Qualitative past Timeline-Based Games

Renato Acampora
University of Udine, Italy

Luca Geatti
University of Udine, Italy

Nicola Gigante
Free University of Bozen-Bolzano, Italy

Angelo Montanari
University of Udine, Italy

Abstract

This extended abstract discusses timeline-based planning, a modeling approach that offers a unique way to model complex systems. Recently, the timeline-based planning framework has been extended to handle general nondeterminism in a game-theoretic setting, resulting in timeline-based games. In this context, the problem of establishing whether a timeline-based game admits a winning strategy and synthesizing such a strategy have been addressed. We propose exploring simpler yet expressive fragments of timeline-based games by leveraging results about the role of past operators in synthesis from temporal logic specifications. The qualitative fragment of timeline-based planning is a good starting point for this exploration. We suggest introducing syntactic restrictions on synchronization rules so that they only constrain the behavior of the system before the current time point, which is expected to lower the complexity of synthesizing timeline-based games to EXPTIME.

2012 ACM Subject Classification Computing methodologies → Planning for deterministic actions

Keywords and phrases Automata, Planning, Temporal Reasoning

Digital Object Identifier 10.4230/LIPIcs.TIME.2023.22

Category Extended Abstract

1 State of the Art

Timeline-based planning, initially proposed for planning and scheduling of space operations [12], offers a unique approach to modelling complex systems. Unlike action-based planning paradigms, such as STRIPS [5], the timeline-based one does not explicitly separate states, actions, and goals; rather, it models the domain as a set of independent yet interacting components, whose behaviour over time is governed by "synchronization rules". A solution plan is a set of timelines describing a behaviour of the system components that satisfies all rules. This approach has been successfully deployed in systems like the Hubble Space Telescope scheduling and control system (HSTS) [11]. The timeline-based planning framework has recently been extended to handle general nondeterminism in a game-theoretic setting (*timeline-based games*) [7]. The resulting formalism allows one to meet time constraints without being affected by the environment's choices, reducing re-planning issues present in systems that only deal with temporal uncertainty [9]. Establishing the existence of a winning strategy for a timeline-based planning game has been proved to be 2EXPTIME-complete.

In [1], Acampora et al. addressed the problem of establishing whether a timeline-based game admits a winning strategy and, if this is the case, synthesizing such a strategy. In particular, they outlined an algorithm, based on a non-trivial construction of a Deterministic Finite Automaton (DFA), that recognizes solution plans for timeline-based planning problems. As it commonly happens, the synthesis of controllers reduces to the "compilation" of the specification into a DFA, which then serves as a game arena. This deterministic model is

30th International Symposium on Temporal Representation and Reasoning (TIME 2023).
Editors: Alexander Artikis, Florian Bruse, and Luke Hunsberger; Article No. 22; pp. 22:1–22:3
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

required for playing games, and it is necessary to meet optimal computational complexity. The winning strategy for one of the players can then be determined through simple reachability games [8]. In [1], a compilation procedure that serves as the core for the DFA's construction is provided, and a controller implementing the specification is built. As a matter of fact, it was the first effective procedure for the synthesis of a controller starting from a timeline-based game specification.

2 Qualitative Past Timeline-Based Games

In view of the high complexity of timeline-based games, we aim at exploring simpler yet expressive fragments. One promising approach is to leverage results about the role of *past operators* in synthesis from temporal logic specification. Recently, it has been shown that writing the specification using past operators can make the synthesis problem exponentially more efficient [2, 6]. Drawing parallels with co-safety properties in Linear Temporal Logic (LTL), where properties express the fact that something good will eventually happen, we plan to apply these findings to timeline-based games. Specifically, we are thinking of introducing suitable syntactic restrictions on synchronization rules so that they only constrain the behavior of the system before the current time point. In such a way, we expect to lower the complexity of synthesizing timeline-based games to EXPTIME, offering an interesting direction for future research.

To this end, a good starting point is the *qualitative* fragment of timeline-based planning [10], a simpler planning formalism that only considers qualitative (ordering) features of timelines. To solve the synthesis problem for such a formalism, one can represent synchronization rules as partial orders and construct an automaton whose states are downward-closed subsets of such partial orders. Each single synchronization rule leads to a deterministic automaton taking care of matching the elements of the partial order when reading the word representing the solution plan. By generating the union of all the automata (one for each rule), we derive an automaton for the whole system. The size of such a deterministic automaton is exponential in the size of the planning problem. Since the automaton can be built on-the-fly and solving the reachability problem requires (nondeterministic) logarithmic space in the size of the automaton, we get that the problem of identifying solution plans for the suggested fragment belongs to PSPACE. Most importantly, the automaton is deterministic and can serve as the game arena for achieving synthesis in exponential time.

3 Temporal Logic Characterization and Symbolic Algorithms

We conclude the abstract with a discussion of two other future directions. In [4], is given the bounded variant of *Timed Propositional Temporal Logic with Past* ($\mathsf{TPTL_b{+}P}$) used to capture timeline-based problems. It seems natural to look for a *cosafety fragment* of $\mathsf{TPTL_b{+}P}$ that captures past timeline-based problems, restricting the syntax of $\mathsf{TPTL_b{+}P}$ to formulas of the form $\mathsf{F}(\alpha)$, where F is the *eventually* operator and α is a pure past $\mathsf{TPTL_b{+}P}$ formula. Modern algorithms for the synthesis of temporal logic specifications deal with a *symbolic* representation of automata, representing the DFA equivalent to the initial formula by means of Boolean formulas only, in contrast to the explicit-state representation where states and transitions of the automaton are represented as memory locations and pointers. The benefits of using a symbolic representation are known from almost 30 years [3]. Producing a symbolic DFA starting from a past temporal logic specification has proven to be very effective [2, 6]. We expect to lift this result to the case of past timeline-based games.

References

1 Renato Acampora, Luca Geatti, Nicola Gigante, Angelo Montanari, and Valentino Picotti. Controller synthesis for timeline-based games. In Pierre Ganty and Dario Della Monica, editors, *Proceedings of the 13th International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2022, Madrid, Spain, September 21-23, 2022*, volume 370 of *EPTCS*, pages 131–146, 2022. doi:10.4204/EPTCS.370.9.

2 Alessandro Artale, Luca Geatti, Nicola Gigante, Andrea Mazzullo, and Angelo Montanari. Complexity of safety and cosafety fragments of linear temporal logic. In *Proceedings of the AAAI Conference on Artificial Intelligence 37(5)*, pages 6236–6244, 2023. doi:10.1609/aaai.v37i5.25768.

3 Jerry R Burch, Edmund M Clarke, Kenneth L McMillan, David L Dill, and Lain-Jinn Hwang. Symbolic model checking: 1020 states and beyond. *Information and computation*, 98(2):142–170, 1992.

4 Dario Della Monica, Nicola Gigante, Angelo Montanari, Pietro Sala, and Guido Sciavicco. Bounded timed propositional temporal logic with past captures timeline-based planning with bounded constraints. In Carles Sierra, editor, *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 1008–1014, 2017. doi:10.24963/ijcai.2017/140.

5 Richard E. Fikes and Nils J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3):189–208, 1971. doi:10.1016/0004-3702(71)90010-5.

6 Giuseppe De Giacomo, Antonio Di Stasio, Francesco Fuggitti, and Sasha Rubin. Pure-past linear temporal and dynamic logic on finite traces. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 4959–4965. ijcai.org, 2020. doi:10.24963/ijcai.2020/690.

7 Nicola Gigante, Angelo Montanari, Andrea Orlandini, Marta Cialdea Mayer, and Mark Reynolds. On timeline-based games and their complexity. *Theoretical Computer Science*, 815:247–269, May 2020. doi:10.1016/j.tcs.2020.02.011.

8 Swen Jacobs, Roderick Bloem, Romain Brenguier, Rüdiger Ehlers, Timotheus Hell, Robert Könighofer, Guillermo A Pérez, Jean-François Raskin, Leonid Ryzhyk, Ocan Sankur, et al. The first reactive synthesis competition (syntcomp 2014). *International journal on software tools for technology transfer*, 19:367–390, 2017.

9 Marta Cialdea Mayer, Andrea Orlandini, and Alessandro Umbrico. Planning and execution with flexible timelines: a formal account. *Acta Informatica*, 53(6-8):649–680, 2016. doi:10.1007/s00236-015-0252-z.

10 Dario Della Monica, Nicola Gigante, Salvatore La Torre, and Angelo Montanari. Complexity of qualitative timeline-based planning. In *27th International Symposium on Temporal Representation and Reasoning, TIME 2020, September 23-25, 2020, Bozen-Bolzano, Italy*, volume 178 of *LIPIcs*, pages 16:1–16:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.TIME.2020.16.

11 Nicola Muscettola. HSTS: Integrating Planning and Scheduling. In Monte Zweben and Mark S. Fox, editors, *Intelligent Scheduling*, chapter 6, pages 169–212. Morgan Kaufmann, 1994. URL: `https://apps.dtic.mil/sti/pdfs/ADA266991.pdf`.

12 Nicola Muscettola, Stephen F. Smith, Amedeo Cesta, and Daniela D'Aloisi. Coordinating space telescope operations in an integrated planning and scheduling architecture. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation, Sacramento, CA, USA, 9-11 April 1991*, pages 1369–1376. IEEE Computer Society, 1991. doi:10.1109/ROBOT.1991.131803.