A Theoretical Limit for Safety Verification Techniques with Regular Fix-point Computations

Y. Boichut

INRIA-PAREO Team
Laboratoire Lorrain de Recherche en Informatique et ses Applications
Campus Scientifique - BP 239 - 54506 Vandoeuvre-lès-Nancy Cedex. FRANCE

P.-C. Héam

INRIA-CASSIS Project
Laboratoire d'Informatique de l'Université de Franche-Comté
16 route de Gray, 25030 Besancon Cedex. FRANCE.

Abstract

In computer aided verification, the reachability problem is particularly relevant for safety analyses. Given a regular tree language L, a term t and a relation R, the reachability problem consits in deciding whether there exist a positive integer n and terms $t_0, t_1, \ldots, t_n$ such that $t_0 \in L$, $t_n = t$ and for every $0 \leq i < n$, $(t_i, t_{i+1}) \in R$. In this case, the term t is said to be reachable, otherwise it is said unreachable. This problem is decidable for particular kinds of relations, but it is known to be undecidable in general, even if L is finite. Several approaches to tackle the unreachability problem are based on the computation of an $\mathcal{R}$-closed regular language containing L. In this paper we show a theoretical limit to this kind of approaches for this problem.

Key words: Reachability problem, regular tree languages, undecidable, theoretical limit.

We assume that the reader is familiar with basic notions and notations on terms and on bottom-up tree automata. For a general reference see [5,1].

Email addresses: boichut@loria.fr (Y. Boichut), heampc@lifc.univ-fcomte.fr (P.-C. Héam).

1 Introduction

In this paper we show a theoretical limit of regular fix-point techniques used for reachability analyses.

Automatic verification of software systems is one of the most challenging research problems in computer aided verification. In this context, regular model-checking has been proposed as a general framework for analysing and verifying infinite state systems. Thus, systems are modelled using regular representations: configurations of the systems are modelled by finite words or trees (of unbounded size) and the dynamic of the systems is modelled by a relation $\mathcal{R}$ (in practice a transducer or a (term) rewriting system). Then, safety analysis of the system is reduced to the computation of regular languages closed under a relation $\mathcal{R}$: given a regular language L, a relation $\mathcal{R}$ and a regular set L_P of *bad configurations*, the question is to decide whether $R^\star(L) \cap L_p = \emptyset$ where $\mathcal{R}^\star$ is the reflexive transitive closure of $\mathcal{R}$. Since $\mathcal{R}^\star(L)$ is in general neither regular nor computable, several approaches handle restricted cases for this problem [7,6,11,15].

However, modelling real systems leads in general out of decidable cases. In this context, several regular fix-point automatic [4] or human guided techniques [12,9,8] were developed in order to prove safety properties. The goal of these techniques is to compute a regular language K_{over} containing L and which is $\mathcal{R}$-closed. The language K_{over} is an over approximation of $\mathcal{R}^\star(L)$ (for language inclusion) and if $K_{\text{over}} \cap L_p = \emptyset$, then $\mathcal{R}^\star(L) \cap L_p = \emptyset$. This approach has been successfully used in order to prove safety of security protocols [10,14,13,3] or recently for static analysis of JAVA programs [2].

In this direction we cannot get away from the question to know whether this kind of fix-point approaches can always be used to prove safety of systems in the following sense: given the model of a system by a regular language L and a relation $\mathcal{R}$, for any language L_p such that $\mathcal{R}^\star(L) \cap L_p = \emptyset$, does there exist an $\mathcal{R}$-closed regular language K_{over} containing L and satisfying $K_{\text{over}} \cap L_p = \emptyset$? This issue can also be formalised as follows: does the following equality hold

$$\mathcal{R}^*(L) = \bigcap_{\mathcal{R}^\star(L) \subseteq K,\ \mathcal{R}(K) \subseteq K} K,$$

where the intersection is restricted to regular languages?

In this paper we give a negative answer to this question.

2 Main result

Proposition 1 *Let* $L = \{f(A, A)\}$, $\mathcal{R} = \{f(x, y) \to f(h(x), h(y)),\ f(h(x), h(y)) \to f(x, y),\ f(h(x), A) \to A, f(A, h(x)) \to A\}$ *where* x *and* y *are variables. One has* $A \notin R^\star(L)$ *but*

$$A \in \bigcap_{L \subseteq K,\ R(K) \subseteq K} K.$$

PROOF. Let $H = \{f(h^k(A), h^k(A)) \mid k \in \mathbb{N}\}$. First we claim that $\mathcal{R}^\star(L) = H$. Starting from $f(A, A)$ and using the rule $f(x, y) \to f(h(x), h(y))$, one has $L \subseteq H \subseteq \mathcal{R}^\star(L)$. Moreover, H is obviously closed by the rule $f(h(x), h(y)) \to f(x, y)$. Therefore, since the two rules $f(h(x), A) \to A$ and $f(A, h(x)) \to A$ cannot be applied to terms in H, it follows that $\mathcal{R}^\star(L) = H$, proving the claim. Furthermore, $A \notin R^*(L)$ Moreover one can easily prove that $\mathcal{R}^\star(L)$ is not regular using classical pumping arguments.

Secondly, let K_{over} be a regular language such that $L \subseteq K_{\text{over}}$ and $R(K_{\text{over}}) \subseteq K_{\text{over}}$. Let also S be the regular language $\{f(h^k(A), h^\ell(A)) \mid k \geq 0, \ell \geq 0\}$. Since $\mathcal{R}^\star(L) \subseteq K_{\text{over}}$, $\mathcal{R}^\star(L) \cap S \subseteq K_{\text{over}} \cap S$. Using the claim, one has $\mathcal{R}^\star(L) \cap S = \mathcal{R}^\star(L)$. Consequently $\mathcal{R}^\star(L) \subseteq K_{\text{over}} \cap S$. Now, it is well known that the intersection of two regular tree languages is regular too. Thus $K_{\text{over}} \cap S$ is regular. However $\mathcal{R}^\star(L)$ is not regular. Consequently, the inclusion $\mathcal{R}^\star(L) \subset K_{\text{over}} \cap S$ is strict. So let t be an element of $K_{\text{over}} \cap S \setminus \mathcal{R}^\star(L)$. The term t is of the form $t = f(h^k(A), h^\ell(A))$ with $k \neq \ell$. Without loss of generality, we may assume that $k > \ell$. Since K_{over} is $\mathcal{R}$-closed and using the rule $f(h(x), h(y)) \to f(x, y)$, the term $f(h^{k-\ell}(A), A)$ is in K_{over}. Now the rule $f(h(x), A) \to A$ can be applied on $f(h^{k-\ell}(A), A)$. Consequently, K_{over} being $\mathcal{R}$-closed, it follows that $A \in K_{over}$, which concludes the proof. □

Thus, we have shown that A will be in every over-approximation computed by a regular fix-point technique. So, we won't be able to show it unreachable.

3 Conclusion

Undoubtedly, regular fix-point techniques mentioned previously have led to great results as seen in introduction. Nevertheless, they are disarmed against the problem illustrated in Proposition 1. This raises several open questions:

- Can we decide whether

$$\mathcal{R}^\star(L) = \bigcap_{L \subseteq K,\ \mathcal{R}(K) \subseteq K, K \text{ regular}} K?$$

- If the answer is no, does there exist decidable conditions on L and $\mathcal{R}$ such that the above equality holds?
- How regular fix-point approaches may be extended in order to handle more cases?

References

[1] F. Baader and T. Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1998.

[2] Y. Boichut, T. Genet, T. Jensen, and L. Le Roux. Rewriting Approximations for Fast Prototyping of Static Analyzers. In *Proc. 18th RTA Conf., Paris (France)*, volume 4533 of *Lecture Notes in Computer Science*, pages 48–62, 2007.

[3] Y. Boichut, P.-C. Héam, and O. Kouchnarenko. Handling algebraic properties in automatic analysis of security protocols. In *Int. Col. on Theorical Aspects of Computing , ICTAC-06*, volume 4281 of *Lecture Notes in Computer Science*, pages 153–167. Springer Berlin/Heidelberg, 2006.

[4] A. Bouajjani, P. Habermehl, A. Rogalewicz, and T. Vojnar. Abstract regular tree model checking. In *Proceedings of 7th International Workshop on Verification of Infinite-State Systems – INFINITY 2005*, number 4 in BRICS Notes Series, pages 15–24, 2005.

[5] H. Comon, M. Dauchet, R. Gilleron, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. Tree automata techniques and applications. `http://www.grappa.univ-lille3.fr/tata/`, 2002.

[6] J.-L. Coquidé, M. Dauchet, R. Gilleron, and V. S. Bottom-up tree pushdown automata and rewrite systems. In R. V. Book, editor, *Rewriting Techniques and Applications, 4th International Conference, RTA-91*, LNCS 488, pages 287–298, Como, Italy, Apr. 10–12, 1991. Springer-Verlag.

[7] Dauchet and Tison. The theory of ground rewrite systems is decidable. In *LICS: IEEE Symposium on Logic in Computer Science*, 1990.

[8] G. Feuillade, T. Genet, and V. Viet Triem Tong. Reachability Analysis over Term Rewriting Systems. *JAR*, 33 (3-4):341–383, 2004.

[9] T. Genet. Decidable approximations of sets of descendants and sets of normal forms. In *Proc. 9th RTA Conf., Tsukuba (Japan)*, volume 1379 of *LNCS*, pages 151–165. Springer-Verlag, 1998.

[10] T. Genet and F. Klay. Rewriting for Cryptographic Protocol Verification. In *In Proc. CADE'2000*, volume 1831 of *LNAI*. Springer-Verlag, 2000.

[11] R. Gilleron and S. Tison. Regular tree languages and rewrite systems. *Fundam. Inform*, 24(1/2):157–174, 1995.

[12] F. Jacquemard. Decidable approximations of term rewriting systems. In H. Ganzinger, editor, *Proc. 7th RTA Conf., New Brunswick (New Jersey, USA)*, pages 362–376. Springer-Verlag, 1996.

[13] M. Nesi and G. Rucci. Formalizing and Analyzing the Needham-Schroeder Symmetric-Key Protocol by Rewriting. In *In Proceedings of the 2nd Workshop on Automated Reasoning for Security Protocol Analysis*, 2005.

[14] H. Ohsaki and T. Takai. ACTAS: A system design for associative and commutative tree automata theory. *Electr. Notes Theor. Comput. Sci*, 124(1):97–111, 2005.

[15] P. Réty and J. Vuotto. Regular sets of descendants by leftmost strategy. *Electr. Notes Theor. Comput. Sci*, 70(6), 2002.