Decidability of TeamLTL model-checking

Context

TeamLTL is a temporal hyperlogic designed to express temporal properties of sets of traces (while logics like LTL only speak about a single trace). The most popular logic of that kind is HyperLTL, which offers a high expressivity but for which basic problems like satisfiability or model-checking are untractable or even undecidable.

The syntax is the one of LTL without negation.

$$\varphi ::= p \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid X\varphi \mid \varphi\, U \varphi \mid \varphi\ R\ \varphi$$

The (synchronous) semantics is defined as follows (asynchronous semantics also exist, but will not be considered here). Most operators are naturally extended to sets of traces:

$$
\begin{array}{ll}
T \vDash p & \text{if } \forall t \in T, p \in t[0] \\
T \vDash \varphi \wedge \psi & \text{if } T \vDash \varphi \text{ and } T \vDash \psi \\
T \vDash X\varphi & \text{if } T[1 \cdots] \vDash \varphi \\
T \vDash \varphi\ U\psi & \text{if } \exists k, T[k \cdots] \vDash \psi \text{ and } \forall j < k, T[j \cdots] \vDash \varphi \\
T \vDash \varphi\ R\ \psi & \text{if } \forall k, T[k \cdots] \vDash \psi \text{ or } \exists j < k, T[j \cdots] \vDash \varphi
\end{array}
$$

The twist is in the $\vee$: it is not defined as a boolean disjunction, but with *splits*:

$$T \vDash \varphi \vee \psi \text{ if } \exists T_1, T_2 \text{ such that } T_1 \cup T_2 = T \text{ and } T_1 \vDash \varphi \text{ and } T_2 \vDash \psi$$

We define $G\varphi$ and $F\varphi$ as usual, as $\bot R\varphi$ and $\top U\varphi$. The difficulty arises with formulas like $G(F\ p \vee F\ q)$, which expresses that "at all times, T can be split in two teams, with all traces of the first one synchronously satisfying p at some point and all the other ones synchronously satisfying q at some other point."

While satisfiability comes down to LTL satisfiability (if a TeamLTL formula is satisfiable then it is satisfiable with a single trace), the decidability of model-checking (given a finite Kripke structure and a TeamLTL formula, does the set of traces of the former satisfy the latter?) is still an open problem.

Credits and related work

This problem was presented to me by Martin Zimmermann. TeamLTL was introduced in this paper [1], with (amongst other results) a PSPACE-completeness proof for model-checking when the Kripke structure is a union of "lassos".

It was later shown that model-checking and satisfiability were undecidable for TeamLTL with the boolean negation $\sim$, as well as a "flattening" negation

$\neg\varphi$, which says that φ should not be satisfied by any trace (seen as a singleton) in the set [2]. To the best of my knowledge, decidability of model-checking when only one of those negations is added is open.

In [3] decidability of model-checking is established for two fragments of TeamLTL (called “left-flat” and “k-coherent”).

References

[1] Andreas Krebs et al. “Team Semantics for the Specification and Verification of Hyperproperties”. In: *43rd International Symposium on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK*. Ed. by Igor Potapov, Paul G. Spirakis, and James Worrell. Vol. 117. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, 10:1–10:16. DOI: 10.4230/LIPIcs.MFCS.2018.10. URL: https://doi.org/10.4230/LIPIcs.MFCS.2018.10.

[2] Martin Lück. “On the complexity of linear temporal logic with team semantics”. In: *Theor. Comput. Sci.* 837 (2020), pp. 1–25. DOI: 10.1016/j.tcs.2020.04.019. URL: https://doi.org/10.1016/j.tcs.2020.04.019.

[3] Jonni Virtema et al. “Linear-Time Temporal Logic with Team Semantics: Expressivity and Complexity”. In: *41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2021, December 15-17, 2021, Virtual Conference*. Ed. by Mikolaj Bojanczyk and Chandra Chekuri. Vol. 213. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 52:1–52:17. DOI: 10.4230/LIPIcs.FSTTCS.2021.52. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2021.52.