that we can reduce our problem to a problem of strategy synthesis for pure LTL objectives:

Theorem 1. *There exists a winning strategy profile for A in $\mathcal{I}$ for LTL objective φ and energy constraint e if, and only if, there exists a winning strategy profile for A in $\mathcal{I}'$ for LTL objective $\varphi \wedge \varphi_e$.*

4 MCMAS Experimental Results

Resorting Theorem 1, we can solve the problem of strategy synthesis for LTL and energy constraint in MCMAS. More precisely we used MCMAS_{SLK}, an extension of MCMAS that adds support to Strategy Logic with Knowledge, and supports reasoning about memoryless strategies with imperfect information [2].

Corridor example. We designed a simple example called *corridor*, in which one or more players move in a hallway divided in a finite number of cells. The players can perform actions *left* or *right*, each associated with an energy weight of -1, meaning that they cost one unit of energy to. In addition, the corridor contains a number of recharge cells, where the players can gain a certain number of energy units by performing a *recharge* action. Each player is an agent, who tracks his actual position in his local state. We also have a special agent for the environment (Env) who stores the position of the recharge cells in his local state, which is observable to all agents. Figure 1 represents an instance of corridor with two recharge cells marked as R and black and white circles representing two players. The goal is to find if there exists a strategy for each player such that both extremities of the corridor are visited infinitely often, while maintaining the energy in a given range. We have two atomic propositions *le* and *re* to express that some agent is currently at the left (resp. right) extremity of the corridor, so that visiting infinitely often both extremities can be expressed by the LTL

	without energy			**with energy**		
players number	**reachable states**	**mean time** (s)	**standard deviation**	**reachable states**	**mean time** (s)	**standard deviation**
1	12	0,0635	0,0695	122	0,054	0,0309
2	144	0,21	0,0946	1084	145,156	1,1707

Table 1. Experimental results for the corridor

formula $\phi = G(F\ le \wedge F\ re)$. Using Theorem 1, we know that solving the problem with LTL objective ϕ and an energy constraint is equivalent to solving it for the LTL objective $\phi \wedge G\,$e-ok in the system $\mathcal{I}'$ enriched with the energy agent Ener.

Thus we check in MCMAS$_{SLK}$, for two players a_1 and a_2, the formula

$$\langle s_1\rangle\, \langle s_2\rangle\, [s_3][s_4](a_1, s_1)(a_2, s_2)(\mathrm{Env}, s_3)(\mathrm{Ener}, s_4)\, G(F\ le \wedge F\ re \wedge \text{e-ok}).$$

MCMAS returns that the formula is false because a memoryless winning strategy for the two players to achieve both objectives does not exist, indeed with memoryless strategies, each player in each cell has to either always go left or always go right, so that they cannot reach both extremities. Similarly, they cannot recharge and exit from a recharge cell at different times, so it is not possible to remain globally in the energy range.

To make the formula true we add two elements to player agents: first a variable called *last_dir* with possible values *left* and *right* that saves the last chosen direction of the player; second change *recharge* with two actions *recharge_left* and *recharge_right* to allow the player to decide to recharge *and* move away.

We report in Table 1 statistics of two implementations, with one and with two players, with 7 cells, without and with energy. The one player version starts from position 4 and has one recharge cell in 3 and energy condition $e = \langle w, 13, [2, 13]\rangle$, and a recharge action adds +11. The second version starts from position 4 for both players and has two recharge cells in 3 and 5, the energy condition is $e = \langle w, 5, [0, 5]\rangle$ and the weight function adds -1 when both players move without a recharge action, +2 when one player does a recharge action and +4 when both do. The reported times include formula verification and winning strategy finding and are based on 30 repetitions executed on an Intel Core i7-8750H @ 2.2 GHz laptop with 8 GB RAM running Linux kernel version 5.0.0-15-generic.

Table 1 shows that adding an energy condition can decrease significantly performance. Also we can see that without energy, the time difference between one player and two players is significant but rather small (0.06 vs 0.21). On the other hand, with energy, adding one player to the system greatly increases execution time (0.05 vs 145). Both phenomena are no surprise, because introducing the energy agent in the modified system $\mathcal{I}'$ makes the combinatorics much worse.

Discussion. Verification tools for MAS mostly deal with qualitative objectives [2, 7, 10, 11]. We made a step towards automatic verification of quantitative objectives in MAS. We demonstrated that some types of quantitative constraints can be expressed as LTL formulas, and thus existing tools such as MCMAS can be used to perform verification.

References

1. Alur, R., Henzinger, T.A., Kupferman, O.: Alternating-time temporal logic. J. ACM **49**(5), 672–713 (2002). https://doi.org/10.1145/585265.585270
2. Čermák, P., Lomuscio, A., Mogavero, F., Murano, A.: Practical verification of multi-agent systems against SLK specifications. Inf. Comput. **261**, 588–614 (2018)
3. Chatterjee, K., Doyen, L.: Energy parity games. In: ICALP. pp. 599–610. Springer (2010)
4. Chatterjee, K., Henzinger, T.A., Piterman, N.: Strategy Logic. Inf. Comput. **208**(6), 677–693 (2010). https://doi.org/10.1016/j.ic.2009.07.004, http://dx.doi.org/10.1016/j.ic.2009.07.004
5. Della Monica, D., Murano, A.: Parity-energy ATL for qualitative and quantitative reasoning in MAS. In: AAMAS. pp. 1441–1449. International Foundation for Autonomous Agents and Multiagent Systems (2018)
6. Ferber, J., Weiss, G.: Multi-agent systems: an introduction to distributed artificial intelligence, vol. 1. Addison-Wesley Reading (1999)
7. Lomuscio, A., Qu, H., Raimondi, F.: MCMAS: an open-source model checker for the verification of multi-agent systems. STTT **19**(1), 9–30 (2017)
8. Mogavero, F., Murano, A., Perelli, G., Vardi, M.Y.: Reasoning about strategies: On the model-checking problem. ACM Trans. Comput. Log. **15**(4), 34:1–34:47 (2014). https://doi.org/10.1145/2631917
9. Mogavero, F., Murano, A., Perelli, G., Vardi, M.Y.: Reasoning about Strategies: on the Satisfiability Problem. Logical Methods in Computer Science. **13**(1) (2017). https://doi.org/10.23638/LMCS-13(1:9)2017
10. Belardinelli, F., Jamroga, W., Kurpiewski, D., Malvone, V., Murano, A.: Strategy Logic with Simple Goals: Tractable Reasoning about Strategies. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019. Ed. by Sarit Kraus. ijcai.org, pp. 88-94 (2019). https://doi.org/10.24963/ijcai.2019/13
11. Čermák, P., Lomuscio, A., Murano, A.: Verifying and Synthesising Multi-Agent Systems against One-Goal Strategy Logic Specifications. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA. Ed. by Blai Bonet and Sven Koenig. AAAI Press, pp. 2038-2044 (2015). ISBN: 978-1-57735-698-1. http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9959