Compositional model checking of monadic least fixed point logic

Matko Botinčan

Department of Mathematics, University of Zagreb
Bijenička cesta 30, 10000 Zagreb, Croatia
mabotinc@math.hr

Supervisor: Dean Rosenzweig

Keywords: model checking, monadic least fixed point logic, compositional verification

Abstract. Compositional model checking refers to reducing the problem of model checking some property of the whole system to the problem of model checking its components. In this paper, we investigate compositional approach for model checking monadic least fixed point logic on transition systems assembled from the components by using the H-sum and the fusion operator.

1 Research Area – Main Themes

The main concern of my recently started PhD research is to investigate possible applications of suitable algorithmic methods of mathematical logic and (finite) model theory to the model checking technique [3] with a hope to obtain yet another tool to increase its feasibility. One particular theme I am currently working on is development of algorithms for solving the model checking problem for certain variants of fixed point logic by means of solving the corresponding model checking games, i.e. computing the participating players' winning strategies [2]. Another theme of my research is use of compositional approach for solving the model checking problem, i.e. reducing the problem of model checking the whole system to the problem of model checking its constituents. This paper presents some of the obtained results related to this second theme.

2 Related Work

The main idea of the compositional approach is to infer the truth value of some property of a composed structure from informations about truth values of certain properties of its components. To state this a bit more precisely, suppose that a structure $\mathfrak{A}$ is composed from the components $\mathfrak{A}_1, \ldots, \mathfrak{A}_n$ and that a property to be checked is φ. The goal is to find an algorithm which for the given φ computes formulae $\varphi_1, \ldots, \varphi_n$ such that $\mathfrak{A} \models \varphi$ iff $\mathfrak{A}_1 \models \varphi_1, \ldots, \mathfrak{A}_n \models \varphi_n$. The main difficulty in finding such an algorithm (or proving that, at least in this form, it does not exist) lies in the choice of (1.) the logical formalisms for expressing properties of the structures and (2.) the set of operators by which a composed structure can be assembled from the components (and, in some cases one could also argue for the choice of (3.) the class of structures).

The compositional approach has first been introduced in the seminal paper of Feferman and Vaught [6] where it was investigated how to determine the truth value of a sentence of the first order logic on the generalized product of Tarski's (first order) structures by looking at its components and the index structure. Many interesting theoretical results were obtained by extending this approach, the most influential probably being the landmark paper of Shelah [12]. However, only until recently, this method lived in a purely theoretical world. Development of the model checking technique and increased interest in its algorithmic aspects brought relevant ideas of the compositional approach to a more practical side [9, 10, 13], or even (for some fragments of CTL and CTL*) rediscovered them [7, 11].

Therefore, the use of the compositional approach in the context of model checking is certainly not new. Nevertheless, there are many questions waiting to be answered, especially those related to the particular choices of (1.) and (2.). In this paper, we present two rather simple results for the monadic least fixed point logic as the choice of (1.) and H-sum and the fusion operation for the choice of (2.) (for brevity, transition systems are taken as the choice of (3.)). The obtained results have very technical and tedious proofs, thus, we only provide mere hints for them.

3 Preliminaries

A transition system is a tuple $\mathcal{A} = (S, (E_\alpha)_\alpha, (P_\iota)_\iota)$, where S is a set of states, $E_\alpha \subseteq S \times S$, for each transition label α, and $P_\iota \subseteq S$, for each atomic proposition ι. In this paper, we regard each transition system $\mathcal{A}$ as a Tarski's (first order) relational structure with the universe S, binary relations $(E_\alpha)_\alpha$ and unary relations $(P_\iota)_\iota$. We associate to every transition system a signature τ consisting of its binary and unary relational symbols. Therefore, for a fixed τ, all τ-transition systems have the same common set of transitions and atomic propositions.

There exist many logical formalisms for expressing properties of transition systems. Some of the most important are those equipped with least (and greatest) fixed points operators (such as, for instance, the modal μ-calculus), giving a good balance between high expressive power and well-behaved algorithmic complexity. The modal μ-calculus is also important for a number of other reasons, in particular, it subsumes a variety of modal and temporal logics used in verification (i.e. LTL, CTL, CTL*, PDL). However, in this paper we shall be mainly concerned with a somewhat more expressive monadic least fixed point logic (MLFP).

MLFP is the logic obtained from the first order logic by adding operators for forming monadic least and greatest fixed points. Namely, if $\psi(R, x)$ is a MLFP-formula and R is a unary relational variable occurring only positively in ψ, then $[\mathbf{lfp}\, Rx.\psi](t)$ and $[\mathbf{gfp}\, Rx.\psi](t)$ (where t is an arbitrary term) are also

MLFP-formulae. Semantics of fixed point MLFP-formulae is defined as follows. Let $\mathcal{A} = (S, (E_\alpha)_\alpha, (P_\iota)_\iota)$ be a τ-transition system and ψ a MLFP$[\tau \cup \{R\}]$-formula (by MLFP$[\tau \cup \{R\}]$ we denote the set of all MLFP-formulae using binary and unary relational symbols from $\tau \cup \{R\}$). Formula ψ gives rise to an operator $\psi^{\mathcal{A}} \colon \mathcal{P}(S) \to \mathcal{P}(S)$ assigning to every set $R \subseteq S$ the set $\psi^{\mathcal{A}}(R) = \{s \in S \mid (\mathcal{A}, R) \models \psi(R, s)\}$. As R occurs in ψ only positively, the operator $\psi^{\mathcal{A}}$ is monotone in R, and thus has a least fixed point $\mathbf{lfp}(\psi^{\mathcal{A}})$ and a greatest fixed point $\mathbf{gfp}(\psi^{\mathcal{A}})$. Now we define that $\mathcal{A} \models [\mathbf{lfp}\, Rx.\psi](t)$ iff the interpretation of t is contained in $\mathbf{lfp}(\psi^{\mathcal{A}})$, and analogously for greatest fixed points.

Our choice for considering the MLFP logic is twofold. Firstly, it contains the modal μ-calculus, while its model checking problem still being of polynomial time complexity in the size of a transition system, and thus represents a highly expressive, but algorithmically manageable logic. Secondly, our main task is to explore practically applicable techniques from (finite) model theory, where a convenient game-theoretical characterization of MLFP in terms of a variant of Ehrenfeucht-Fraïssé games with pebbles [1, 5] happens to be of crucial importance.

Let us start exposition of the compositional approach by first considering the case (already analyzed in the literature) in which the operation for composing structures is chosen to be a disjoint union (by $\mathcal{A} \sqcup \mathcal{B}$ we denote the disjoint union of transition systems $\mathcal{A}$ and $\mathcal{B}$). The first step towards a composition theorem for this operation can be stated as follows.

Theorem 1 ([1]). *Let $\mathcal{A}$, $\mathcal{B}$, $\mathcal{A}'$, $\mathcal{B}'$ be τ-transition systems. If $\mathcal{A}$ and $\mathcal{B}$ satisfy the same* MLFP$[\tau]$*-sentences as $\mathcal{A}'$ and $\mathcal{B}'$, respectively, then $\mathcal{A} \sqcup \mathcal{B}$ satisfies the same* MLFP$[\tau]$*-sentences as $\mathcal{A}' \sqcup \mathcal{B}'$.*

Proof. By combining the winning strategies for the two pairs of transition systems in the Ehrenfeucht-Fraïssé game for MLFP.

Let us denote with $\mathbf{T}(\mathcal{A})$ the set of all MLFP$[\tau]$-sentences satisfied in $\mathcal{A}$. The previous theorem rephrased states that $\mathbf{T}(\mathcal{A} \sqcup \mathcal{B})$ is uniquely determined by $\mathbf{T}(\mathcal{A})$ and $\mathbf{T}(\mathcal{B})$. However, the compositional approach requires more, namely, that it is possible to effectively compute $\mathbf{T}(\mathcal{A} \sqcup \mathcal{B})$ from $\mathbf{T}(\mathcal{A})$ and $\mathbf{T}(\mathcal{B})$. For the case of the disjoint union it is indeed possible to obtain such a stronger statement.

Theorem 2. *For every* MLFP$[\tau]$*-formula φ one can effectively compute a reduction sequence, i.e. a sequence of* MLFP$[\tau]$*-formulae $\langle \vartheta_1, \dots, \vartheta_k, \eta_1, \dots, \eta_l \rangle$ and a boolean function $F_\varphi \colon \{0,1\}^{k+l} \to \{0,1\}$ such that for every $\mathcal{A}$ and $\mathcal{B}$ we have $\mathcal{A} \sqcup \mathcal{B} \models \varphi$ iff $F_\varphi(a_1, \dots, a_k, b_1, \dots, b_l) = 1$, where $a_j = 1$ iff $\mathcal{A} \models \vartheta_j$ and $b_j = 1$ iff $\mathcal{B} \models \eta_j$.*
The time complexity of the computation of the reduction sequence is exponential in the quantifier depth of φ.

Proof. By analyzing the proof of Theorem 1 and tedious book keeping, similarly as done in [8].

A theorem like this also holds for more general sum-like transition systems, obtained as images under quantifier free MSO transductions [8] (the corresponding operation is then called sum-like operation). An example of its application (for the case of first order formulae) is given in [9], where it is shown that transition systems obtained by uniform graph substitution (pipelines) can be model checked by model checking their constituents.

4 Results

Our interest is to explore whether previous results can be obtained for some other (more or less practically motivated) classes of operations on transition systems, not known to be sum-like. Here we first consider the so called H-sum (also known as union with indentification by names) which have been evidenced to be of particular relevance in building large computer chips from small components [8].

Let $\mathcal{A} = (S^{\mathcal{A}}, (E_\alpha^{\mathcal{A}})_\alpha, (P_\iota^{\mathcal{A}})_\iota)$ and $\mathcal{B} = (S^{\mathcal{B}}, (E_\alpha^{\mathcal{B}})_\alpha, (P_\iota^{\mathcal{B}})_\iota)$ be two τ-transition systems such that for $H := S^{\mathcal{A}} \cap S^{\mathcal{B}}$, it holds $E_\alpha^{\mathcal{A}} \cap (H \times H) = E_\alpha^{\mathcal{B}} \cap (H \times H)$ and $P_\iota^{\mathcal{A}} \cap H = P_\iota^{\mathcal{B}} \cap H$. The H-sum of $\mathcal{A}$ and $\mathcal{B}$ is given by the transition system $\mathcal{A} \oplus_H \mathcal{B} = (S^{\mathcal{A}} \cup S^{\mathcal{B}}, (E_\alpha^{\mathcal{A}} \cup E_\alpha^{\mathcal{B}})_\alpha, (P_\iota^{\mathcal{A}} \cup P_\iota^{\mathcal{B}})_\iota)$. We have obtained that analogues of Theorems 1 and 2 also hold for H-sums of transition systems.

Theorem 3. $\mathbf{T}(\mathcal{A} \oplus_H \mathcal{B})$ *is uniquely determined by* $\mathbf{T}(\mathcal{A})$ *and* $\mathbf{T}(\mathcal{B})$. *Moreover, there exists an algorithm which for every* MLFP$[\tau]$*-sentence computes a reduction sequence, i.e. a sequence of formulae as described in Theorem 2.*

Proof. By using Ehrenfeucht-Fraïssé games for MLFP and computing the reduction sequence similarly as in the proof of Theorem 2.

Another operation on transition systems that we consider is a fusion operation which fuses all elements of the set of states satisfying some unary predicate [4]. This operation has shown to be useful in representing relational structures by (short) algebraic expressions and evaluating MSO-formulae on them. Here we are interested in it as a possible candidate for propagation of MLFP model checking results between a transition system and its subsystems.

Let $\mathcal{A} = (S, (E_\alpha)_\alpha, (P_\iota)_\iota)$ be a τ-transition system and assume that $P_\theta \neq \emptyset$. The τ-transition system $\mathcal{A}' := \mathrm{Fuse}_{P_\theta}(\mathcal{A})$ is defined as follows. The set of states of $\mathcal{A}'$ is $S^{\mathcal{A}'} = (S \setminus P_\theta) \cup \{s_\theta\}$, where s_θ is a new element not in S. For unary relations, we have $P_\theta^{\mathcal{A}'} = \{s_\theta\}$, and for $\iota \neq \theta$, $P_\iota^{\mathcal{A}'} = (P_\iota \cap S^{\mathcal{A}'}) \cup \{s_\theta\}$, if $P_\iota \cap P_\theta \neq \emptyset$, and $P_\iota^{\mathcal{A}'} = P_\iota \cap S^{\mathcal{A}'} = P_\iota$, otherwise. For binary relations, we have $E_\alpha^{\mathcal{A}'} = (E_\alpha \cap (S^{\mathcal{A}'} \times S^{\mathcal{A}'})) \cup \{(s, s_\theta) \mid s \in S^{\mathcal{A}'} \wedge \exists s' \in P_\theta \text{ such that } (s, s') \in E_\alpha\} \cup \{(s_\theta, s) \mid s \in S^{\mathcal{A}'} \wedge \exists s' \in P_\theta \text{ such that } (s', s) \in E_\alpha\}$. In the case when $P_\theta = \emptyset$, $\mathrm{Fuse}_{P_\theta}(\mathcal{A})$ is defined to be equal to $\mathcal{A}$.

Although the fuse operation can be defined by using MSO-transduction [4], the corresponding defining formula is not quantifier free, and therefore, an analogue of Theorem 2 cannot be directly obtained. However, we have established that an analogue of Theorem 1 does hold.

Theorem 4. $\mathbf{T}(\mathrm{Fuse}_{P_\theta}(\mathcal{A}))$ *is uniquely determined by* $\mathbf{T}(\mathcal{A})$.

Proof. Again by using Ehrenfeucht-Fraïssé games for MLFP.

Nevertheless, it is still not clear how to effectively compute (e.g. by some kind of reduction sequence) $\mathbf{T}(\mathrm{Fuse}_{P_\theta}(\mathcal{A}))$ from $\mathbf{T}(\mathcal{A})$. We strongly believe that this computation is possible, i.e. that an analogue of Theorem 2 should hold, too.

5 Conclusions and Future Work

In this paper, we have considered the compositional approach for solving the model checking problem for monadic least fixed point logic. We have shown that a composition theorem holds for model checking MLFP properties on transition systems assembled with a use of the H-sum operation. However, for transition systems obtained with the fusion operation, we were only able to prove a partial result towards compositionality.

It is hard to expect that the compositional approach would replace any of the "traditional" methods for model checking due to, in general, high complexity of computing reduction sequences. However, it could help to make the process of incremental model checking more efficient [9, 10]. Namely, in realistic situations, the system under verification is developed in a stepwise manner, i.e. the system at a certain step of development differs from the one at the next step only in a small (number of) components. If the compositional approach would be exploited in such cases, model checking the whole system could be reduced to (re)computation of the truth values of only a small number of formulae from the reduction sequence. Our preliminary analysis of the computational complexity of the whole procedure together with those from [9] and [10] show that for large systems considerable gains in efficiency could be (at least theoretically) possible. We hope to obtain more insight into this aspect after making a prototype implementation of the whole procedure.

A careful reader has probably observed that we have not mentioned the synchronized product as a possible choice for an operator for composing transition systems, although it is one of the most crucial operations in modeling concurrent and reactive systems. Unfortunately, it is not difficult to see that Theorems like 1 and 2 do not hold for direct products and MLFP logic (theorems do not hold even for strictly weaker logics such as e.g. transitive closure logic). However, for the case of first order logic they do hold (in fact, that was the main result of [6]). A similar result has been recently obtained for finitely synchronized products of (infinite) transition systems and first order logic extended by reachability predicates [13]. It would be interesting to investigate this border more precisely, and to analyze whether such result could be obtained for some stronger logical formalisms and weaker product-like operations.

References

1. U. Bosse. An Ehrenfeucht-Fraïssé game for fixed point logic and stratified fixed point logic. In Proceedings of CSL '92, Lecture Notes in Computer Science vol. 702, 100–114, Springer, 1993.
2. M. Botinčan. Solving a subclass of backtracking games. In preparation.
3. E. M. Clarke, O. Grumberg and D. A. Peled. Model Checking. MIT Press, 1999.
4. B. Courcelle and J. A. Makowsky. Fusion in Relational Structures and the Verification of Monadic Second-Order Properties. Mathematical Structures in Computer Science 12: 203–235, 2002.
5. H.-D. Ebbinghaus and J. Flum. Finite Model Theory, 2nd Edition. Springer, 1999.
6. S. Feferman and R. Vaught. The first-order properties of products of algebraic systems. Fundamenta Mathematicae, 47: 57-103, 1959.
7. O. Grumberg and D. E. Long. Model checking and modular verification. ACM Transactions on Programming Languages and Systems 16 (3): 843–871, 1994.
8. J. A. Makowsky. Algorithmic aspects of the Feferman-Vaught Theorem. Annals of Pure and Applied Logic, 126 (1-3): 159–213, 2004.
9. J. A. Makowsky and E. Ravve. Incremental Model Checking for Decomposable Structures. In Proceedings of MFCS '95, Lecture Notes in Computer Science vol. 969, 540–551, Springer, 1995.
10. J. A. Makowsky and E. Ravve. Incremental Model Checking for Fixed Point Properties on Decomposable Structures. Technical Report TR844, revised version, April 1995, Department of Computer Science, Technion-Israel Institute of Technology, Haifa, Israel, 1995.
11. K. Schneider. Model Checking on Product Structures. In Proceedings of FMCAD '98, Lecture Notes in Computer Science vol. 1522, 483–500, Springer, 1998.
12. S. Shelah. The monadic theory of order. Annals of Mathematics 102: 379-419, 1975.
13. S. Wöhrle and W. Thomas. Model Checking Synchronized Products of Infinite Transition Systems. In Proceedings of LICS '04, 2–11, IEEE Computer Society, 2004.