Branching in Well-Structured Transition Systems

Sylvain Schmitz
Université de Paris, CNRS, IRIF, France
IUF, Paris, France
schmitz@irif.fr

Abstract

The framework of well-structured transition systems has been highly successful in providing generic algorithms to show the decidability of verification problems for infinite-state systems. In some of these applications, the executions in the system at hand are actually trees, and need to be "lifted" to executions over sets of configurations in order to fit in the framework. The downside of this approach is that we might lose precision when analysing the computational complexity of the algorithms, compared to reasoning over branching executions.

2012 ACM Subject Classification Theory of computation → Computational complexity and cryptography; Theory of computation → Program reasoning; Theory of computation → Verification by model checking

Keywords and phrases fast-growing complexity, well-structured transition system

Digital Object Identifier 10.4230/LIPIcs.CSL.2021.3

Category Invited Talk

Funding ANR-17-CE40-0028 BraVAS

Acknowledgements The presentation is based on joint work with Ranko Lazić.

1 Outline

In this talk, I intend to present a few ideas developed jointly with Ranko Lazić in [18] and investigate how to adapt the framework of *well-structured transition systems* (WSTS), due chiefly to Abdulla, Čerāns, Jonsson, and Tsay [1] and Finkel and Schnoebelen [10], in order to handle tree computations. The WSTS framework supplies generic algorithms for model-checking infinite-state systems, where the algorithms' termination relies on a *well-quasi-ordering* [16] of the configurations compatible with the transition relation.

Lifting Branching Systems. Well-structured transitions systems have found numerous applications since their inception in the 1990's, and these already encompass applications for infinite-state systems with branching executions rather than linear ones. In relation to logic in computer science, some of my favourite examples include provability in substructural logics like the conjunctive-implicational fragment of relevance logic [20, 25] or propositional linear logic with either contraction or weakening [17], and satisfiability for fragments of XPath over data trees [14, 6, 9].

Indeed, one can *lift* a branching transition relation to reason instead over linear executions over sets of configurations. Depending on the exact setup, the well-quasi-ordering on configurations is similarly lifted using either the *Smyth* quasi-ordering – also known as the *minoring* quasi-ordering – , or the *Hoare* quasi-ordering – also known as the *majoring* quasi-ordering. In the applications to substructural or data logics mentioned above, the configurations are essentially vectors of natural numbers in $\mathbb{N}^d$ for some d (ordered componentwise), and in those cases the two quasi-orderings over sets of configurations are well [13, 19] and compatible with the lifted transition relations, thereby defining a WSTS.

Algorithmic Complexity. While this lifting approach is successful for establishing decidability results, it is less so when trying to prove complexity upper bounds. In most algorithmic uses of well-quasi-orderings, one can rely on generic combinatorial analyses to extract upper bounds [7, 24, 21, 23, etc.]. The obtained bounds are typically non primitive-recursive, and depend primarily on the quasi-ordering. This approach has been applied to several classes of WSTS, and in many cases these gigantic worst-case complexity upper bounds are really a testament to the expressiveness of the corresponding classes of WSTS, as they are matched with tight lower bounds [12, 15, 11, 4, 21, 22, etc.].

In the case of the Smyth and Hoare quasi-orderings over subsets of $\mathbb{N}^d$ however, the complexity bounds on the lifted WSTS typically do not match the lower bounds. In that respect, Balasubramanian [3] recently improved the upper bounds of Abriola, Figueira, and Senno [2] and his hyper-Ackermannian bounds for the Hoare quasi-ordering over finite subsets of $\mathbb{N}^d$ are tight. But those lower bounds might not be realisable through the lifting of a branching transition system, and so far the known complexity lower bounds for all the mentioned applications [25, 5, 8, 17] are Ackermannian or lower.

References

1 Parosh A. Abdulla, Karlis Čerāns, Bengt Jonsson, and Yih-Kuen Tsay. Algorithmic analysis of programs with well quasi-ordered domains. *Information and Computation*, 160(1–2):109–127, 2000. doi:10.1006/inco.1999.2843.

2 Sergio Abriola, Santiago Figueira, and Gabriel Senno. Linearizing well-quasi orders and bounding the length of bad sequences. *Theoretical Computer Science*, 603:3–22, 2015. doi:10.1016/j.tcs.2015.07.012.

3 A. R. Balasubramanian. Complexity of controlled bad sequences over finite sets of $\mathbb{N}^d$. In *Proceedings of LICS 2020*. ACM, 2020. doi:10.1145/3373718.3394753.

4 Normann Decker and Daniel Thoma. On freeze LTL with ordered attributes. In *Proceedings of FoSSaCS 2016*, volume 9634 of *Lecture Notes in Computer Science*, pages 269–284. Springer, 2016. doi:10.1007/978-3-662-49630-5_16.

5 Stéphane Demri and Ranko Lazić. LTL with the freeze quantifier and register automata. *ACM Transactions on Computational Logic*, 10(3):16, 2009. doi:10.1145/1507244.1507246.

6 Diego Figueira. Alternating register automata on finite words and trees. *Logical Methods in Computer Science*, 8(1), 2012. doi:10.2168/LMCS-8(1:22)2012.

7 Diego Figueira, Santiago Figueira, Sylvain Schmitz, and Philippe Schnoebelen. Ackermannian and primitive-recursive bounds with Dickson's Lemma. In *Proceedings of LICS 2011*, pages 269–278. IEEE, 2011. doi:10.1109/LICS.2011.39.

8 Diego Figueira and Luc Segoufin. Future-looking logics on data words and trees. In *Proceedings of MFCS '09*, volume 5734 of *Lecture Notes in Computer Science*, pages 331–343. Springer, 2009. doi:10.1007/978-3-642-03816-7_29.

9 Diego Figueira and Luc Segoufin. Bottom-up automata on data trees and vertical XPath. *Logical Methods in Computer Science*, 13(4), 2017. doi:10.23638/LMCS-13(4:5)2017.

10 Alain Finkel and Philippe Schnoebelen. Well-structured transition systems everywhere! *Theoretical Computer Science*, 256(1–2):63–92, 2001. doi:10.1016/S0304-3975(00)00102-X.

11 Christoph Haase, Sylvain Schmitz, and Philippe Schnoebelen. The power of priority channel systems. *Logical Methods in Computer Science*, 10(4):4:1–4:39, December 2014. doi:10.2168/LMCS-10(4:4)2014.

12 Serge Haddad, Sylvain Schmitz, and Philippe Schnoebelen. The ordinal recursive complexity of timed-arc Petri nets, data nets, and other enriched nets. In *Proceedings of LICS 2012*, pages 355–364. IEEE Press, 2012. doi:10.1109/LICS.2012.46.

13 Petr Jančar. A note on well quasi-orderings for powersets. *Information Processing Letters*, 72(5–6):155–160, 1999. doi:10.1016/S0020-0190(99)00149-0.

14 Marcin Jurdziński and Ranko Lazić. Alternating automata on data trees and XPath satisfiability. *ACM Transactions on Computational Logic*, 12(3), 2011. `doi:10.1145/1929954.1929956`.

15 Prateek Karandikar and Sylvain Schmitz. The parametric ordinal-recursive complexity of Post embedding problems. In *Proceedings of FoSSaCS 2013*, volume 7794 of *Lecture Notes in Computer Science*, pages 273–288. Springer, 2013. `doi:10.1007/978-3-642-37075-5_18`.

16 Joseph B. Kruskal. The theory of well-quasi-ordering: A frequently discovered concept. *Journal of Combinatorial Theory, Series A*, 13(3):297–305, 1972. `doi:10.1016/0097-3165(72)90063-5`.

17 Ranko Lazić and Sylvain Schmitz. Non-elementary complexities for branching VASS, MELL, and extensions. *ACM Transactions on Computational Logic*, 16(3):20:1–20:30, 2015. `doi:10.1145/2733375`.

18 Ranko Lazić and Sylvain Schmitz. The ideal view on Rackoff's coverability technique. *Information and Computation*, 2020. In press. `doi:10.1016/j.ic.2020.104582`.

19 Alberto Marcone. Fine analysis of the quasi-orderings on the power set. *Order*, 18(4):339–347, 2001. `doi:10.1023/A:1013952225669`.

20 Jacques Riche and Robert K. Meyer. Kripke, Belnap, Urquhart and relevant decidability & complexity. In *Proceedings of CSL 1998*, volume 1584 of *Lecture Notes in Computer Science*, pages 224–240. Springer, 1999. `doi:10.1007/10703163_16`.

21 Fernando Rosa-Velardo. Ordinal recursive complexity of unordered data nets. *Information and Computation*, 254(1):41–58, 2017. `doi:10.1016/j.ic.2017.02.002`.

22 Sylvain Schmitz. *Algorithmic Complexity of Well-Quasi-Orders*. Habilitation thesis, École Normale Supérieure Paris-Saclay, 2017. URL: `http://tel.archives-ouvertes.fr/tel-01663266`.

23 Sylvain Schmitz. The parametric complexity of lossy counter machines. In *Proceedings of ICALP 2019*, volume 132 of *Leibniz International Proceedings in Informatics*, pages 129:1–129:15. LZI, 2019. `doi:10.4230/LIPIcs.ICALP.2019.129`.

24 Sylvain Schmitz and Philippe Schnoebelen. Multiply-recursive upper bounds with Higman's Lemma. In *Proceedings of ICALP 2011*, volume 6756 of *Lecture Notes in Computer Science*, pages 441–452. Springer, 2011. `doi:10.1007/978-3-642-22012-8_35`.

25 Alasdair Urquhart. The complexity of decision procedures in relevance logic II. *Journal of Symbolic Logic*, 64(4):1774–1802, 1999. `doi:10.2307/2586811`.