Verification of Generic, Relational Transition Systems

(Discussion/Short Paper)

Diego Calvanese[1,2], Giuseppe De Giacomo[3], Marco Montali[1] and Fabio Patrizi[3]

[1]Free University of Bozen-Bolzano, Italy
[2]Umeå University, Sweden
[3]Sapienza University of Rome, Italy

Abstract

Generic, relational transition systems form an interesting class of infinite-state transition systems that naturally captures the execution semantics of a variety of formalisms expressing processes operating over (relational) data. Examples of such data-aware processes include action theories in the situation calculus in AI and data-centric business processes in BPM. In this extended abstract, we summarize the main body of results produced in a decade-long research program focused on the verification of generic, relational transition systems against properties specified using variants of first-order temporal logics.

Keywords

data-aware processes, first-order temporal logics, verification, action theories, state-boundedness

1. Introduction

Relational transition systems (RTSs) are infinite-state transition systems whose states are labeled by first-order (FO) interpretations [1]. *Generic RTSs* are RTSs with the property that states with isomorphic interpretations induce the same transitions modulo renaming of objects [2, 3]. This implies that, whenever the current state has a successor state where new, (locally) fresh objects are injected, what matters about such fresh objects is only how they relate to each other and to those already present in the current state. Genericity of an RTS is indeed reminiscent of the well-known notion of genericity in first-order logic and relational databases [4].

Generic RTSs naturally capture the execution semantics of a variety of dynamic systems operating over relational data (henceforth called *data-aware dynamic systems*), which have been investigated in AI, BPM, and data management. Example of such systems are: (i) action theories in the Situation Calculus [5, 6, 7], possibly operating over description-logic knowledge bases [8, 9, 10, 11]; (ii) multiagent systems with data-aware, interacting agents [12, 13, 14, 15]; (iii) artifact- and data-centric business processes [16, 17, 18]; (iv) variants of colored Petri nets where tokens carry (tuples of) data that can be compared by (in)equality [19, 20, 21, 22].

PMAI@IJCAI22: International IJCAI Workshop on Process Management in the AI era, July 23, 2022, Vienna, Austria

calvanese@inf.unibz.it (D. Calvanese); degiacomo@diag.uniroma1.it (G. De Giacomo); montali@inf.unibz.it (M. Montali); patrizi@diag.uniroma1.it (F. Patrizi)
http://www.inf.unibz.it/~calvanese/ (D. Calvanese); http://www.diag.uniroma1.it/degiacomo/ (G. De Giacomo); https://www.inf.unibz.it/~montali/ (M. Montali); http://www.diag.uniroma1.it/patrizi/ (F. Patrizi)
0000-0001-5174-9693 (D. Calvanese); 0000-0001-9680-7658 (G. De Giacomo); 0000-0002-8021-3430 (M. Montali); 0000-0002-9116-251X (F. Patrizi)

CEUR Workshop Proceedings (CEUR-WS.org)

A long series of works has investigated how to analyze these systems, focusing in particular on verification [3, 13, 18, 23, 7] and synthesis (typically, planning) [24, 9, 11]. Notably, properties of interest have to combine, in this setting, temporal/dynamic operators with the ability of querying the interpretations contained in the state, in turn making it possible to predicate on the (un)desired evolution of objects and relations as the system unfolds. Natural candidates to express such propreties are thus first-order temporal logics.

In this extended abstract, we summarize the main body of results on verification of first-order temporal logics over generic RTSs that we have obtained in a decade-long research program.

2. Verification of Generic RTSs

Verification of generic RTSs is undecidable even under severe restrictions on the system dynamics and on the signature of the FO interpretations. In fact, it is undecidable to even check reachability of a proposition in data-aware dynamic systems that manipulate two unary relations through simple actions that can *(i)* test whether one of the relations is empty, *(ii)* insert a new object in one of them, *(iii)* remove an object from one of them provided that an object exists [25, 26].

State-boundedness. In this simple systems, undecidability resides in the ability of the system to accumulate unboundedly many data in a single state. To control this ability, the notion of *state-boundedness* has been introduced in [12, 18] and further studied in [27]. In a state-bounded RTS, every state contains a bounded number of objects. Infinitely many distinct objects can be potentially observed along a run, but only boundedly many can be accumulated in each state.

State-boundedness is a semantic property, decidable to check for a given bound and undecidable to check if the bound is not known [8, 7]. In this light, different data-aware dynamic systems that provide compact ways of specifying generic RTSs have been studied, with the aim of identifying classes that guarantee that the corresponding RTS is indeed state-bounded. This has been done considering: (i) sufficient, syntactic conditions over the system specification [8, 27]; (ii) action theories with fading memory [7]; (iii) resource-constrained colored Petri nets [20]; (iv) controlled generation of fresh identifiers and other modeling guidelines [28, 29].

State-boundedness is essential towards singling out decidable classes for verification. Distinct results are obtained for branching- vs linear-time first-order temporal logics.

Branching-time FO temporal logics. In [2, 3], it is shown that verification of the full FO μ-calculus is decidable over state-bounded, generic RTSs. This is proved constructively, showing how to compute a finite-state abstract RTSs that is guaranteed to satisfy all and only the FO μ-calculus properties of the original one. The abstraction is built by considering the input RTS, the bound, and the number of variables contained in the formula of interest. Notably, the abstraction can be computed also without fixing a specific value for the bound [8].

In [8, 2, 3], fragments of the full FO μ-calculus are studied. Of particular interest is the one with *persistent quantification*, where FO quantification tracks over time only the identity of objects that remain active in consecutive states (whereas objects disappearing from a state are not tracked anymore). For this fragment, it is shown that an abstract RTS can be constructed independently from the formula to verify, that is, considering only the input RTS.

Linear-time FO temporal logics. In [3], it is shown that FO LTL behaves radically differently from the FO μ-calculus: verification of FO LTL properties is undecidable over generic, state-bounded RTSs with a bound of 1. This is particularly interesting, as it implies that in the FO setting, the μ-calculus does *not* subsume LTL, differently from the propositional setting. In [26], the reason for undecidability is singled out, thanks to a reduction from LTL with freeze quantifiers [30]: it resides in the ability of the logic to unrestrictedly quantify over objects that may be arbitrarily far away from each other. This, in turn, hints that decidability may hold for FO LTL with persistent quantification. As shown in [26], this is indeed the case, and further decidability results are obtained for the problem of monitoring state-bounded, evolving traces.

Akcnowledgements

This work is partly supported by the ERC Advanced Grant WhiteMech (No. 834228), the EU ICT-48 2020 project TAILOR (No. 952215), the Sapienza Project DRAPE, the Italian Basic Research (PRIN) projects HOPE and PINPOINT, the unibz ID 2020 project SMART-APP, and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

[1] D. Calvanese, G. De Giacomo, M. Montali, Foundations of data-aware process analysis: A database theory perspective, in: Proc. of PODS, ACM, 2013, pp. 1–12.

[2] D. Calvanese, G. De Giacomo, M. Montali, F. Patrizi, On first-order μ-calculus over Situation Calculus action theories, in: Proc. of KR, AAAI Press, 2016, pp. 411–420.

[3] D. Calvanese, G. De Giacomo, M. Montali, F. Patrizi, First-order μ-calculus over generic transition systems and applications to the Situation Calculus, Information and Computation 259 (2018) 328–347.

[4] S. Abiteboul, R. Hull, V. Vianu, Foundations of Databases, Addison Wesley, 1995.

[5] J. McCarthy, P. J. Hayes, Some philosophical problems from the standpoint of artificial intelligence, Machine Intelligence 4 (1969) 463–502.

[6] R. Reiter, Knowledge in Action. Logical Foundations for Specifying and Implementing Dynamical Systems, The MIT Press, 2001.

[7] G. De Giacomo, Y. Lesperance, F. Patrizi, Bounded Situation Calculus action theories, Artificial Intelligence 237 (2016) 172–203.

[8] B. Bagheri Hariri, D. Calvanese, G. De Giacomo, R. De Masellis, P. Felli, M. Montali, Description logic knowledge and action bases, J. of Artificial Intelligence Research 46 (2013) 651–686. doi:10.1613/jair.3826.

[9] D. Calvanese, M. Montali, F. Patrizi, M. Stawowy, Plan synthesis for knowledge and action bases, in: Proc. of IJCAI, AAAI Press, 2016, pp. 1022–1029.

[10] P. A. Abdulla, C. Aiswarya, M. F. Atig, M. Montali, O. Rezine, Recency-bounded verification of dynamic database-driven systems, in: Proc. of PODS, ACM, 2016, pp. 195–210.

[11] S. Borgwardt, J. Hoffmann, A. Kovtunova, M. Krötzsch, B. Nebel, M. Steinmetz, Expressivity of planning with Horn description logic ontologies, in: Proc. of AAAI, 2022.

[12] F. Belardinelli, A. Lomuscio, F. Patrizi, An abstraction technique for the verification of artifact-centric systems, in: Proc. of KR, 2012, pp. 319–328.

[13] F. Belardinelli, A. Lomuscio, F. Patrizi, Verification of agent-based artifact systems, J. of Artificial Intelligence Research 51 (2014) 333–376.

[14] M. Montali, D. Calvanese, G. De Giacomo, Verification of data-aware commitment-based multiagent system, in: Proc. of AAMAS, IFAAMAS/ACM, 2014, pp. 157–164.

[15] D. Calvanese, G. Delzanno, M. Montali, Verification of relational multiagent systems with data types, in: Proc. of AAAI, AAAI Press, 2015, pp. 2031–2037.

[16] K. Bhattacharya, N. S. Caswell, S. Kumaran, A. Nigam, F. Y. Wu, Artifact-centered operational modeling: Lessons from customer engagements, IBM Systems J. 46 (2007) 703–721.

[17] A. Deutsch, R. Hull, F. Patrizi, V. Vianu, Automatic verification of data-centric business processes, in: Proc. of ICDT, 2009, pp. 252–267.

[18] B. Bagheri Hariri, D. Calvanese, G. De Giacomo, A. Deutsch, M. Montali, Verification of relational data-centric dynamic systems with external services, in: Proc. of PODS, 2013, pp. 163–174.

[19] F. Rosa-Velardo, D. de Frutos-Escrig, Decidability and complexity of Petri nets with unordered data, Theoretical Computer Science 412 (2011) 4439–4451.

[20] M. Montali, A. Rivkin, Model checking Petri nets with names using data-centric dynamic systems, Formal Aspects of Computing 28 (2016) 615–641.

[21] A. Polyvyanyy, J. M. E. M. van der Werf, S. Overbeek, R. Brouwers, Information systems modeling: Language, verification, and tool support, in: Proc. of CAiSE, volume 11483 of *LNCS*, Springer, 2019, pp. 194–212.

[22] S. Ghilardi, A. Gianola, M. Montali, A. Rivkin, Petri nets with parameterised data - Modelling and verification, in: Proc. of BPM, volume 12168 of *LNCS*, Springer, 2020, pp. 55–74.

[23] G. De Giacomo, Y. Lesperance, F. Patrizi, Bounded Situation Calculus action theories and decidable verification, in: Proc. of KR, 2012, pp. 467–477.

[24] J. A. Baier, S. A. McIlraith, Planning with first-order temporally extended goals using heuristic search, in: Proc. of AAAI, AAAI Press, 2006, pp. 788–795.

[25] P. A. Abdulla, C. Aiswarya, M. F. Atig, M. Montali, O. Rezine, Complexity of reachability for data-aware dynamic systems, in: Proc. of the 18th Int. Conf. on Application of Concurrency to System Design (ACSD), IEEE Computer Society, 2018, pp. 11–20.

[26] D. Calvanese, G. De Giacomo, M. Montali, F. Patrizi, Verification and monitoring for first-order LTL with persistence-preserving quantification over finite and infinite traces, in: Proc. of IJCAI, AAAI Press, 2022. To appear.

[27] B. Bagheri Hariri, D. Calvanese, A. Deutsch, M. Montali, State-boundedness in data-aware dynamic systems, in: Proc. of KR, AAAI Press, 2014, pp. 458–467.

[28] D. Solomakhin, M. Montali, S. Tessaris, R. De Masellis, Verification of artifact-centric systems: Decidability and modeling issues, in: Proc. of ICSOC, volume 8274 of *LNCS*, Springer, 2013, pp. 252–266.

[29] M. Montali, D. Calvanese, Soundness of data-aware, case-centric processes, Int. J. on Software Tools for Technology Transfer 18 (2016) 535–558.

[30] S. Demri, R. Lazic, LTL with the freeze quantifier and register automata, ACM Trans. on Computational Logic 10 (2009).