Regular Model Checking of Epistemic Properties

Daniel STAN
Epita Paris

February 24, 2023

Abstract

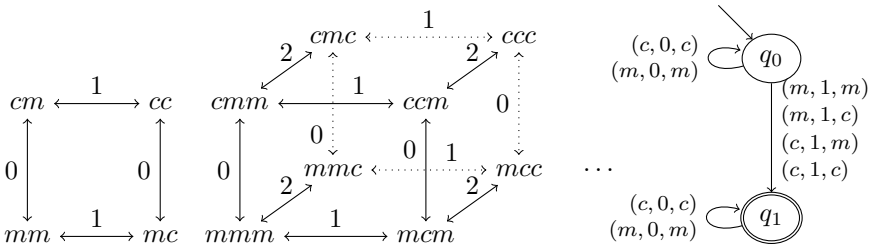
In this talk, we consider systems of multiple agents and reason about their common knowledge of their environment. As the number of agents is a parameter, we model a parameterised family of systems by a finite length-preserving transducer, or for short, a finite deterministic automaton on a product alphabet. In the simple setting of modal and public announcement logics, one can compute the set of states satisfying a given specification as a regular language, hence the regular model checking (RMC) problem is decidable. However, reasoning on a parameterised system might require new constructions, as for example a public announcement operator iterated a finite but arbitrary number of times. In this new setting, we explain how to resort to more advanced RMC techniques and in particular to active learning techniques such as the L* algorithm proposed by Angluin [4].

Joint work with Anthony W. Lin initially published at AAMAS'21 [15].

1 Regular Kripke Structures

Throughout this presentation, we consider as an illustration the muddy children puzzle in knowledge reasoning [10]: Suppose that there are a total of N children, where M of them has a mud on their forehead. Each child can observe whether another child's status, but not himself. Initially, their father declares that there is a muddy child (i.e. with a mud on their forehead). The rest of the protocol goes in rounds: At each round, he asks them whether they know if they are muddy, and they individually answer by yes or no. By unanimously answering no, another fact is commonly deduced by the children: Even the muddy children –who see one less muddy child than the others– cannot conclude on their own state. After a few rounds (more precisely M rounds), all children will discover the so-called common knowledge of which children (including themselves) are muddy and which are not, regardless of the value of the parameters M and N (e.g. see [10]).

Given an observation relation (model), one would like to check that a given sequence of announcements (specification) will eventually lead to the expected common knowledge. Moreover, we assume here that the number of involved



(a) Family of Kripke structures, self loops are omitted.

(b) Its length preserving transducer.

Figure 1: Muddy Children model encoding.

agents is a fixed but unknown parameter and would like to verify the model for all possible values of this parameter. The muddy children puzzle described above is therefore a typical example of a *parameterised verification problem* [6, 3, 11, 12] but with respect to epistemic properties.

We represent common knowledge in a system by a finite Kripke structure, where transitions $\overset{a}{\rightsquigarrow}$ between two states are interpreted as *indistinguishability* by agent a. As an example, Figure 1a depicts a family of Kripke structures for the muddy children puzzle. A state is a word of the set $S = \{m, c\}^*$ and is labelled by a subset of the set of atomic propositions, here $AP = \{m\}$.

In the parameterised setting, we aim at providing a finite representation of this parameterised family. In the spirit of *regular model checking* [1, 2, 7, 8, 17], we resort to a regular encoding of the transitions $\dot{\rightsquigarrow}$, that is to say, length-preserving transducer, or for short, a finite deterministic automaton. Formally, an indistinguishability transition $s_1 \overset{a}{\rightsquigarrow} s_2$ between states $s_1 = x_1 \cdots x_n$ and $s_2 = y_1 \cdots y_n$ by agent $a \in [1, n]$ is encoded by the word $(x_1, 0, y_1) \cdot (x_2, 0, y_2) \cdots (x_a, 1, y_a) \cdots (x_n, 0, y_n)$. The set of all such words in $\Sigma \times \{0, 1\} \times \Sigma$ turns out to be a regular language, which can be recognized by a finite automaton as depicted in Figure 1b.

2 Parameterised Public Announcement Logic

In order to specify epistemic properties, we introduce now a variant of *public announcement logic* (PAL). In order to account for the parameterised number of agents, the logic allows for quantification over agents. A formula in *parameterised public announcement logic* (PPAL) is formally defined by the following grammar:

$$\varphi ::= \top \mid \varphi \wedge \varphi \mid \neg\varphi \mid \exists a : \varphi \mid a = 0 \mid a = b + k \mid p_a \mid [a]\varphi \mid [\varphi!]\,\varphi$$

Where a, b are index/agent variables, $k \in \mathbb{N}$ is any integral constant and $p \in AP$ is any atomic proposition.

These notations and their semantics are directly inspired by [5]: Similarly to the modal operator $\Box$, the formula $[a]\varphi$ asks whether agent *a knows* that a sub-formula φ holds, meaning that no matter what transitions $\overset{a}{\rightsquigarrow}$ labelled by a is taken, the formula φ holds in the target state. The public announcement $[\varphi!]\,\psi$ relates to the broadcast nature (!) of the operator: the model is updated by only keeping states satisfying then φ then evaluating ψ.

As an example the sentence"after this announcement, every child knows their own state", is encoded by:

$$[\exists i : m_i \wedge \forall j, i \neq j \rightarrow \neg m_j!]\,\forall i, [i]m_i \vee [i]\neg m_i$$

An important first result is that *parameterised model checking problem for* PPAL is decidable: if the family $\mathcal{M}$ of Kripke structures is given as a length preserving transducer, then one can compute the semantics of any PPAL formula φ. More precisely, the set of satisfying states is a computable regular set which can be queried for membership.

3 Active Learning Techniques

Finally, we introduce extensions of PPAL through the two following extra operators: (a) Common knowledge by all agents: $[Agt^*]\varphi$; (b) Iterated Public Announcement: $[\varphi_1!]^*\,\varphi_2 \equiv \exists k : \underbrace{[\varphi_1!]\ldots[\varphi_1!]}_{k}\,\varphi_2$.

The latter operator is necessary for expressing termination of the example protocol: The children eventually conclude on their own status, but only after an arbitrary but finite number (M) of public announcements, which cannot be expressed by a fixed PPAL formula.

Although the introduced operators make the model checking problem undecidable, we provide semi-decision procedures thanks to the following observations: (a) Satisfaction of the $[Agt^*]$ operator is essentially similar to safety checking in regular model checking. It can therefore be addressed by active learning techniques [9, 13]: Algorithms as Angluin L* [4] can indeed be instantiated to reconstruct the set of safe states. Termination is ensured if, and only if, the target set is regular, that is to say, when the formula has a regular semantics; (b) Similarly, we provide a semi-decision procedure for computing the semantics of a formula $[\varphi!]^*\,\psi$. The procedure involves the active learning of a so-called *disappearance relation* for the formula ψ.

We conclude by discussing implementation details [14] and possible extensions to dynamic epistemic properties. A worth mentioning extension is the introduction of a more powerful public announcement operator to model simple version of the Dining Cryptographer protocol (joint work with Felix Thoma from TU Kaiserslautern). We refer the readers to [16, 15] for complete references (joint work with Anthony W. Lin).

References

[1] Parosh Aziz Abdulla. Regular model checking. *Int. J. Softw. Tools Technol. Transf.*, 14(2):109–118, 2012.

[2] Parosh Aziz Abdulla, Bengt Jonsson, Marcus Nilsson, and Mayank Saksena. A survey of regular model checking. In Philippa Gardner and Nobuko Yoshida, editors, *CONCUR 2004 - Concurrency Theory, 15th International Conference, London, UK, August 31 - September 3, 2004, Proceedings*, volume 3170 of *Lecture Notes in Computer Science*, pages 35–48. Springer, 2004.

[3] Benjamin Aminof, Aniello Murano, Sasha Rubin, and Florian Zuleger. Automatic verification of multi-agent systems in parameterised grid-environments. In Catholijn M. Jonker, Stacy Marsella, John Thangarajah, and Karl Tuyls, editors, *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, Singapore, May 9-13, 2016*, pages 1190–1199. ACM, 2016.

[4] Dana Angluin. Learning regular sets from queries and counterexamples. *Inf. Comput.*, 75(2):87–106, November 1987.

[5] Alexandru Baltag and Bryan Renne. Dynamic Epistemic Logic. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2016 edition, 2016.

[6] Roderick Bloem, Swen Jacobs, Ayrat Khalimov, Igor Konnov, Sasha Rubin, Helmut Veith, and Josef Widder. *Decidability of Parameterized Verification*. Synthesis Lectures on Distributed Computing Theory. Morgan & Claypool Publishers, 2015.

[7] A. Blumensath. Automatic structures. Master's thesis, RWTH Aachen, 1999.

[8] A. Blumensath and E. Grädel. Finite presentations of infinite structures: Automata and interpretations. *Theory Comput. Syst.*, 37(6):641–674, 2004.

[9] Yu-Fang Chen, Chih-Duo Hong, Anthony W. Lin, and Philipp Rümmer. Learning to prove safety over parameterised concurrent systems. In *2017 Formal Methods in Computer Aided Design, FMCAD 2017, Vienna, Austria, October 2-6, 2017*, pages 76–83, 2017.

[10] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Vardi. *Reasoning about Knowledge*. MIT Press, 1995.

[11] Panagiotis Kouvaros and Alessio Lomuscio. Automatic verification of parameterised multi-agent systems. In Maria L. Gini, Onn Shehory, Takayuki Ito, and Catholijn M. Jonker, editors, *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '13, Saint Paul, MN, USA, May 6-10, 2013*, pages 861–868. IFAAMAS, 2013.

[12] Panagiotis Kouvaros and Alessio Lomuscio. Parameterised verification for multi-agent systems. *Artificial Intelligence*, 234:152 – 189, 2016.

[13] Daniel Neider and Nils Jansen. Regular model checking using solver technologies and automata learning. Lecture Notes in Computer Science, 2013.

[14] Daniel Stan and Anthony W. Lin. MCPPAL: Regular Model Checking for Parametric Public Announcement Logic (Artifact). https://zenodo.org/record/4507467, 2021. Source `https://arg-git.informatik.uni-kl.de/ds/mcppal`.

[15] Daniel Stan and Anthony W. Lin. Regular model checking approach to knowledge reasoning over parameterized systems. In Frank Dignum, Alessio Lomuscio, Ulle Endriss, and Ann Nowé, editors, *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, pages 1254–1262. ACM, 2021.

[16] Daniel Stan and Anthony Widjaja Lin. Regular model checking approach to knowledge reasoning over parameterized systems, 2021.

[17] Anthony Widjaja To and Leonid Libkin. Algorithmic metatheorems for decidable LTL model checking over infinite systems. In *FoSSaCS*, pages 221–236, 2010.