Article:

Kontinen, J., Sandström, M. orcid.org/0000-0002-6365-2562 and Virtema, J. orcid.org/0000-0002-1582-3718 (2025) Set semantics for asynchronous TeamLTL: expressivity and complexity. Information and Computation, 304. 105299. ISSN 0890-5401

https://doi.org/10.1016/j.ic.2025.105299

Information and Computation 304 (2025) 105299

Contents lists available at ScienceDirect

Information and Computation

journal homepage: www.elsevier.com/locate/yinco

Set semantics for asynchronous TeamLTL: Expressivity and complexity

Juha Kontinen [a,1], Max Sandström [a,b,*,2], Jonni Virtema [a,b,3]

[a] University of Helsinki, Department of Mathematics and Statistics, Pietari Kalmin katu 5, 00014, Helsinki, Finland
[b] University of Sheffield, School of Computer Science, Western Bank, S10 2TN, Sheffield, UK

ARTICLE INFO

Article history:
Received 15 December 2023
Received in revised form 24 March 2025
Accepted 26 March 2025
Available online 27 March 2025

Keywords:
Hyperproperties
Linear temporal logic
Team semantics

ABSTRACT

We introduce and develop a set-based semantics for asynchronous TeamLTL. We consider two canonical logics in this setting: the extensions of TeamLTL by the Boolean disjunction and by the Boolean negation. We relate the new semantics with the original semantics based on multisets and establish one of the first positive complexity theoretic results in the temporal team semantics setting. In particular we show that both logics enjoy normal forms that can be utilised to obtain results related to expressivity and complexity (decidability) of the new logics.

1. Introduction

Linear temporal logic (LTL) is one of the most prominent logics for the specification and verification of reactive and concurrent systems. The core idea in model checking, as introduced in 1977 by Amir Pnueli [22], is to specify the correctness of a program as a set of infinite sequences, called traces, which define the acceptable executions of the system. In LTL-model checking one is concerned with trace sets that are definable by an LTL-formula. Ordinary LTL and its progeny are well suited for specification and verification of *trace properties*. These are properties of systems that can be checked by going through all executions of the system in isolation. A canonical example here is *termination*; a system terminates if each run of the system terminates. However not all properties of interest are trace properties. Many properties that are of prime interest, e.g., in information flow security, require a richer framework. The term *hyperproperty* was coined by Clarkson and Schneider [3] to refer to properties which relate multiple execution traces. An illustrative example is *bounded termination*; one cannot check whether a system terminates in bounded time by only checking traces in isolation. Checking hyperproperties is vital in information flow security where dependencies between secret inputs and publicly observable outputs of a system are considered potential security violations. Commonly known properties of that type are noninterference [24,20] and observational determinism [30]. Hyperproperties are not limited to the area of information flow control; e.g., distributivity and other system properties like fault tolerance can be expressed as hyperproperties [5].

During the past decade, the need for being able to formally specify hyperproperties has led to the creation of families of novel logics for this purpose, seeing as established temporal logics such as LTL can only specify trace properties. The

* Corresponding author at: University of Helsinki, Department of Mathematics and Statistics, Pietari Kalmin katu 5, 00014, Helsinki, Finland.
E-mail addresses: juha.kontinen@helsinki.fi (J. Kontinen), max.sandstrom@helsinki.fi (M. Sandström), j.t.virtema@sheffield.ac.uk (J. Virtema).

[1] Supported by the Academy of Finland grant 345634.
[2] Supported by the Academy of Finland grant 322795.
[3] Supported by the Academy of Finland grant 345634 and the DFG grant VI 1045/1-1.

https://doi.org/10.1016/j.ic.2025.105299

two main families of new logics are the so-called *hyperlogics* and logics that adopt *team semantics*. In the former approach, temporal logics such as LTL, computation tree logic (CTL), and quantified propositional temporal logic (QPTL) are extended with explicit trace and path quantification, resulting in logics like HyperLTL [2], HyperCTL* [2], and HyperQPTL [23,4]. The latter approach (which we adopt here) is to lift the semantics of temporal logics to sets of traces directly by adopting team semantics yielding logics such as TeamLTL [15,7] and TeamCTL [14,7].

Krebs et al. [15] introduced two versions of LTL with team semantics: a synchronous semantics and an asynchronous variant that differ on how the evolution of time is linked between computation traces when temporal operators are evaluated. In the synchronous semantics time proceeds in lock-step, while in the asynchronous variant time proceeds independently on each trace. For example the formula "Fterminate" (here F denotes the future-operator and "terminate" is a proposition depicting that a trace has terminated) defines the hyperproperty "bounded termination" under synchronous semantics, while it expresses the trace property "termination" under asynchronous semantics. The elegant definition of bounded termination exemplifies one of the main distinguishing factors of team logics from hyperlogics; namely the ability to refer directly to unbounded number of traces. Each hyperlogic-formula has a fixed number of trace quantifiers that delineate the traces involved in the evaluation of the formula. Another distinguishing feature of team logics lies in their ability to enrich the logical language with novel atomic formulae for stating properties of teams. The most prominent of these are the *dependence atom* $\mathrm{dep}(\bar{x}, \bar{y})$ (stating that the values of the variables $\bar{x}$ functionally determine the values of $\bar{y}$) and *inclusion atom* $\bar{x} \subseteq \bar{y}$ (expressing the inclusion dependency that all the truth value combinations occurring for $\bar{x}$ must also occur as truth value combinations for $\bar{y}$ in the order of the variables tuples).

As an example, let $o_1, \ldots, o_n$ be public observable bits and assume that c is a bit revealing confidential information. The atom $(o_1, \ldots o_n, c) \subseteq (o_1, \ldots o_n, \neg c)$ expresses a form of non-inference by stating that an observer cannot infer the value of the confidential bit from the outputs.

While HyperLTL and other hyperlogics have been studied extensively, many of the basic properties of TeamLTL are still not well understood. Krebs et al. [15] showed that synchronous TeamLTL and HyperLTL are incomparable in expressivity and that the asynchronous variant collapses to LTL. Not much was known about the complexity aspects of TeamLTL until Lück [18] showed that the complexity of satisfiability and model checking of synchronous TeamLTL with Boolean negation $\sim$ is equivalent to the decision problem of third-order arithmetic. Subsequently, Virtema et al. [29] embarked for a more fine-grained analysis of the complexity of synchronous TeamLTL and discovered a decidable syntactic fragment (the so-called *left-flat fragment*) and established that already a very weak access to the Boolean negation suffices for undecidability. They also showed that synchronous TeamLTL and its extensions can be translated to HyperQPTL$^+$, which is an extension of HyperLTL by (non-uniform) quantification of propositions. Kontinen and Sandström [12] defined translations between extensions of TeamLTL and the three-variable fragment of first-order team logic to utilise the better understanding of first-order team semantics. They also showed that any logic effectively residing between synchronous TeamLTL extended with the Boolean negation and second-order logic inherits the complexity properties of the extension of TeamLTL with the Boolean negation. Finally, Gutsfeld et al. [7] re-imagined the setting of temporal team semantics to be able to model richer forms of (a)synchronicity by developing the notion of time-evaluation functions. In addition to re-imagining the framework, they discovered decidable logics which however relied on restraining time-evaluation functions to be either *k-context-bounded* or *k-synchronous*. It is worth noting that recently asynchronous hyperlogics have been considered also in several other articles (see, e.g., [8,1]).

Almost all complexity theoretic results previously obtained for TeamLTL have been negative, and the few positive results have required drastic restrictions in syntax or semantics. In this article we take a fresh look at expressive extensions of asynchronous TeamLTL. Recent works on synchronous TeamLTL have revealed that quite modest extensions of synchronous TeamLTL are undecidable. Thus, our study of asynchronous TeamLTL partly stems from our desire to discover decidable, but expressive logics for hyperproperties.

Until now, all the papers on temporal team semantics have explicitly or implicitly adopted a semantics based on multisets of traces. In the team semantics literature, this often carries the name *strict semantics*, in contrast to *lax semantics* which is de facto a set-based semantics. Since, in the literature, hyperproperties are defined as sets of sets of traces (as opposed to sets of multisets of traces), a question arises: what would be a suitable set semantics for team-based logics? Note that the distinction between sets and multisets do not manifest in the synchronous team logics (in absence of quantitative atomic statements) and has thus so far remained unstudied.

In database theory, it is ubiquitous that tasks that are computationally easy under set based semantics become intractable in the multiset case. In the team semantics setting this can be already seen in the model checking problem of propositional inclusion logic, $\mathrm{PL}(\subseteq)$, which is P-complete under lax semantics, but NP-complete under strict semantics [10]. Our new set-based framework offers a setting that drops the accuracy that accompanies adoption of multiset semantics in favour of better computational properties. Consider the following formula expressing a form of strong non-inference in parallel computation: $\mathsf{G}((o_1, \ldots, o_n, c) \subseteq (o_1, \ldots, o_n, \neg c))$, where $o_1, \ldots, o_n$ are observable outputs and c is confidential. In the synchronous setting, the formula expresses that during a synchronous computation, at any given time, an observer cannot infer the value of the secret c from the outputs. In the asynchronous setting, the formula states a stronger property that the above property holds for all computations (not only synchronous). In the multiset setting the number of parallel computation nodes is fixed, while in the new lax semantics, we drop that restriction, and consider an undefined number of computation nodes. The condition is stronger in lax semantics; and intuitively easier to falsify, which makes model checking in practice easier.

Table 1
Expressivity hierarchy of the asynchronous logics considered in the paper. Logics with lax or strict semantics are here referred with the superscripts l and s, respectively. For the definitions of left flatness, quasi flatness, and left downward closure, we refer to Definitions 11 and 19. †: This follows since only TeamLTL$^l(\varovee)$ is downward closed (cf. Theorem 12 and Definition 19). Theorem 12 implies that for TeamLTL$(\sim)$-formulae in quasi-flat form the strict and lax semantics coincide.

TeamLTL$^{s/l}$		left-flat–TeamLTL$^s(\varovee)$	$\overset{\text{Cor. 16}}{<}$	TeamLTL$^s(\varovee)$
$\wedge$ Ex. 10		$\parallel\mid$ Thm. 12		
TeamLTL$^l(\varovee)$	$\overset{\text{Thm. 14}}{\equiv}$	left-flat–TeamLTL$^l(\varovee)$	$\overset{\dagger}{<}$	quasi-flat–TeamLTL$^{s/l}(\sim)$
				$\parallel\mid$ Thm. 20
				left-dc–TeamLTL$^l(\sim)$

Table 2
Complexity results of this paper. All results are completeness results if not otherwise specified. PL$(\sim)$ refers to the propositional fragment of TeamLTL$(\sim)$ which embeds also to left-dc-TeamLTL$^l(\sim)$. †: All PSPACE-completeness results for satisfiability in strict semantics and TeamLTL^l follow directly from classical LTL by downward closure and singleton equivalence similar to [15, Proposition 5.4]. ATIME-ALT(exp, poly) refers to alternating exponential time with a polynomial number of alternations while TOWER(poly) refers to problems that can be decided by a deterministic TM in time bounded by an exponential tower of 2's of polynomial height.

Logic	Complexity of		References
(asynchronous semantics)	model checking	satisfiability	
LTL	PSPACE	PSPACE	[25]
PL$(\sim)$	ATIME-ALT(exp, poly)	ATIME-ALT(exp, poly)	[9]
TeamLTL$^{l/s}$	PSPACE	PSPACE	[15], Theorem 9
left-flat-TeamLTL$^{s/l}(\varovee)$	PSPACE	PSPACE	Theorem 22
TeamLTL$^l(\varovee)$	PSPACE	PSPACE	Theorem 22
TeamLTL$^s(\varovee)$	???	PSPACE	†
TeamLTL^s(dep)	NEXPTIME-hard	PSPACE	[15]
left-dc-TeamLTL$^l(\sim)$	in TOWER(poly)	in TOWER(poly)	Theorem 22

Table 3
Complexity results for synchronous strict semantics. All results are completeness results if not otherwise specified. †: All PSPACE-completeness results for satisfiability follow directly from classical LTL by downward closure and singleton equivalence similar to [15, Proposition 5.4]. ‡: For the fragment without disjunction($\vee$).

Logic	Complexity of		References
(sync. strict semantics)	model checking	satisfiability	
TeamLTL	PSPACE‡	PSPACE	[15]
left-flat-TeamLTL$(\varovee)$	in EXPSPACE	PSPACE	[29]
TeamLTL(dep)	NEXPTIME-hard	PSPACE	[15]
TeamLTL$(\varovee)$	???	PSPACE	†
TeamLTL$(\varovee, \subseteq)$	Σ_1^0-hard	Σ_1^0-hard	[29]
TeamLTL$(\sim)$	third-order arithmetic	third-order arithmetic	[18]

Our contribution. We introduce and develop a set-based semantics for asynchronous TeamLTL, which we name *lax semantics* and write TeamLTL^l. We consider two canonical logics in this setting: the extensions of TeamLTL^l by the Boolean disjunction TeamLTL$^l(\varovee)$ and by the Boolean negation TeamLTL$^l(\sim)$. By developing the basic theory of lax asynchronous TeamLTL, we discover some fascinating connections between the strict and lax semantics. We discover that both logics enjoy normal forms that can be utilised to obtain expressivity and complexity results. Tables 1 and 2 summarise our results. For comparison, Table 3 summarises the known results on complexity of synchronous TeamLTL.

2. Preliminaries

We start of by defining the syntax common for the logics discussed in this article. Fix a finite set AP of *atomic propositions*. The set of formulae of LTL (over AP) is generated by the grammar:

$$\varphi ::= p \mid \neg p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc \varphi \mid \mathsf{G}\varphi \mid \varphi \mathsf{U} \varphi,$$

where $p \in$ AP. We adopt the convention that formulae are given in negation normal form, i.e., $\neg$ is allowed only in front of atomic propositions. Note that this set of LTL-formulae are expressively complete.

Next we define the structures for the semantics of the logic. As the name of the logic implies, the structures are linear Kripke models, where special consideration is given to how far in the chain a possible world resides. A *trace* t over AP is an infinite sequence from $(2^{\mathrm{AP}})^\omega$. For a natural number $i \in \mathbb{N}$, we denote by $t[i]$ the $(i+1)$th letter of t and by $t[i,\infty]$ the suffix $(t[j])_{j \geq i}$ of t. Semantics of LTL is defined in the usual manner (see e.g., [21]). For example, $t \models p$ if and only if $p \in t[0]$ and $t \models \bigcirc\varphi$ if and only if $t[1,\infty] \models \varphi$. The *truth value* of a formula φ on a trace t is denoted by $[\![\varphi]\!]_t \in \{0,1\}$. The logical constants $\top, \bot$ and the operators F and W can be defined in the usual way:

$$\bot := p \wedge \neg p,\ \top := p \vee \neg p,\ \mathsf{F}\varphi := \top\mathsf{U}\varphi, \text{ and } \varphi\mathsf{W}\psi := (\varphi\mathsf{U}\psi) \vee \mathsf{G}\varphi.$$

Next we present the so-called asynchronous team semantics for LTL introduced in [15]. In [15], the release operator was defined slightly erroneously; we fix the issue here by taking G as primitive and defining R using G and U. Informally, a multiset of traces T is a collection of traces with possible repetitions. Formally, we represent T as a set of pairs (i,t), where i is an *index* (from some suitable large set) and t is a trace. We stipulate that the elements of a multiset have distinct indices. When there is no risk of confusion, we omit the index and write t instead of (i,t). For multisets T and S, $T \uplus S$ denotes the disjoint union of T and S (obtained by stipulating that traces in S and T have disjoint sets of indices). Note that all the functions f with domain T are actually of the form $f((i,t))$ and may map different copies of the trace t differently. A *team* (*multiteam*, resp.) is a set (multiset, resp.) of traces. If $f\colon T \to \mathbb{N}$ is a function, we define the updated team $T[f,\infty] := \{t[f(t),\infty] \mid t \in T\}$, where f determines for each trace a point in time it updates to. For functions $f\colon T \to \mathbb{N}$ and $f'\colon T' \to \mathbb{N}$, we write $f' < f$, if $T' \subseteq T$ and $f'(t) < f(t)$ for all $t \in T'$. The underlying team $\mathrm{support}(T) := \{t \mid (i,t) \in T\}$ of a multiteam T is called the *support* of T.

Example 1. Suppose the team T consists of two traces of the same computation of alternating p and q states, i.e. $T := \{(1,t),(2,t)\}$, where $t := (\{p\}\{q\})^\omega$. See T pictured below:

$$\begin{cases} \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \dots \\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \dots \end{cases}$$

Now if we update the team with the function $f\colon T \to \mathbb{N}$ defined through $f((1,t)) = 1$ and $f((2,t)) = 2$, we obtain the team $T[f,\infty]$, which is the same as T, except the second trace is offset by one step. See the team $T[f,\infty]$ pictured below:

$$\begin{cases} \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \dots \\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \dots \end{cases}$$

Definition 2 *(Team semantics for* LTL*).* Let T be a multiteam, and φ and ψ LTL-formulae. The asynchronous team semantics of TeamLTL is defined as follows.

$$\begin{array}{lll}
T \models l & \Leftrightarrow & t \models l \text{ for all } t \in T,\ \text{ where } l \in \{p, \neg p \mid p \in \mathrm{AP}\} \text{ is a literal} \\
T \models \varphi \wedge \psi & \Leftrightarrow & T \models \varphi \text{ and } T \models \psi \\
T \models \varphi \vee \psi & \Leftrightarrow & \exists T_1, T_2 \text{ s.t. } T_1 \uplus T_2 = T \text{ and } T_1 \models \varphi \text{ and } T_2 \models \psi \\
T \models \bigcirc\varphi & \Leftrightarrow & T[1,\infty] \models \varphi,\ \text{ where 1 is the constant function } t \mapsto 1 \\
T \models \mathsf{G}\varphi & \Leftrightarrow & \forall f\colon T \to \mathbb{N} \quad T[f,\infty] \models \varphi \\
T \models \varphi\mathsf{U}\psi & \Leftrightarrow & \exists f\colon T \to \mathbb{N} \quad T[f,\infty] \models \psi \text{ and } T'[f',\infty] \models \varphi,\ \text{ for all } f'\colon T' \to \mathbb{N} \text{ s.t. } f' < f, \\
& & \text{where } T' := \{t \in T \mid f(t) \neq 0\}
\end{array}$$

The synchronous variant of the semantics is obtained by allowing f to range only over constant functions. We take the asynchronous semantics as the standard semantics and write TeamLTL for asynchronous TeamLTL.

Example 3. Consider the team $T := \{(1,t_1),(2,t_2),(3,t_2)\}$, where $t_1 := \{q\}^\omega$ and $t_2 := \{p\}\{p,q\}\{q\}^\omega$, and consider the formula $\varphi := p\mathsf{U}q$. The team T looks as below:

$$\begin{cases} \{q\}\ \ \{q\}\ \ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \end{cases}$$

If we think of the function $f\colon T \to \mathbb{N}$, defined through $f((1,t_1)) = 0$, $f((2,t_2)) = 1$, and $f((3,t_2)) = 2$, we notice that $t_1[f((1,t_1)),\infty] \models q$, $t_2[f((2,t_2)),\infty] \models q$, and $t_2[f((3,t_2)),\infty] \models q$, as seen in the representation of the updated team $T[f,\infty]$ below:

$$\begin{cases} \{q\}\ \ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{q\}\ \ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \end{cases}$$

Thus $T[f,\infty] \models q$. Furthermore, when we consider the functions defined on $T' := \{(2,t_2),(3,t_2)\}$, which are smaller than f, as described above, we notice that there are two possibilities; f_1 and f_2 defined as follows: $f_1((2,t_2)) = f_2((2,t_2)) = 0$, $f_1((3,t_2)) = 0$, and $f_2((3,t_2)) = 1$. Sketched out we get T' below:

$$\begin{cases} \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots, \end{cases}$$

the update of T' with f_1, $T'[f_1,\infty]$, below:

$$\begin{cases} \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots, \end{cases}$$

and finally the update of T' with f_2, $T'[f_2,\infty]$, below:

$$\begin{cases} \{p\}\ \ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p,q\}\ \ \{q\}\ \ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \end{cases}$$

Here we notice that $t_2[f_1((2,t_2)),\infty] \models p$, $t_2[f_2((2,t_2)),\infty] \models p$, $t_2[f_1((3,t_2)),\infty] \models p$, and $t_2[f_2((3,t_2)),\infty] \models p$. Therefore $T'[f_1,\infty] \models p$ and $T'[f_2,\infty] \models p$. By the asynchronous semantics of LTL then $T \models \varphi$.

We also consider the *Boolean disjunction* $\varovee$ and *Boolean negation* $\sim$ interpreted as usual:

$T \models \varphi \varovee \psi \Leftrightarrow (T \models \varphi$ or $T \models \psi)$, and

$T \models \sim\varphi \Leftrightarrow T \not\models \varphi$.

Next we define some important semantic properties of formulae studied in the literature. A logic has one of these properties if every formula of the logic has the property. It is easy to check that TeamLTL has all the properties listed below [15] whereas its extension with the Boolean disjunction has all but flatness and the extension with Boolean negation has none. The negative results transfer from the propositional case; it is easy to check that neither $p \varovee q$ nor $\sim p$ are flat, and that the latter also violates the empty team property and downward closure. Singleton equivalence is a meaningful property only for formulae that are syntactically LTL. The positive results can be proven via a straightforward induction on the structure of formulae. Furthermore, we will later establish in Theorem 4, TeamLTL with the Boolean disjunction is expressively complete for downward closed LTL-properties of teams, while the extension of TeamLTL with the Boolean negation can express all LTL-properties of teams (for a formal statement of this, see Theorem 4).

(Downward closure) If $T \models \varphi$ and $S \subseteq T$, then $S \models \varphi$.
(Empty team property) $\emptyset \models \varphi$.
(Flatness) $T \models \varphi$ iff $\{t\} \models \varphi$ for all $t \in T$.
(Singleton equivalence) $\{t\} \models \varphi$ iff $t \models \varphi$.

We will now justify our choice of semantics. The semantic rules for literals, conjunction, and disjunction are the standard ones in team semantics, and which have been motivated numerous times in the literature [26]. The two main desirable properties for the logic to have are flatness and singleton equivalence, which also motivated the original definition of asynchronous TeamLTL [15]. The given semantics for $\bigcirc$ is the only possible one that satisfies flatness. The same is true for F (i.e., $\top \mathsf{U} \varphi$) and G; moreover the semantics clearly capture the intuitive meanings of asynchronously in the future and asynchronously globally, respectively. The given semantics for U preserves flatness and singleton equivalence, and adequately captures the intuitive meaning of asynchronous until. The framework of asynchronous TeamLTL then allows us to define different variants of the familiar temporal operators. E.g., $\varphi \mathsf{W}_1 \psi := \mathsf{G}\varphi \vee \varphi \mathsf{U} \psi$ and $\varphi \mathsf{W}_2 \psi := \mathsf{G}\varphi \varovee \varphi \mathsf{U} \psi$ define different variants of *weak until*; the first of which is flat, while the second is not.

$$T \models \varphi \mathsf{W}_1 \psi \quad \Leftrightarrow \quad \exists T_1, T_2 \text{ s.t. } T_1 \uplus T_2 = T,\ T_1 \models \mathsf{G}\varphi \text{ and } T_2 \models \varphi \mathsf{U} \psi$$

$$T \models \varphi \mathsf{W}_2 \psi \quad \Leftrightarrow \quad T \models \mathsf{G}\varphi \text{ or } T \models \varphi \mathsf{U} \psi$$

Similarly $\varphi \mathsf{R}_1 \psi := \psi \mathsf{U}((\psi \wedge \varphi) \vee \mathsf{G}\psi)$ and $\varphi \mathsf{R}_2 \psi := \psi \mathsf{U}((\psi \wedge \varphi) \varovee \mathsf{G}\psi)$ give rise to different variants of *release*. Moreover, with the Boolean negation, $\sim$, one can define additional dual operators.

A defining feature of team semantics is the ability to enrich logics with novel atomic statements describing properties of teams in a modular fashion. For example, *dependence atoms* $\mathrm{dep}(\varphi_1,\dots,\varphi_n,\psi)$ and *inclusion atoms* $\varphi_1,\dots,\varphi_n \subseteq \psi_1,\dots,\psi_n$, with $\varphi_1,\dots,\varphi_n,\psi,\psi_1,\dots,\psi_n$ being LTL-formulae, have been studied extensively in first-order and modal team semantics. The dependence atom states that the truth value of ψ is functionally determined by that of $\varphi_1,\dots,\varphi_n$, whereas the inclusion atom states that each value combination of $\varphi_1,\dots,\varphi_n$ must also occur as a value combination for $\psi_1,\dots,\psi_n$. Formally:

$$T \models \mathrm{dep}(\varphi_1,\dots,\varphi_n,\psi) \text{ iff } \forall t,t' \in T : \Big(\bigwedge_{1\le j\le n} [\![\varphi_j]\!]_t = [\![\varphi_j]\!]_{t'}\Big) \Rightarrow [\![\psi]\!]_t = [\![\psi]\!]_{t'}$$

$$T \models \varphi_1, \ldots, \varphi_n \subseteq \psi_1, \ldots, \psi_n \text{ iff } \forall t \in T\, \exists t' \in T : \bigwedge_{1 \le j \le n} [\![\varphi_j]\!]_t = [\![\psi_j]\!]_{t'}$$

Consider the following exemplary formula: $\mathsf{G}\,\mathrm{dep}(i_1, i_2, o) \vee \mathsf{G}\,\mathrm{dep}(i_2, i_3, o)$. The formula states that the executions of the system can be decomposed into two parts; in the first part, the output o is determined by the inputs i_1 and i_2, and in the second part, o is determined by the inputs i_2 and i_3. Consider another formula $\mathrm{dep}(\bar{s}, \mathsf{F}a)$, where $\bar{s}$ indicates the positions (on/off) of a sequence of switches and a indicates whether an action occurs. Now the formula $\mathrm{dep}(\bar{s}, \mathsf{F}a)$ expresses the functional dependence that whether the action takes place somewhere in the future is functionally determined by the current positions of the switches.

If $\mathcal{A}$ is a collection of atoms and connectives, TeamLTL($\mathcal{A}$) denotes the extension of TeamLTL with the atoms and connectives in $\mathcal{A}$. It is straightforward to see (in analogy to the modal team semantics setting [11]) that any dependency such as the ones above is determined by a finite set of n-ary Boolean relations. Let B be a set of n-ary Boolean relations. We define the property $[\varphi_1, \ldots, \varphi_n]_B$ for an n-tuple $(\varphi_1, \ldots, \varphi_n)$ of LTL-formulae:

$$T \models [\varphi_1, \ldots, \varphi_n]_B \quad \text{iff} \quad \{([\![\varphi_1]\!]_t, \ldots, [\![\varphi_n]\!]_t) \mid t \in T\} \in B.$$

Expressions of the form $[\varphi_1, \ldots, \varphi_n]_B$ are *generalised atoms*. It was shown in [29] that, in the synchronous setting, TeamLTL($\sim$) is expressively complete with respect to all generalised atoms, whereas the extension of TeamLTL($\owedge$) with the so-called *flattening operator* can express any generalised atoms that preserve downward closure. Preserving downward closure means that if the atom is applied to a downward closed formula, the resulting formula remains downward closed. These results readily extend to the asynchronous setting. Moreover the flattening operator renders itself unnecessary due to flatness of asynchronous TeamLTL. The results imply, e.g., that the dependence atoms (which preserves downward closure) can be expressed in both of the logics TeamLTL($\sim$) and TeamLTL($\owedge$), and inclusion atoms in turn are expressible in TeamLTL($\sim$). The proof of the following theorem is essentially the same as the proof of [28, Proposition 17]. Below $L \equiv L'$ denotes the equi-expressivity of the logics L and L'.

Theorem 4. *Let $\mathcal{A}$, $\mathcal{D}$ be the sets of all generalised atoms, and all generalised atoms preserving downward closure, respectively. Then* TeamLTL($\mathcal{D}$, $\owedge$) $\equiv$ TeamLTL($\owedge$) *and* TeamLTL($\mathcal{A}$, $\sim$) $\equiv$ TeamLTL($\sim$).

As a consequence of the above theorem, we may focus our study to the two canonical logics TeamLTL($\owedge$) and TeamLTL($\sim$). As TeamLTL($\owedge$) can express all downward closed generalised atoms and TeamLTL($\sim$) can express all generalised atoms, our results concerning expressivity of logics can be readily extended to cover those classes of atoms, respectively. Moreover, our complexity theoretic results could also be extended to cover generalised atoms by using a suitable convention for deciding the input size a generalised atom contributes. Hence, for the rest of the paper, we focus on the logics TeamLTL($\owedge$) and TeamLTL($\sim$).

3. Set-based semantics for TeamLTL

Next we define a relaxed version of the asynchronous semantics. We call it *lax* semantics, as it corresponds to the so-called lax semantics of first-order team semantics (see e.g., [6]). From now on we refer to the semantics of Definition 2 as strict semantics. The possibility of considering lax semantics for TeamLTL was suggested by Lück already in [19], but the full definition was not given. Intuitively, lax semantics can always be obtained from a strict one by checking what strict semantics would yield if multiteams were enriched with unbounded many copies of each of its traces. More concretely, when designing a set-based semantics one may consider the restriction of multiset semantics where the only allowed multiteams are those T such that the multiplicities of $t \in T$ are $\aleph_0$. For instance, in the case of splitjunction, one would allow only those splits that yield subteams of the aforementioned property. One of the defining features of lax semantics is that it is unable to distinguish multiplicities, which is formalised by Proposition 8 below.

We need some notation for the new definition. We write $\mathcal{P}(\mathbb{N})^+$ to denote $\mathcal{P}(\mathbb{N}) \setminus \{\emptyset\}$. For a team T and function $f\colon T \to \mathcal{P}(\mathbb{N})^+$, we set $T[f, \infty] := \{t[s, \infty] \mid t \in T, s \in f(t)\}$. For $T' \subseteq T$, $f\colon T \to \mathcal{P}(\mathbb{N})^+$, and $f'\colon T' \to \mathcal{P}(\mathbb{N})^+$, we define that $f' < f$ if and only if

$$\forall t \in T'\colon \min(f'(t)) \le \min(f(t)) \text{ and, if } \max(f(t)) \text{ exists, } \max(f'(t)) < \max(f(t)).$$

Example 5. To illustrate how set based teams work, let us consider the same situation as in Example 1. Let T be the team only consisting of the trace $t := (\{p\}\{q\})^\omega$. See T pictured below:

$$\Big\{ \quad \{p\} \ \{q\} \ \{p\} \ \{q\} \ \{p\} \ \{q\} \ \{p\} \ \{q\} \ \ldots$$

Now if we update the team with the function $f\colon T \to \mathcal{P}(\mathbb{N})^+$ defined through $f(t) = \{1, 2\}$, we obtain the team $T[f, \infty]$, which is a team consisting of two traces, where the second trace is offset by one step. See the team $T[f, \infty]$ pictured below:

$$\begin{cases} \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \dots \\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \{q\}\ \{p\}\ \dots \end{cases}$$

Definition 6 *(TeamLTL^l).* Let T be a team, and φ and ψ TeamLTL-formulae. The lax semantics is defined as follows. We only list the cases that differ from the strict semantics.

$$\begin{array}{lcl} T \models^l \varphi \vee \psi & \Leftrightarrow & \exists T_1, T_2 \text{ s.t. } T_1 \cup T_2 = T \text{ and } T_1 \models \varphi \text{ and } T_2 \models \psi \\ T \models^l \mathsf{G}\varphi & \Leftrightarrow & \forall f : T \to \mathcal{P}(\mathbb{N})^+ \text{ it holds that } T[f,\infty] \models^l \varphi \\ T \models^l \varphi \mathsf{U} \psi & \Leftrightarrow & \exists f : T \to \mathcal{P}(\mathbb{N})^+ \text{ such that } T[f,\infty] \models^l \psi \text{ and} \\ & & \forall f' : T' \to \mathcal{P}(\mathbb{N})^+ \text{s.t. } f' < f, \text{ it holds that } T'[f',\infty] \models^l \varphi \text{ or } T' = \emptyset, \\ & & \text{where } T' := \{t \in T \mid \max(f(t)) \neq 0\} \end{array}$$

In the context we will be considering in this article, the subformulae φ in the definition of the until operator U always have the empty team property and thus we disregard the possibility that the team T' is empty in our proofs, as that case follows from the empty team property.

Example 7. Let's consider the same situation as in Example 3. To that end, let $T := \{t_1, t_2\}$, where $t_1 := \{q\}^\omega$ and $t_2 := \{p\}\{p,q\}\{q\}^\omega$, and consider the formula $\varphi := p\mathsf{U}q$. See below a sketch of team T:

$$\begin{cases} \{q\}\ \ \{q\}\ \ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \end{cases}$$

When we think of the function $f : T \to \mathcal{P}(\mathbb{N})^+$, defined through $f(t_1) = \{0\}$, and $f(t_2) = \{1,2\}$ we notice that $t_1[0,\infty] \models q$, $t_2[1,\infty] \models q$, and $t_2[2,\infty] \models q$, since the proposition symbol q is in state 0 of t_1 and in state 1 and 2 of trace t_2, as we can see from the updated team $T[f,\infty]$ pictured below:

$$\begin{cases} \ \{q\}\ \ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \end{cases}$$

Thus $T[f,\infty] \models^l q$. Furthermore, when we consider the functions defined on $T' := \{t_2\}$, which are smaller than f, when defined as described above, we notice that there are three possibilities; f_1, f_2, and f_3 defined as follows: $f_1(t_2) = \{0\}$, $f_2(t_2) = \{1\}$, and $f_3(t_2) = \{0,1\}$. Visually, we get the team T':

$$\{\ \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots,$$

and the updates $T'[f_1,\infty]$:

$$\{\ \{p\}\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots,$$

$T'[f_2,\infty]$:

$$\{\ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots,$$

and finally $T'[f_3,\infty]$:

$$\begin{cases} \ \{p\}\ \ \{p,q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \\ \{p,q\}\ \ \{q\}\ \ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \{q\}\ \dots \end{cases}$$

Here we notice that $t_2[0,\infty] \models p$, and $t_2[1,\infty] \models p$. Therefore, by the flatness of p it holds that $T'[f_1,\infty] \models p$, $T'[f_2,\infty] \models p$, and $T'[f_3,\infty] \models p$. Thus by the flatness of p and the lax asynchronous semantics of LTL $T \models^l \varphi$.

The above set-based semantics can also be viewed in terms of multisets. If we would like to define a multiset-based logic that would simulate the above set-based semantics, the functions $f : T \to \mathcal{P}(\mathbb{N})^+$ would need to be quantified uniformly. That is, we would restrict our consideration to functions where $f((i,t)) = f((j,t))$. Furthermore, the semantics for disjunction would have to be defined in a way that omits references to multiplicities. This logic would then have the property that a multiset-team satisfies a formula if and only if its support (which is a set-based team) satisfies the formula using the semantics of Definition 6.

In order to relate our new logics to the old multiteam based ones, we extend the lax semantics to multiteams T by stipulating that $T \models^l \varphi$ if and only if $\text{support}(T) \models^l \varphi$.

The following proposition shows that TeamLTL$^l(\sim)$ satisfies the so-called *locality* property. For a trace t over AP$'$ and AP $\subseteq$ AP$'$, the *reduction of* t to AP, $t_{\restriction \mathrm{AP}}$, is a sequence from $(2^{\mathrm{AP}})^\omega$ such that $p \in t[i]$ if and only if $p \in t_{\restriction \mathrm{AP}}[i]$, for all $p \in \mathrm{AP}$ and $i \in \mathbb{N}$. For a team T over AP$'$ we define the *reduction of* T to AP by $T_{\restriction \mathrm{AP}} = \{t_{\restriction \mathrm{AP}} \mid t \in T\}$.

Proposition 8. *Let T be a team and φ a* TeamLTL$^l(\sim)$*-formula with variables in* AP. *Now $T \models^l \varphi$ if and only if $T_{\upharpoonright \mathrm{AP}} \models^l \varphi$.*

Proof. The proof is by induction on the structure of φ.

Suppose φ is a literal. By definition, $T \models^l \varphi$ if only if $t \models \varphi$ for all $t \in T$. Now, seeing as φ is a literal of AP, the latter is equivalent with $t \models \varphi$ for all $t \in T_{\upharpoonright \mathrm{AP}}$. This, by definition, is equivalent with $T_{\upharpoonright \mathrm{AP}} \models^l \varphi$.

Suppose $\varphi = \psi_1 \wedge \psi_2$. By definition, $T \models^l \varphi$ if and only if $T \models^l \psi_1$ and $T \models^l \psi_2$. By the induction hypothesis, the latter holds if and only if $T_{\upharpoonright \mathrm{AP}} \models^l \psi_1$ and $T_{\upharpoonright \mathrm{AP}} \models^l \psi_2$, which by definition is equivalent with $T_{\upharpoonright \mathrm{AP}} \models^l \varphi$.

Suppose $\varphi = \psi' \vee \psi''$. By definition, $T \models^l \varphi$ if and only if there are subteams $T' \cup T'' = T$ such that $T' \models^l \psi'$ and $T'' \models^l \psi''$. By the induction hypothesis, the latter two claims are equivalent with $T'_{\upharpoonright \mathrm{AP}} \models^l \psi'$ and $T''_{\upharpoonright \mathrm{AP}} \models^l \psi''$. Now $T'_{\upharpoonright \mathrm{AP}} \cup T''_{\upharpoonright \mathrm{AP}} = T_{\upharpoonright \mathrm{AP}}$, hence $T_{\upharpoonright \mathrm{AP}} \models^l \varphi$. For the converse, assume that $T_{\upharpoonright \mathrm{AP}} \models^l \varphi$ is witnessed by subteams $T'_{\upharpoonright \mathrm{AP}}$ and $T''_{\upharpoonright \mathrm{AP}}$. Now it is easy to check that $T' = \{t \in T \mid t_{\upharpoonright \mathrm{AP}} \in T'_{\upharpoonright \mathrm{AP}}\}$ and $T'' = \{t \in T \mid t_{\upharpoonright \mathrm{AP}} \in T''_{\upharpoonright \mathrm{AP}}\}$ witness $T \models^l \varphi$.

Suppose $\varphi = \bigcirc \psi$. By definition, $T \models^l \varphi$ if and only if $T[1, \infty] \models^l \psi$, which, by the induction hypothesis, is equivalent with $T[1, \infty]_{\upharpoonright \mathrm{AP}} \models^l \psi$. Note that $T[1, \infty]_{\upharpoonright \mathrm{AP}} = T_{\upharpoonright \mathrm{AP}}[1, \infty]$, whereby $T \models^l \bigcirc \psi$ holds if and only if $T_{\upharpoonright \mathrm{AP}} \models^l \bigcirc \psi$ holds.

Suppose $\varphi = \mathsf{G}\psi$. By definition $T \models^l \varphi$ is equivalent with that $T[f, \infty] \models^l \psi$ for all functions $f : T \to \mathcal{P}(\mathbb{N}^+)$. By the induction hypothesis the latter is equivalent with $T[f, \infty]_{\upharpoonright \mathrm{AP}} \models^l \psi$ for functions f as before. Now for any $f' : T_{\upharpoonright \mathrm{AP}} \to \mathcal{P}(\mathbb{N}^+)$ there is some f, such that $T[f, \infty]_{\upharpoonright \mathrm{AP}} = T_{\upharpoonright \mathrm{AP}}[f', \infty]$, since we can pick the function $f(s) = f'(t)$ for all $s \in T$ such that $s_{\upharpoonright \mathrm{AP}} = t$. Similarly, for each f we obtain a corresponding f' by taking its restriction to AP. Hence $T[f, \infty] \models^l \psi$ holds for all f if and only if $T_{\upharpoonright \mathrm{AP}}[f', \infty] \models^l \psi$ holds for all f', and therefore $T \models^l \mathsf{G}\psi$ is equivalent with $T_{\upharpoonright \mathrm{AP}} \models^l \mathsf{G}\psi$.

Suppose $\varphi = \psi_1 \mathsf{U} \psi_2$. Assume $T \models^l \varphi$. By definition there is a function $f_2 : T \to \mathcal{P}(\mathbb{N}^+)$ such that $T[f_2, \infty] \models^l \psi_2$ and for all $f_1 < f_2$ it holds that $T^0[f_1, \infty] \models^l \psi_1$, where $T^0 := \{t \in T \mid \max(f(t)) \neq 0\}$. By the induction hypothesis then $T[f_2, \infty]_{\upharpoonright \mathrm{AP}} \models^l \psi_2$ and $T^0[f_1, \infty]_{\upharpoonright \mathrm{AP}} \models^l \psi_1$ for f_1 and f_2 as previously. We define the function $f_2' : T_{\upharpoonright \mathrm{AP}} \to \mathcal{P}(\mathbb{N}^+)$ by setting $f_2'(s) := \bigcup_i f_2(t^i)$, where $t^i \in T$ are such that $t^i_{\upharpoonright \mathrm{AP}} = s$. Now $T_{\upharpoonright \mathrm{AP}}[f_2', \infty] = T[f_2, \infty]_{\upharpoonright \mathrm{AP}}$. Furthermore, by a similarly defined f_1', we get that $T^0_{\upharpoonright \mathrm{AP}}[f_1', \infty] = T^0[f_1, \infty]_{\upharpoonright \mathrm{AP}}$. Thus $T[f_2, \infty]_{\upharpoonright \mathrm{AP}} \models^l \psi_2$ if and only if $T_{\upharpoonright \mathrm{AP}}[f_2', \infty] \models^l \psi_2$ and for all $f_1 < f_2$ it holds that $T^0[f_1, \infty]_{\upharpoonright \mathrm{AP}} \models^l \psi_1$ if and only if $T^0_{\upharpoonright \mathrm{AP}}[f_1, \infty] \models^l \psi_1$. Therefore $T_{\upharpoonright \mathrm{AP}} \models^l \varphi$. The converse follows analogously.

Suppose $\varphi = \sim \psi$. By definition $T \models^l \varphi$ if and only if $T \not\models^l \psi$, which in turn is equivalent with $T_{\upharpoonright \mathrm{AP}} \not\models^l \psi$ by the induction hypothesis. This, again, is equivalent with $T_{\upharpoonright \mathrm{AP}} \models^l \varphi$, due to the definition. $\square$

The next theorem displays that lax semantics enjoys the same fundamental properties as its strict counterpart.

Theorem 9. TeamLTL^l *satisfies downward closure, empty team property, singleton equivalence, and flatness.*

Proof. The proofs proceed by induction over the structure of the formulae. Note that while downward closure follows from flatness, we need that the induction steps work with the weaker assumption of downward closure for the result to generalise to non-flat extensions of the logic.

Downward closure: Let $\varphi \in$ TeamLTL be a formula and T, S teams such that $S \subseteq T$ and $T \models^l \varphi$. We need to show that $S \models^l \varphi$ as well.

For atomic φ the claim is immediately true: $T \models^l \varphi$ if and only if $t \models \varphi$ for all $t \in T$, which also holds for all $t \in S$, and thus $S \models^l \varphi$.

For conjunction, the claim follows immediately from the induction hypothesis. Let's consider the case of disjunction. Suppose $T \models^l \varphi \vee \psi$. By definition then there are $T_1, T_2 \subseteq T$ s.t. $T_1 \cup T_2 = T$, $T_1 \models^l \varphi$ and $T_2 \models^l \psi$. By the induction hypothesis $S \cap T_1 \models^l \varphi$ and $S \cap T_2 \models^l \psi$, since $S \cap T_1 \subseteq T_1$ and $S \cap T_2 \subseteq T_2$. Furthermore $(S \cap T_1) \cup (S \cap T_2) = S$, and therefore $S \models^l \varphi \vee \psi$.

The case for $\bigcirc$ is straightforward. Suppose $T \models^l \bigcirc\varphi$. By definition $T[1, \infty] \models^l \varphi$. Since $S[1, \infty] \subseteq T[1, \infty]$ follows from $S \subseteq T$, we obtain $S[1, \infty] \models^l \varphi$ by the induction hypothesis. Thus $S \models \bigcirc\varphi$.

Next we suppose $T \models^l \mathsf{G}\varphi$. By definition then for all $f : T \to \mathcal{P}(\mathbb{N})^+$ it holds that $T[f, \infty] \models^l \varphi$. Now, for all $f : S \to \mathcal{P}(\mathbb{N})^+$, $S[f, \infty] \subseteq T[f', \infty]$, where f' is any extension of f to T. Hence by the induction hypothesis $S[f, \infty] \models^l \varphi$ for all f and $S \models^l \mathsf{G}\varphi$.

Suppose then that $T \models^l \varphi \mathsf{U} \psi$. For any function $h : T \to \mathcal{P}(\mathbb{N})^+$, let h_S denote the reduct of h to the domain S. From $S \subseteq T$, we get

$$S[h_S, \infty] \subseteq T[h, \infty]. \tag{1}$$

By definition, there is a function $f : T \to \mathcal{P}(\mathbb{N})^+$ such that $T[f, \infty] \models^l \psi$. Moreover, for all $f' : T_0 \to \mathcal{P}(\mathbb{N})^+$ such that $f' < f$, we have $T_0[f', \infty] \models^l \varphi$, where $T_0 := \{t \in T \mid \max(f(t)) \neq 0\}$. By the induction hypothesis and (1), we have that $S[f_S, \infty] \models^l \psi$. Moreover, for all $g : S_0 \to \mathcal{P}(\mathbb{N})^+$, where $S_0 := \{t \in S \mid \max(f_S(t)) \neq 0\}$, such that $g < f_S$, $S[g, \infty] \models^l \varphi$ follows by the induction hypothesis and the fact that every g is equal to f'_S for a suitable f'. Thus $S \models^l \varphi \mathsf{U} \psi$.

Empty team property: Suppose $T = \emptyset$. The claim is clear for atomic formulae; since the team is empty, $p \in t(0)$ and $p \notin t(0)$ holds for all $p \in$ AP and $t \in T$. The cases for conjunction and disjunction follow immediately from the induction hypothesis. The cases for temporal operators follow immediately from the induction hypotheses as well, since $\emptyset[f,\infty]=\emptyset$ for any $f\colon T\to\mathcal{P}(\mathbb{N})^+$.

Flatness: We prove by induction on φ that $T\models\varphi$ if $\{t\}\models\varphi$ for all $t\in T$, for every team T. The only if direction follows directly from downward closure.

The case for atomic formulae holds by definition, and the cases for conjunction and the next step operator follows directly from the induction hypothesis. Let us consider the case for disjunction. Suppose φ and ψ are LTL formulae, and consider the formula $\varphi\vee\psi$. Assume that for all $t\in T$, $\{t\}\models\varphi$ or $\{t\}\models\psi$. Let T_1 and T_2 be the sets of traces in T that satisfy φ and ψ, respectively. Clearly $T_1\cup T_2=T$, and by induction hypothesis $T_1\models^l\varphi$ and $T_2\models^l\psi$. Thus $T\models^l\varphi\vee\psi$.

For the case of until, suppose $\{t\}\models^l\varphi\mathsf{U}\psi$ for all $t\in T$. Now, for each $t\in T$, there exists a function $f_t\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$ such that $\{t\}[f_t,\infty]\models\psi$ and for all intermediary $f_t'\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$, defined for such traces t where $\max(f(t))\neq 0$, such that $f_t'<f_t$ it holds that $\{t\}[f_t',\infty]\models\varphi$. We define the functions $f\colon T\to\mathcal{P}(\mathbb{N})^+$, and $g\colon T'\to\mathcal{P}(\mathbb{N})^+$ through $f(t):=f_t(t)$ and $g(t):=\{j\in\mathbb{N}\mid j<\sup(f(t))\}$. By the induction hypothesis $T[f,\infty]\models^l\psi$ and $T'[g,\infty]\models^l\varphi$. Let $f'\colon T'\to\mathcal{P}(\mathbb{N})^+$ be any function such that $f'<f$. If we can show that $T'[f',\infty]\models^l\psi$, we obtain $T\models\varphi\mathsf{U}\psi$ and are done. To that end, clearly $T'[f',\infty]\subseteq T'[g,\infty]$, and thus we obtain $T'[f',\infty]\models^l\psi$ from downward closure.

Finally for the case for G, suppose $\{t\}\models^l\mathsf{G}\varphi$ for all $t\in T$. Now for each trace t and function $f\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$ it holds that $\{t\}[f,\infty]\models^l\varphi$. Now by the induction hypothesis, for all functions $F\colon T\to\mathcal{P}(\mathbb{N})^+$ such that $F(t):=f(t)$ it holds that $T[F,\infty]\models^l\varphi$. Now the functions F are all possible functions. Thus $T\models^l\mathsf{G}\varphi$.

Singleton equivalence: We prove by induction on φ that, for every trace t, $\{t\}\models^l\varphi$ if and only if $t\models\varphi$.

The case for literals is stated in the definition, whereas the case for conjunction follows directly from the induction hypothesis. Hence, consider the next step operator. Suppose $\varphi=\bigcirc\psi$. Now by definition $\{t\}\models^l\varphi$ is equivalent with $\{t\}[1,\infty]\models^l\psi$, which in turn holds if and only if $t[1,\infty]\models\psi$ by induction hypothesis. By the definition of the next step operator, the latter is equivalent with $t\models\bigcirc\psi$.

Suppose $\varphi=\psi\vee\theta$. By definition $\{t\}\models^l\varphi$ is equivalent with $\{t\}\models^l\psi$ or $\{t\}\models^l\theta$. By the induction hypothesis, the latter two are equivalent with $t\models\psi$ or $t\models\theta$, and thus by definition $t\models\psi\vee\theta$ if and only if $\{t\}\models^l\varphi$.

Suppose $\varphi=\psi_1\mathsf{U}\psi_2$. Assume $\{t\}\models^l\varphi$. By definition there is a function $f\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$ such that $\{t\}[f,\infty]\models^l\psi_2$ and for all intermediary functions $f'\colon T'\to\mathcal{P}(\mathbb{N})^+$ it holds that $T'[f',\infty]\models^l\psi_1$, where $T':=\{t\}$, if $f(t)\neq\{0\}$ and otherwise $T'=\emptyset$. We assume $f(t)\neq\{0\}$, as the other case is trivial. Let $k:=\min(f(t))$. By induction hypothesis and downward closure $t[k,\infty]\models\psi_2$. Now for every singleton-valued function $f'\colon T'\to\mathcal{P}(\mathbb{N})^+$ defined by $f'(t):=\{k'\}$, such that $k'<k$, it holds that $\{t\}[f',\infty]\models^l\psi_1$. Hence by the induction hypothesis, for all $k'<k$ it holds that $t[k',\infty]\models\psi_1$. Thus $t\models\varphi$.

Now assume $t\models\varphi$. By definition there exists a number $k\geq 0$ such that $t[k,\infty]\models\psi_2$ and for all $0\leq k'<k$ it holds that $t[k',\infty]\models\psi_1$. Thus we can define a function $f\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$ such that $f(t):=\{k\}$ and functions $f'\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$ such that $f'(t):=\{k'\}$. Now by the induction hypothesis $\{t\}[f,\infty]\models^l\psi_2$ and $\{t\}[f',\infty]\models^l\psi_1$, and furthermore by flatness the latter actually holds for all intermediary functions g. Therefore $\{t\}\models^l\varphi$.

Suppose $\varphi=\mathsf{G}\psi$. Assume $\{t\}\models^l\varphi$. By definition for all functions $f\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$ it holds that $\{t\}[f,\infty]\models^l\psi$. Especially this holds for every function f_k such that $f_k(t):=\{k\}$. By the induction hypothesis then $t[k,\infty]\models\psi$ for all k. Thus by definition $t\models\mathsf{G}\psi$.

Now assume $t\models\varphi$. By definition for all $k\geq 0$ it holds that $t[k,\infty]\models\psi$. By the induction hypothesis it follows that $\{t\}[f,\infty]\models^l\psi$ for all $f\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$ such that $f=\{k\}$. Thus, by flatness $\{t\}[f',\infty]\models^l\psi$ for all functions $f'\colon\{t\}\to\mathcal{P}(\mathbb{N})^+$. Thus $\{t\}\models^l\mathsf{G}\psi$. $\square$

The following example establishes that the new lax semantics differs from the strict semantics, and that in the old semantics multiplicities matter. Moreover, we obtain $\mathrm{TeamLTL}^l<\mathrm{TeamLTL}^l(\varovee)$ by showing that the latter is not flat.

Example 10. Let φ be the formula $\mathsf{G}(p\varovee q)$, $T_1:=\{t\}$ and $T_2:=\{(1,t),(2,t)\}$, where $t:=\{p\}\{q\}^\omega$. It is easy to check that $T_1\models\varphi$ but $T_1\not\models^l\varphi$ (which is witnessed by $T_1[f,\infty]\not\models^l p\varovee q$ for $f(t):=\{0,1\}$). Likewise, T_2 is not a model for φ under the strict semantics, as the multiple copies of the trace can be updated independently, which may lead to one of them satisfying p and the other q. In other words, $T_2\not\models\varphi$, as for instance the function f defined by $f((1,t))=0$ and $f((2,t))=1$ is such that $T_2[f,\infty]\not\models p$ and $T_2[f,\infty]\not\models q$. Moreover, if we let $s_1:=\{p\}^\omega$ and $s_2:=\{q\}^\omega$, we have that $\{s_i\}\models^l\varphi$, for $i\in\{1,2\}$, but $\{s_1,s_2\}\not\models^l\varphi$.

We will also consider the following fragments of $\mathrm{TeamLTL}(\varovee)$ and $\mathrm{TeamLTL}(\sim)$.

Definition 11. A formula φ of $\mathrm{TeamLTL}(\varovee)$ is called *left-flat*, if in all of its subformulae of the form $\mathsf{G}\psi$ and $\psi\mathsf{U}\theta$, the subformula ψ is a LTL-formula. A formula φ of $\mathrm{TeamLTL}(\sim,\varovee)$ is called *left-downward closed*, if in all of its subformulae of the form $\mathsf{G}\psi$ and $\psi\mathsf{U}\theta$, the subformula ψ is an $\mathrm{TeamLTL}(\varovee)$-formula.

We will later show that the above syntactic restriction for flatness could be replaced by a semantic restriction (see Corollary 15).

Theorem 12. *For all $\varphi\in\mathrm{TeamLTL}^l(\varovee)$ the following two claims hold:*

1. *φ is downward closed and has the empty team property, and*
2. *if φ is left-flat, then $T \models \varphi$ iff* support$(T) \models^l \varphi$ *for all multiteams T.*

Proof. In order to show (1), it suffices to extend the proofs of Theorem 9 with a case for $\varovee$. For downward closure: Let T be a team of traces and let $S \subseteq T$. Suppose $T \models^l \psi \varovee \varphi$. By definition $T \models^l \psi$ or $T \models^l \varphi$. Without loss of generality, we may assume $T \models^l \psi$, which entails by the induction hypothesis that $S \models^l \psi$ and thus, by definition, $S \models^l \psi \varovee \varphi$. For the empty team property the claim follows immediately from the induction hypothesis.

The proof of claim (2) is a simple induction on the structure of φ. We show the claim for $\varphi = \alpha \mathsf{U} \psi$, where α is an LTL-formula. Assume $T \models \varphi$. Then there exists $f\colon T \to \mathbb{N}$ such that $T[f,\infty] \models \psi$ and $T'[f',\infty] \models \alpha$, for all $f'\colon T' \to \mathbb{N}$ such that $f' < f$, where $T' = \{t \in T \mid f(t) \neq 0\}$. By flatness then for all $(i,t) \in T$ and $k < f((i,t))$ it holds that $\{(i, t[k,\infty])\} \models \alpha$. Define $F\colon T \to \mathcal{P}(\mathbb{N})^+$ by $F(t) := \{f((i,t)) \mid (i,t) \in T\}$. It is easy to check that support$(T[f,\infty]) =$ support$(T)[F,\infty]$. Now by application of the induction hypothesis support$(T)[F,\infty] \models^l \psi$. Pick then an arbitrary $F'\colon T' \to \mathcal{P}(\mathbb{N})^+$ such that $F' < F$ and a trace $t \in T$ such that $F(t) \neq \{0\}$. Consider $k \in F'(t)$. As $F' < F$, it follows that $k < \max(F(t))$, and thus by the previous note $\{(i, t[k,\infty])\} \models \alpha$. Thus, by the flatness of α and the induction hypothesis support$(T)[F',\infty] \models^l \alpha$. As F' was arbitrary, the support of T, updated with any function smaller than F satisfy α. The proof of the converse implication is similar.

Assume support$(T) \models^l \varphi$ and let $G\colon$ support$(T) \to \mathcal{P}(\mathbb{N})^+$ be such that support$(T)[G,\infty] \models^l \psi$. By downward closure we may assume that G is single valued. Now it is easy to pick $g\colon T \to \mathbb{N}$ such that support$(T[g,\infty]) =$ support$(T)[G,\infty]$. From the induction hypothesis it follows that $T[g,\infty] \models \psi$. Just like above, using the fact that α is flat, it follows that $T \models \varphi$. $\square$

The restriction to left-flat formulae in case (2) above is necessary by Example 10.

4. Normal forms for TeamLTL with Boolean disjunction and negation

In this section we develop normal forms for our logics, which we then utilise to obtain strong expressivity and complexity results.

Definition 13. A formula φ is in $\varovee$-disjunctive normal form if it is of the form

$$\bigvee_{i \in I} \alpha_i,$$

where α_i are LTL-formulae.

Every formula of TeamLTL$^l(\varovee)$ can be transformed into an equivalent $\varovee$-disjunctive normal form. This result is similar to the one proved in [27] for team-based modal logic ML$(\varovee)$. In the following $|\varphi|$ denotes the length of the formula φ.

Theorem 14. *Every $\varphi \in$ TeamLTL$^l(\varovee)$ is logically equivalent to a formula $\varphi^* = \bigvee_{i \in I} \alpha_i$ in $\varovee$-disjunctive normal form, where $|\alpha_i| \leq |\varphi|$ and $|I| = 2^k$, where k is the number of $\varovee$ in φ.*

Proof. The proof proceeds by induction on the structure of formulae. Note that atomic formulae are already in the normal form and that the case for $\varovee$ is trivial. The remaining cases are defined as follows:

$$(\psi \wedge \theta)^* := \bigvee_{i \in I, j \in J} (\alpha_i^\psi \wedge \alpha_j^\theta) \qquad (\psi \vee \theta)^* := \bigvee_{i \in I, j \in J} (\alpha_i^\psi \vee \alpha_j^\theta)$$

$$(\bigcirc \psi)^* := \bigvee_{i \in I} \bigcirc \alpha_i^\psi \qquad (\mathsf{G}\psi)^* := \bigvee_{i \in I} \mathsf{G}\alpha_i^\psi$$

$$(\psi \mathsf{U} \theta)^* := \bigvee_{i \in I, j \in J} (\alpha_i^\psi \mathsf{U} \alpha_j^\theta).$$

where α_i^ψ and α_j^θ are the flat formulae in the disjunctive normal forms of ψ and θ respectively, and I and J are the respective index sets.

Suppose $\varphi = \psi \wedge \theta$ and that $\psi \equiv \bigvee_{i \in I} \alpha_i^\psi$ and $\theta \equiv \bigvee_{i \in J} \alpha_j^\theta$ (induction hypothesis). Now $T \models^l \varphi$ if and only if $T \models^l \psi$ and $T \models^l \theta$. The latter holds, if and only if $T \models^l \alpha_k^\psi$ and $T \models^l \alpha_{k'}^\theta$, for some $k \in I$ and $k' \in J$. This is equivalent with $T \models^l \alpha_k^\psi \wedge \alpha_{k'}^\theta$, for some $k \in I$ and $k' \in J$. Finally, this can be equivalently expressed as $T \models^l \bigvee_{i,j} (\alpha_i^\psi \wedge \alpha_j^\theta)$, i.e. $T \models^l \varphi^*$.

Suppose $\varphi = \psi \vee \theta$ and that $\psi \equiv \bigvee_{i \in I} \alpha_i^\psi$ and $\theta \equiv \bigvee_{i \in J} \alpha_j^\theta$. By definition $T \models^l \varphi$ if and only if there exists $T' \cup T'' = T$ such that $T' \models^l \psi$ and $T'' \models^l \theta$. By the induction hypothesis the latter is equivalent with $T' \models^l \bigvee_{i \in I} \alpha_i^\psi$ and $T'' \models^l \bigvee_{j \in J} \alpha_j^\theta$.

By definition this holds if and only if there are $k' \in I$ and $k'' \in J$ such that $T' \models^l \alpha^{\psi}_{k'}$ and $T'' \models^l \alpha^{\theta}_{k''}$, which is equivalent with $T \models^l \alpha^{\psi}_{k'} \vee \alpha^{\theta}_{k''}$, by definition. Equivalently then $T \models^l \varovee_{i\in I, j\in J}(\alpha^{\psi}_i \vee \alpha^{\theta}_j)$.

Suppose $\varphi = \bigcirc\psi$ and that $\psi \equiv \varovee_{i\in I} \alpha^{\psi}_i$. By definition $T \models^l \varphi$ is equivalent with $T[1,\infty] \models^l \psi$. By the induction hypothesis the latter holds if and only if $T[1,\infty] \models^l \varovee_{i\in I} \alpha^{\psi}_i$, which by definition is equivalent with $T[1,\infty] \models^l \alpha^{\psi}_k$ for some $k \in I$. The latter holds if and only if $T \models^l \bigcirc\alpha^{\psi}_k$ for some $k \in I$, which is equivalent with $T \models^l \varovee_{i\in I} \bigcirc\alpha^{\psi}_i$.

Suppose $\varphi = \mathsf{G}\psi$ and that $\psi \equiv \varovee_{i\in I} \alpha^{\psi}_i$. Suppose that $T \models^l \varphi$. By definition for all functions $f: T \to \mathcal{P}(\mathbb{N})^+$ it holds that $T[f,\infty] \models^l \psi$. By the induction hypothesis $T[f,\infty] \models^l \varovee_{i\in I} \alpha^{\psi}_i$ for all f. Especially this holds for the total function defined for every $t \in T$ by $f_{max}(t) := \mathbb{N}$. Thus $T[f_{max},\infty] \models^l \alpha^{\psi}_k$ for some $k \in I$. By downward closure it holds that $T[f',\infty] \models^l \alpha^{\psi}_k$ for all $f': T \to \mathcal{P}(\mathbb{N})^+$. Hence $T \models^l \mathsf{G}\alpha^{\psi}_k$, and thus $T \models^l \varovee_{i\in I} \mathsf{G}\alpha^{\psi}_i$. The other direction is analogous.

Suppose $\varphi = \psi\mathsf{U}\theta$ and that $\psi \equiv \varovee_{i\in I} \alpha^{\psi}_i$ and $\theta \equiv \varovee_{j\in J} \alpha^{\theta}_j$. Suppose $T \models^l \varphi$. By definition there exists a function $f: T \to \mathcal{P}(\mathbb{N})^+$ such that $T[f,\infty] \models^l \theta$ and for all functions $f': T' \to \mathcal{P}(\mathbb{N})^+$ such that $f' < f$, $T'[f',\infty] \models^l \psi$, where $T' := \{t \in T \mid f(t) \neq 0\}$. Hence by the induction hypothesis $T[f,\infty] \models^l \varovee_{j\in J} \alpha^{\theta}_j$, which is equivalent with $T[f,\infty] \models^l \alpha^{\theta}_k$ for some $k \in J$, and, for the function $f_{max}: T' \to \mathcal{P}(\mathbb{N})^+$ defined through $f_{max}(t) := \{n \in \mathbb{N} \mid n < m,$ for some $m \in f(t)\}$ (which is well-defined, as $f(t) \neq \{0\}$ for all $t \in T'$), it holds that $T'[f_{max},\infty] \models^l \varovee_{i\in I} \alpha^{\psi}_i$, which in turn is equivalent with $T'[f_{max},\infty] \models^l \alpha^{\psi}_{k'}$ for some $k' \in I$. By downward closure the latter holds for all intermediary functions, and thus $T \models^l \alpha^{\psi}_{k'}\mathsf{U}\alpha^{\theta}_k$ and finally $T \models^l \varovee_{i\in I, j\in J}(\alpha^{\theta}_i \mathsf{U}\alpha^{\psi}_j)$ as wanted. The converse is analogous.

For showing the size estimates stated in the theorem, it suffices to note that our translation to $\varovee$-disjunctive normal from can be equivalently stated:

$$\varphi \equiv \bigvee\!\!\!\!\bigcirc_{i\in I} \alpha^{\psi}_i \equiv \bigvee\!\!\!\!\bigcirc_{f\in F} \varphi^f,$$

where F is the set of all selection functions f that select, separately for each occurrence, either the left disjunct ψ or the right disjunct θ of each subformula of the form $\psi \varovee \theta$ of φ, and φ^f denotes the formula obtained from φ by substituting each occurrence of a subformula of type $(\psi \varovee \theta)$ by $f(\psi \varovee \theta)$. The size estimates follow immediately from this observation. $\square$

Using this normal form we can now show that the flat fragment of TeamLTL$^l(\varovee)$ is subsumed by TeamLTL^l, which means that using the Boolean disjunction does not expand the set of flat formulae. In other words, the logics TeamLTL$^l(\varovee)$ and TeamLTL^l define exactly the same trace properties.

Corollary 15. *For every flat* TeamLTL$^l(\varovee)$*-formula there exists an equivalent* TeamLTL^l*-formula.*

Proof. Let $\varphi \in$ TeamLTL$^l(\varovee)$ be flat, and let $\varovee_i \psi_i$ be an equivalent formula given by Theorem 14, where ψ_i are TeamLTL^l-formulae. The following equivalences hold:

$$T \models^l \varphi \quad\Leftrightarrow\quad \forall t\in T : \{t\} \models^l \bigvee\!\!\!\!\bigcirc_{i} \psi_i \quad\Leftrightarrow\quad \forall t\in T : \{t\} \models^l \bigvee_i \psi_i \quad\Leftrightarrow\quad T \models^l \bigvee_i \psi_i$$

The first equivalence follows from flatness of φ and since $\varovee_i \psi_i$ is equivalent to φ. The second equivalence follows, for it is easy to check that, for logics that have the empty team property, $\varovee_i$ and $\bigvee_i$ are interchangeable over singleton teams. The last equivalence follows from the flatness of TeamLTL^l (Theorem 9). $\square$

The normal form also helps us clarify the hierarchy between the lax and the strict semantics of LTL extended with the Boolean disjunction, where the strict semantics is strictly more expressive.

Corollary 16. TeamLTL$^l(\varovee) <$ TeamLTL$(\varovee)$.

Proof. Let φ be a TeamLTL$^l(\varovee)$-formula. By Theorem 14, φ can be equivalently written as a disjunction $\varovee_i \alpha_i$ of TeamLTL^l-formulae. Now, for each multiteam T,

$$\mathrm{support}(T) \models^l \varphi \quad\Leftrightarrow\quad \mathrm{support}(T) \models^l \bigvee\!\!\!\!\bigcirc_{i} \alpha_i \quad\Leftrightarrow\quad T \models \bigvee\!\!\!\!\bigcirc_{i} \alpha_i,$$

where the last equivalence is due to Theorem 12 and the fact that the formulae α_i are left-flat (since they are LTL-formulae). Hence, for any given TeamLTL$^l(\varovee)$-formula, the normal form formula $\varovee_i \alpha_i$ is, in fact, an equivalent TeamLTL$(\varovee)$ formula, from which TeamLTL$^l(\varovee) \leq$ TeamLTL$(\varovee)$ follows.

For showing the strict inclusion, we generalise the result from Example 10 and show that for the TeamLTL($\varovee$) formula $\mathsf{G}(p \varovee q)$ there exists no equivalent TeamLTL$^l(\varovee)$ formula. For a contradiction, suppose that $\varphi \in$ TeamLTL$^l(\varovee)$ is equivalent with $\mathsf{G}(p \varovee q)$. By Theorem 14, we may assume that φ is a disjunction $\bigovee_i \alpha_i$ of n TeamLTL^l-formulae. Define $t_i := \{p\}^i\{q\}^\omega$ for $i \leq n+1$. Now clearly $\{(1,t_i)\} \models \mathsf{G}(p \varovee q)$ and thus $\{t_i\} \models^l \varphi$, for each i. By the semantics of $\varovee$ this implies that for each i there exists $j_i \leq n$ such that $\{t_i\} \models^l \alpha_{j_i}$. Now from the pigeonhole principle, there exists $1 \leq k < l \leq n+1$ such that $j_k = j_l$. Thus $\{t_k\} \models^l \alpha_{j_k}$ and $\{t_l\} \models^l \alpha_{j_k}$, from which $\{t_k, t_l\} \models^l \alpha_{j_k}$ follows, by flatness of α_{j_k}. Thus $\{t_k, t_l\} \models^l \varphi$ and $\{(1,t_k),(1,t_l)\} \models \mathsf{G}(p \varovee q)$, which is clearly false. $\square$

The following corollary is also a direct consequence of Theorem 14.

Corollary 17. *The operator G distributes over the Boolean disjunction $\varovee$ for* TeamLTL($\varovee$)*-formulae.*

The following example shows that the above corollary does not hold in general, specifically for formulae that are not downward closed.

Example 18. Let φ be the formula $\mathsf{G}(\sim\neg p_1 \varovee \sim\neg p_2)$ and $T := \{t\}$, where $t := (\{p_1\}\{p_2\})^\omega$. It is now easy to check that $T \models^l \varphi$ but $T \not\models^l \mathsf{G}\sim\neg p_i$ for $i \in \{1,2\}$.

A normal form, similar to the one in Theorem 14, can also be obtained for TeamLTL($\sim$). However, since the extension is not downward closed, it only holds for a specific fragment of the logic. The following normal form has been introduced and used in [17,16] to analyse the complexity of modal team logic and FO^2 in the team semantics context. Below φ^d denotes a formula obtained by transforming $\neg\varphi$ into negation normal form in the standard way in LTL.

Definition 19. A formula φ is quasi-flat if φ is of the form:

$$\bigovee_{i\in I}(\alpha_i \wedge \bigwedge_{j\in J_i} \exists\beta_{i,j}),$$

where α_i and $\beta_{i,j}$ are LTL-formulae, and $\exists\beta_{i,j}$ is an abbreviation for the formula $\sim\beta^d_{i,j}$.

Note that, for LTL-formulae α and β, we have $T \models^l \alpha$ if and only if $t \models \alpha$, for all $t \in T$. Moreover $T \models^l \exists\beta$, if and only if there exists some trace $t \in T$ such that $t \models \beta$.

Theorem 20. *Every left-downward closed formula $\varphi \in$* TeamLTL$^l(\sim, \varovee)$ *is logically equivalent to a quasi-flat formula φ^*.*

Proof. Proof by induction over the structure of φ. Atoms are flat, and hence are in the normal form. The translations and the proofs of correctness for the cases of conjunction, disjunction, and Boolean negation are analogous to the simpler modal framework of [17,16].

Suppose $\varphi = \psi \wedge \theta$ and assume that ψ is equivalent to $\bigovee_{i\in I}(\alpha_i^\psi \wedge \bigwedge_{j\in J_i}\exists\beta^\psi_{i,j})$ and θ to $\bigovee_{i\in I'}(\alpha_i^\theta \wedge \bigwedge_{j\in J'_i}\exists\beta^\theta_{i,j})$. By the distributive laws of conjunction and disjunction, φ is clearly equivalent to

$$\bigovee_{i\in I, k\in I'}(\alpha_i^\psi \wedge \alpha_k^\theta \wedge \bigwedge_{j\in J_i}\exists\beta^\psi_{i,j} \wedge \bigwedge_{j\in J'_k}\exists\beta^\theta_{k,j}).$$

Suppose $\varphi = \psi \vee \theta$. By the induction hypothesis and an argument analogous to the disjunction case of the proof of Theorem 14, φ is equivalent to

$$\bigovee_{i\in I, k\in I'}\big((\alpha_i^\psi \wedge \bigwedge_{j\in J_i}\exists\beta^\psi_{i,j}) \vee (\alpha_k^\theta \wedge \bigwedge_{j\in J'_k}\exists\beta^\theta_{k,j})\big). \tag{2}$$

The above formula expresses that T can be split into two parts: T_1 in which each trace satisfies α_i and the subformulae $\beta_{i,j}$ are satisfied by some traces, and T_2 in which each trace satisfies α_k and the subformulae $\beta_{k,j}$ are satisfied by some traces. But this is equivalent to saying that T can be split into two parts: T_1 in which each trace satisfies α_i, and T_2 in which each trace satisfies α_k; and the subformulae $\alpha_i \wedge \beta_{i,j}$ and $\alpha_k \wedge \beta_{k,j}$ are satisfied by some traces in T, and thus the formula (2) is equivalent with

$$\bigovee_{i\in I, k\in I'}\big((\alpha_i^\psi \vee \alpha_k^\theta) \wedge \bigwedge_{j\in J_i}\exists(\alpha_i^\psi \wedge \beta^\psi_{i,j}) \wedge \bigwedge_{j\in J'_k}\exists(\alpha_j^\theta \wedge \beta^\theta_{k,j})\big)$$

that is in the normal form.

Suppose $\varphi = \sim\psi$ and assume as induction hypothesis that ψ is equivalent to the formula $\varovee_{i\in I}(\alpha_i \wedge \bigwedge_{j\in J_i} \exists\beta_{i,j})$. Now φ is clearly equivalent to the formula

$$\bigwedge_{i\in I}(\exists\alpha_i^d \varovee \bigvee_{j\in J_i} \beta_{i,j}^d).$$

This formula can be expanded back to the normal form with exponential blow-up using the distributivity law of propositional logic.

Suppose $\varphi = \bigcirc\psi$ and assume that ψ is equivalent to $\varovee_{i\in I}(\alpha_i \wedge \bigwedge_{j\in J_i} \exists\beta_{i,j})$. It is now easy to check that φ is equivalent to $\varovee_{i\in I}(\bigcirc\alpha_i \wedge \bigwedge_{j\in J_i} \exists\bigcirc\beta_{i,j})$.

Suppose $\varphi = \mathsf{G}\psi$. Since φ is left-downward closed, ψ and hence $\mathsf{G}\psi$ are TeamLTL$^l(\varovee)$-formulae. By Theorem 14, φ can equivalently be written in the form $\varovee_i \alpha_i$, where α_i are LTL-formulae.

Suppose $\varphi = \psi\mathsf{U}\theta$. By assumption φ is left-downward closed, hence ψ is equivalent with a formula of the form $\varovee_{i\in I}\alpha_i^\psi$ (by the previous theorem) and θ is equivalent to $\varovee_{k\in I'}(\alpha_k^\theta \wedge \bigwedge_{j\in J_k}\exists\beta_{k,j}^\theta)$. Now using the fact that ψ is downward closed, it is easy to see that φ is logically equivalent with the formula:

$$\varovee_{i\in I,k\in I'} \big(\alpha_i^\psi \mathsf{U}(\alpha_k^\theta \wedge \bigwedge_{j\in J_k} \exists\beta_{k,j}^\theta)\big). \tag{3}$$

It now suffices to show that the disjuncts (for any $i \in I, k \in I'$) of (3) can be equivalently expressed as:

$$\big(\alpha_i^\psi \mathsf{U}\alpha_k^\theta \wedge \bigwedge_{j\in J_k} \exists(\alpha_i^\psi \mathsf{U}(\alpha_k^\theta \wedge \beta_{k,j}^\theta))\big). \tag{4}$$

We will show the logical implication from (4) to (3). Assume

$$T \models^l \big(\alpha_i^\psi \mathsf{U}\alpha_k^\theta \wedge \bigwedge_{j\in J_k} \exists(\alpha_i^\psi \mathsf{U}(\alpha_k^\theta \wedge \beta_{k,j}^\theta))\big).$$

Let f be such that $T[f,\infty] \models^l \alpha_k^\theta$ and that $T[g,\infty] \models^l \alpha_i^\psi$, for all $g < f$. In order to show

$$T \models^l \alpha_i^\psi \mathsf{U}(\alpha_k^\theta \wedge \bigwedge_{j\in J_k} \exists\beta_{k,j}^\theta), \tag{5}$$

we need to make sure that traces witnessing the truth of the formulae $\exists\beta_{k,j}^\theta$ can be found in $T[f,\infty]$. Here we can use the assumption that $T \models^l \bigwedge_{j\in J_k}\exists(\alpha_i^\psi \mathsf{U}(\alpha_k^\theta \wedge \beta_{k,j}^\theta))$ implying that for each $j \in J_k$ there exists $t_j \in T$ such that $t_j \models \alpha_i^\psi \mathsf{U}(\alpha_k^\theta \wedge \beta_{k,j}^\theta)$. Let now n_j be such that $t_j[n_j,\infty] \models \alpha_k^\theta \wedge \beta_{k,j}^\theta$ and that $t_j[l,\infty] \models \alpha_i^\psi$ for all $l < n_j$. Now by the flatness of the formulae α_i^ψ, α_k^θ, and $\beta_{k,j}^\theta$, the function f' defined by

$$f'(t) := \begin{cases} f(t) \cup \{t_j[n_j,\infty]\} & \text{if } t = t_j, \text{ for some } j \in J_k \\ f(t) & \text{otherwise} \end{cases}$$

witnesses (5). The converse is proved analogously. □

5. Computational properties

In this section we analyse the computational properties of the logics studied in the previous section. We focus on the complexity of the model checking and satisfiability problems.

For the model checking problem, one has to determine whether a team of traces generated by a given finite Kripke structure satisfies a given formula. We consider Kripke structures of the form $\mathrm{K} := (W, R, \eta, w_0)$, where W is a finite set of states, $R \subseteq W^2$ a left-total transition relation, $\eta\colon W \to 2^{\mathrm{AP}}$ a labelling function, and $w_0 \in W$ an initial state of K. A path σ through K is an infinite sequence $\sigma \in W^\omega$ such that $\sigma[0] := w_0$ and $(\sigma[i], \sigma[i+1]) \in R$ for every $i \geq 0$. The *trace of* σ is defined as $t(\sigma) := \eta(\sigma[0])\eta(\sigma[1])\cdots \in (2^{\mathrm{AP}})^\omega$. A Kripke structure K then *generates* the trace set $\mathrm{Traces(K)} := \{t(\sigma) \mid \sigma \text{ is a path through K}\}$.

Definition 21. The *model checking problem* of a logic $\mathcal{L}$ is the following decision problem: Given a formula $\varphi \in \mathcal{L}$ and a Kripke structure K over AP, determine whether $Traces(K) \models \varphi$. The *(countable) satisfiability problem* of a logic $\mathcal{L}$ is the following decision problem: Given a formula $\varphi \in \mathcal{L}$, determine whether $T \models \varphi$ for some (countable) $T \neq \emptyset$.

Below we will use the fact that the model checking and satisfiability problems of LTL are PSPACE-complete [25]. Furthermore, we use the facts that the satisfiability problem of propositional team logic, PL($\sim$), is ATIME-ALT(exp, poly)-complete [9], and that the complexity of modal team logic is complete for the class $\mathrm{TOWER(poly)} := \mathrm{TIME}(\exp_{n^{O(1)}}(1))$, where $\exp_0(1) := 1$ and $\exp_{k+1}(1) := 2^{\exp_k(1)}$ [17,16]. Recall that ATIME-ALT(exp, poly) refers to alternating exponential time with a polynomial number of alternations.

Theorem 22.

1. *The model checking and satisfiability problems of* $\mathrm{TeamLTL}^l(\varovee)$ *are* PSPACE-*complete.*
2. *The model checking and satisfiability problems of the left-flat fragment of* $\mathrm{TeamLTL}(\varovee)$ *are* PSPACE-*complete.*
3. *The model checking problem of the left-downward closed fragment of* $\mathrm{TeamLTL}^l(\sim, \varovee)$ *is* PSPACE-*hard and it is contained in* TOWER(poly).
4. *The satisfiability problem of the left-downward closed fragment of* $\mathrm{TeamLTL}^l(\sim, \varovee)$ *is contained in* TOWER(poly) *and it is* ATIME-ALT(exp, poly)-*hard.*

Proof. Let us first consider the proofs of claims 1 and 2. Note that PSPACE-hardness holds already for LTL-formulae, hence it suffices to show containment in PSPACE. Furthermore, note that 2 follows immediately from 1 and Theorem 12. Assume a formula $\varphi \in \mathrm{TeamLTL}^l(\varovee)$ and a Kripke structure K is given as input. By Theorem 14, φ is logically equivalent with a formula of the form $\bigvee\!\!\!\!\bigcirc_{f \in F} \varphi^f$, where f varies over selection functions choosing, separately for each occurrence, either the left disjunct ψ or the right disjunct θ of each subformula of the form $\psi \varovee \theta$ of φ. Now, without constructing the full formula $\bigvee\!\!\!\!\bigcirc_{f \in F} \varphi^f$, using polynomial space with respect to the size of φ, it is possible to check whether $\mathrm{Traces(K)} \models \varphi_f$ for some $f \in F$. Hence the upper bound follows from the fact that LTL model checking is in PSPACE. The upper bound for satisfiability follows analogously.

Let us then consider the proof of claim (4). The proof of claim (3) is analogous. For the lower bound, it suffices to note that propositional team logic PL($\sim$) is a fragment of the left-downward closed fragment of $\mathrm{TeamLTL}^l(\sim, \varovee)$ and hence its satisfiability problem can be trivially reduced to the satisfiability problem of the left-downward closed fragment. Therefore ATIME-ALT(exp, poly)-hardness follows by the result of [9].

For the upper bound, we first transform an input formula φ into an equivalent quasi-flat formula of the form $\bigvee\!\!\!\!\bigcirc_{i \in I}(\alpha_i \wedge \bigwedge_{j \in J_i} \exists \beta_{i,j})$. Analogously to [17,16], this formula can be computed in time $\mathrm{TIME}(\exp_{O(|\varphi|)}(1))$. It is now easy to see that the quasi-flat formula is satisfiable if and only if there exists $i \in I$, such that $\mathrm{SAT}(\alpha_i \wedge \beta_{i,j}) = 1$ for all $j \in J_i$. Since LTL-satisfiability checking is contained in $\mathrm{PSPACE} \subseteq \mathrm{TIME}(2^{n^{O(1)}})$, the overall complexity of the above procedure is in $\mathrm{TIME}(\exp_{(|\varphi|^{O(1)})}(1))$. $\square$

6. Connections to other forms of asynchronicity

In [7] the authors introduced a novel team-based logic that can deal with different modes of asynchronous hyperproperties by using so-called *time evaluation functions* (tefs). Time evaluation functions facilitate fine-grained asynchronous interactions between traces. Intuitively, given a trace $t \in T$ and a value of the global clock $i \in \mathbb{N}$, a tef τ outputs the value $\tau(i, t)$ of the local clock of trace t at global time i. If T is a multiset of traces, a *time evaluation function* for T is a function $\tau : \mathbb{N} \times T \to \mathbb{N}$ that satisfies the following two conditions. We write $\tau(i)$ to denote the function $T \to \mathbb{N}$ defined by $t \mapsto \tau(i, t)$.

- *stepwiseness* – $\forall i \in \mathbb{N}\, \forall t \in T : \tau(i+1, t) \in \{\tau(i, t), \tau(i, t) + 1\}$,
- *strict monotonicity* – $\forall i \in \mathbb{N} : \tau(i) \neq \tau(i+1)$.

A tef is *initial*, if $\tau(0, t) = 0$ for each $t \in T$.

It was shown in [7] that when tefs are assumed to be synchronous, we obtain exactly synchronous TeamLTL as defined in [15]. In this section, we take a closer look on the connections between asynchronous TeamLTL and team-based logics with tefs. We identify a logic with tefs that corresponds almost exactly to asynchronous TeamLTL and to the left-flat fragment of asynchronous TeamLTL($\varovee$). This connection establishes the first non-trivial decidability result for logics with tefs without putting heavy restrictions on tefs.

We give the syntax of TeamCTL with an additional synchronous next operator $\bigcirc$ that was shown in [7] to be expressible in the logic with the help of the Boolean disjunction $\varovee$ and an additional proposition symbol that is set to alternate uniformly and synchronously (see [7, Theorem 4.2] for details).

$$\varphi ::= p \mid \neg p \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \bigcirc \varphi \mid \bigcirc_\exists \varphi \mid \bigcirc_\forall \varphi \mid \mathsf{G}_\exists \varphi \mid \mathsf{G}_\forall \varphi \mid \varphi \mathsf{U}_\exists \psi \mid \varphi \mathsf{U}_\forall \psi$$

Next we define the semantics. Note that, while in [7] TeamCTL*-formulae were evaluated with respect to pairs (T, τ), we consider only TeamCTL-formulae in this article, and therefore we choose to internalise τ into T. The cases for the semantics of the propositional atoms, Boolean connectives, and $\bigcirc$ are the same as for asynchronous TeamLTL (see Definition 2). Note that here the functions $\tau(i)$ take the role of the functions f of Definition 2.

$$
\begin{aligned}
T \models \bigcirc_{\exists}\varphi \quad &\Leftrightarrow \quad \text{there is an initial tef } \tau \text{ s.t. } T[\tau(1),\infty] \models \varphi\\
T \models \bigcirc_{\forall}\varphi \quad &\Leftrightarrow \quad \text{for all initial tefs } \tau\text{, we have } T[\tau(1),\infty] \models \varphi\\
T \models \mathsf{G}_{\exists}\varphi \quad &\Leftrightarrow \quad \text{there is an initial tef } \tau \text{ s.t. } T[\tau(k),\infty] \models \varphi\text{, for all } k \in \mathbb{N}\\
T \models \mathsf{G}_{\forall}\varphi \quad &\Leftrightarrow \quad \text{for all initial tefs } \tau\text{, we have } T[\tau(k),\infty] \models \varphi\text{, for all } k \in \mathbb{N}\\
T \models \varphi\mathsf{U}_{\exists}\psi \quad &\Leftrightarrow \quad \text{there is an initial tef } \tau \text{ and } k \in \mathbb{N} \text{ s.t. } T[\tau(k),\infty] \models \psi \text{ and}\\
&\qquad\qquad \forall m: 0 \le m < k \Rightarrow T[\tau(m),\infty] \models \varphi\\
T \models \varphi\mathsf{U}_{\forall}\psi \quad &\Leftrightarrow \quad \text{for all initial tefs } \tau\text{, } \exists k \in \mathbb{N} \text{ s.t. } T[\tau(k),\infty] \models \psi \text{ and}\\
&\qquad\qquad \forall m: 0 \le m < k \Rightarrow T[\tau(m),\infty] \models \varphi
\end{aligned}
$$

We identify a collection of the above temporal operators that match as closely as possible with the operators of asynchronous TeamLTL. We will not be utilising all of the operators introduced above, but chose to introduce a full selection to emphasise that in this setting all temporal operators have two variants; existential and universal. For a collection of temporal operators $\mathcal{C}$, we write TeamCTL($\mathcal{C}$) to denote the logic built from propositional atoms by using $\wedge$, $\vee$, and the operators in $\mathcal{C}$.

In order to deal with the asynchronous until operator, we need to do two concessions. Firstly, we need to restrict ourselves to the left-flat fragment (cf. Definition 11). Secondly, instead of until, we use the strong release operator $\psi\,\mathsf{M}\,\varphi$ with the following semantics:

$$T \models^l \psi\,\mathsf{M}\,\varphi \Leftrightarrow \exists f: T \to \mathcal{P}(\mathbb{N})^+ \text{ s.t } T[f,\infty] \models^l \psi \text{ and } T[f',\infty] \models^l \varphi \text{ for all } f': T \to \mathcal{P}(\mathbb{N})^+ \text{ s.t. } f' \le f,$$

where $f' \le f$ if and only if

$$\forall t \in T: \min(f'(t)) \le \min(f(t)) \text{ and, if } \max(f(t)) \text{ exists, } \max(f'(t)) \le \max(f(t)).$$

The F and $\bigcirc$ modalities can be used without any restrictions. It is now easy to check that $\psi\,\mathsf{M}\,\varphi$ and $\varphi\mathsf{U}(\varphi \wedge \psi)$ are equivalent for all flat φ.

Lemma 23. *If φ and ψ are TeamLTL^l-formulae and φ is flat, then $\psi\,\mathsf{M}\,\varphi$ and $\varphi\mathsf{U}(\varphi \wedge \psi)$ are equivalent (in TeamLTL^l).*

Proof. Let φ and ψ as described in the formulation of the lemma and let T be an arbitrary team. Now, by the semantics of weak release,

$$T \models^l \psi\,\mathsf{M}\,\varphi \Leftrightarrow \exists f: T \to \mathcal{P}(\mathbb{N})^+ \text{ s.t } T[f,\infty] \models^l \psi \text{ and } T[f',\infty] \models^l \varphi \text{ for all } f': T \to \mathcal{P}(\mathbb{N})^+ \text{ s.t. } f' \le f.$$

Now since φ is flat, the right-hand side of the above equivalence is equivalent with

$$\exists f: T \to \mathcal{P}(\mathbb{N})^+ \text{ s.t } T[f,\infty] \models^l \psi \text{ and } T[f_{\max},\infty] \models^l \varphi,$$

where $f_{\max}$ is defined such that $f_{\max}(t) = \{i \in \mathbb{N} \mid i \le j \text{ for some } j \in f(t)\}$. Again, by flatness of φ, the above is equivalent with

$$
\begin{aligned}
&\exists f: T \to \mathcal{P}(\mathbb{N})^+ \text{ such that } T[f,\infty] \models^l \varphi \wedge \psi \text{ and}\\
&\forall f': T' \to \mathcal{P}(\mathbb{N})^+ \text{ s.t. } f' < f\text{, it holds that } T'[f',\infty] \models^l \varphi \text{ or } T' = \emptyset,\\
&\text{where } T' := \{t \in T \mid \max(f(t)) \neq 0\},
\end{aligned}
$$

where $f' < f$ is as defined for Definition 6. Finally, by the semantics of until, the above is equivalent with $T \models^l \varphi\mathsf{U}(\varphi \wedge \psi)$. □

Finally, we say that two formulae φ and ψ are *fin-equivalent*, if $T \models \varphi \Leftrightarrow T \models \psi$ holds for all finite multiteams T. With these restrictions, we can prove an equivalence between left-flat-TeamCTL($\bigcirc$, $\mathsf{G}_{\forall}$, $\mathsf{M}_{\exists}$, $\varovee$) and TeamLTL^l($\varovee$). Here $\psi\,\mathsf{M}_{\exists}\,\varphi$ is defined as $\varphi\mathsf{U}_{\exists}(\varphi \wedge \psi)$.

We first need to show that the normal form for TeamLTL^l($\varovee$) still holds when U is replaced with M (cf. Theorem 14).

Theorem 24. *Every TeamLTL^l($\varovee$)-formula φ using M instead of U is logically equivalent to a formula $\varphi^* = \bigvarovee_{i \in I} \alpha_i$ in $\varovee$-disjunctive normal form using M instead of U, where $|\alpha_i| \le |\varphi|$ and $|I| = 2^k$, where k is the number of $\varovee$ in φ.*

Proof. We modify the proof for Theorem 14 by describing the case for M. Otherwise the proof is identical.

Suppose $\varphi = \theta \mathsf{M} \psi$ and that $\psi \equiv \bigovee_{i \in I} \alpha_i^{\psi}$ and $\theta \equiv \bigovee_{j \in J} \alpha_j^{\theta}$. Suppose $T \models^l \theta \mathsf{M} \psi$. By definition there exists a function $f : T \to \mathcal{P}(\mathbb{N})^+$ such that $T[f, \infty] \models^l \theta$ and for all functions $f' : T \to \mathcal{P}(\mathbb{N})^+$ such that $f' \leq f$, $T[f', \infty] \models^l \psi$. Hence by the induction hypothesis $T[f, \infty] \models^l \bigovee_{j \in J} \alpha_j^{\theta}$, which is equivalent with $T[f, \infty] \models^l \alpha_k^{\theta}$ for some $k \in J$, and $T[f_{\max}, \infty] \models^l \bigovee_{i \in I} \alpha_i^{\psi}$, which in turn is equivalent with $T[f_{\max}, \infty] \models^l \alpha_{k'}^{\psi}$ for some $k' \in I$. By downward closure $T[f', \infty] \models^l \alpha_{k'}^{\psi}$ holds for all intermediary functions $f' \leq f$. Thus $T \models^l \alpha_k^{\theta} \mathsf{M} \alpha_{k'}^{\psi}$ and finally $T \models^l \bigovee_{i \in I, j \in J} (\alpha_i^{\theta} \mathsf{M} \alpha_j^{\psi})$ as required. The converse is analogous. $\square$

Theorem 25. *For every left-flat-*TeamCTL($\bigcirc$, $\mathsf{G}_\forall$, $\mathsf{M}_\exists$, $\bigovee$)*-formula there exists a fin-equivalent formula of* TeamLTL^l($\bigovee$) *using* M *instead of* U*, and vice versa.*

Proof. By Theorem 24, every TeamLTL^l($\bigovee$)-formula using M instead of U is equivalent to some left-flat-TeamLTL^l($\bigovee$)-formula using M instead of U. Furthermore, by Lemma 23, this equivalence can be extended to left-flat-TeamLTL^l($\bigovee$)-formulae where until is restricted to occur in the form $\psi \mathsf{U}(\psi \wedge \theta)$. Finally, Theorem 12 extends the equivalence to left-flat-TeamLTL^s($\bigovee$)-formulae where until is restricted to occur in the form $\psi \mathsf{U}(\psi \wedge \theta)$. Hence, we prove the equivalence between left-flat-TeamCTL($\bigcirc$, $\mathsf{G}_\forall$, $\mathsf{M}_\exists$, $\bigovee$) and left-flat-TeamLTL^s($\bigovee$) where until is restricted to occur in the form $\psi \mathsf{U}(\psi \wedge \theta)$.

The translations simply swap $\varphi \mathsf{U}(\varphi \wedge \psi)$ with $\psi \mathsf{M}_\exists \varphi$ and $\mathsf{G}\varphi$ with $\mathsf{G}_\forall \varphi$. Correctness of the translations can be proven by induction on the structure of formulae. The only non-trivial cases are the cases for strong release and globally.

The case for globally follows from the following chain of equivalences:

$$\begin{aligned} T \models \mathsf{G}\varphi \quad &\Leftrightarrow \quad \forall f : T \to \mathbb{N} \quad T[f, \infty] \models \varphi \\ &\Leftrightarrow \quad \text{for all initial tefs } \tau\text{, we have } T[\tau(k), \infty] \models \varphi\text{, for all } k \in \mathbb{N} \\ &\Leftrightarrow \quad T \models \mathsf{G}_\forall \varphi. \end{aligned}$$

The first and the last equivalence are simply the semantics of the respective operators. The second equivalence follows from the assumption that φ is flat.

Assume $\varphi \mathsf{U}(\varphi \wedge \psi)$ is such that φ is flat. Now, by the semantics of until,

$$\begin{aligned} T \models \varphi \mathsf{U}(\varphi \wedge \psi) \quad \Leftrightarrow \quad &\exists f : T \to \mathbb{N} \text{ such that } T[f, \infty] \models \varphi \wedge \psi \text{ and} \\ &\forall f' : T' \to \mathbb{N} \text{ s.t. } f' < f\text{, we have } T'[f', \infty] \models \varphi \\ &\text{where } T' := \{t \in T \mid \max(f(t)) \neq 0\}. \end{aligned}$$

Now, since φ is flat, the right-hand-side of the above equivalence is equivalent with

$$\exists f : T \to \mathbb{N} \text{ such that } T[f, \infty] \models \varphi \wedge \psi \text{ and } \{t[i, \infty]\} \models \varphi, \text{ for every } t \in T \text{ and } i \leq f(t)$$

By flatness of φ together with the induction hypothesis, the above is equivalent with the statement that

$$\text{there is an initial tef } \tau \text{ and } k \in \mathbb{N} \text{ s.t. } T[\tau(k), \infty] \models \varphi \wedge \psi \text{ and } \forall m : 0 \leq m < k \Rightarrow T[\tau(m), \infty] \models \varphi,$$

which, by the semantics of $\mathsf{U}_\exists$ is equivalent with $T \models \varphi \mathsf{U}_\exists(\varphi \wedge \psi)$, which can be rewritten as $T \models \psi \mathsf{M}_\exists \varphi$. The correspondence between quantification of tefs and functions of the form $f : T \to \mathbb{N}$ relies on the fact that T is finite. In Example 28 we show a situation where the correspondence breaks down due to the team T being infinite. $\square$

Corollary 26. *For every* TeamCTL($\bigcirc$, $\mathsf{G}_\forall$, $\mathsf{M}_\exists$)*-formula there exists a fin-equivalent* TeamLTL*-formula using* M *instead of* U*, and vice versa. (Note that the logics* TeamLTL *and* TeamLTL^l *are equi-expressive by Theorem 12.)*

By combining Theorem 25 to Theorems 12 and 22, we obtain the following:

Corollary 27. *The model checking problem of left-flat-*TeamCTL($\bigcirc$, $\mathsf{G}_\forall$, $\mathsf{M}_\exists$, $\bigovee$) *restricted to finite teams is* PSPACE*-complete.*

We showed that over finite sets of traces the left-flat fragment of TeamCTL($\bigcirc$, $\mathsf{G}_\forall$, $\mathsf{M}_\exists$, $\bigovee$) coincides with the left-flat fragment of TeamLTL($\bigovee$) using M instead of U. The following example shows that the simple translation given in the proof does not work over arbitrary sets of traces.

Example 28. Let T consist of the traces $t_k = \{p\}^k\{p, q\}\{q\}^\omega$, $k \in \mathbb{N}$. Let $f : T \to \mathcal{P}(\mathbb{N})^+$ be defined such that $f(t_k) = \{k\}$, for each $k \in \mathbb{N}$. Clearly $T[f, \infty] \models^l q$ and $T[f', \infty] \models^l p$ for all $f' : T \to \mathcal{P}(\mathbb{N})^+$ such that $f' \leq f$. Hence $T \models^l q \mathsf{M} p$ in asynchronous TeamLTL. In contrast, $T \not\models q \mathsf{M}_\exists p$ in TeamCTL, since there is no tef τ such that $T[\tau(m), \infty] \models q$ for some m.

7. Conclusion

We introduced a novel set-based semantics for asynchronous TeamLTL. We showed several results on the expressive power and complexity of the extensions of TeamLTL^l by the Boolean disjunction $\text{TeamLTL}^l(\varovee)$ and by the Boolean negation $\text{TeamLTL}^l(\sim)$. In particular, our results show that the complexity properties of the former logic are comparable to that of LTL and that the left-downward closed fragment of the latter also has decidable model-checking and satisfiability problems. See Table 1 on page 3 for an overview of our expressivity results and Table 2 for our complexity results. We obtained these results on $\text{TeamLTL}^l(\varovee)$ and $\text{TeamLTL}^l(\sim)$ via normal forms that also allowed us to relate the expressive power of these logics to the corresponding logics in the strict semantics. Our results show that, while the synchronous TeamLTL can be viewed as a fragment of second-order logic, the asynchronous $\text{TeamLTL}(\varovee)$ under the lax semantics is a sublogic of HyperLTL (see [2] for a definition). In fact, subsequent work [13] has revealed that $\text{TeamLTL}^l(\varovee)$ is equiexpressive with the closure of universal one variable fragment of HyperLTL with conjunctions and disjunctions. Moreover, it was shown in [13] that the expressivity of left-downward closed $\text{TeamLTL}^l(\sim)$ coincides with that of the Boolean closure of one variable HyperLTL. Furthermore, our decidability results show, e.g., that it will probably be possible to devise a complete proof system for $\text{TeamLTL}^l(\varovee)$. Section 6 relates and applies our results to recently defined logics whose asynchronicity is formalised via time evaluation functions [7]. We conclude with open questions:

- Does Theorem 20 extend to all formulae of $\text{TeamLTL}^l(\sim)$? Note that any quasi-flat–$\text{TeamLTL}(\sim)$-formula can be rewritten in HyperLTL [13, Theorem 13].
- Can the result (iii) of Theorem 22 be accompanied by a matching lower bound (i.e., TOWER(poly)-hardness result)?
- Can a syntactic characterisation (similar to Corollary 15) be obtained for the downward closed fragment of $\text{TeamLTL}^l(\sim)$? We believe that $\text{TeamLTL}^l(\varovee)$ is a promising candidate, as its extensions with infinite conjunctions and disjunctions suffice for all downward closed properties of teams.
- What is the complexity of model checking for $\text{TeamLTL}(\varovee)$ under the strict semantics?

CRediT authorship contribution statement

Juha Kontinen: Writing – review & editing, Writing – original draft, Formal analysis, Conceptualization. **Max Sandström:** Writing – review & editing, Writing – original draft, Formal analysis, Conceptualization. **Jonni Virtema:** Writing – review & editing, Writing – original draft, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Max Sandstrom reports financial support was provided by Research Council of Finland grant 322795. Juha Kontinen reports financial support was provided by Research Council of Finland grant 345634. Jonni Virtema reports financial support was provided by Research Council of Finland grant 345634. Jonni Virtema reports financial support was provided by German Research Foundation grant VI 1045/1-1. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] J. Baumeister, N. Coenen, B. Bonakdarpour, B. Finkbeiner, C. Sánchez, A temporal logic for asynchronous hyperproperties, in: CAV (1), Springer, 2021, pp. 694–717.
[2] M.R. Clarkson, B. Finkbeiner, M. Koleini, K.K. Micinski, M.N. Rabe, C. Sánchez, Temporal logics for hyperproperties, in: POST 2014, 2014, pp. 265–284.
[3] M.R. Clarkson, F.B. Schneider, Hyperproperties, J. Comput. Secur. 18 (2010) 1157–1210.
[4] N. Coenen, B. Finkbeiner, C. Hahn, J. Hofmann, The hierarchy of hyperlogics, in: LICS 2019, IEEE, 2019, pp. 1–13.
[5] B. Finkbeiner, C. Hahn, P. Lukert, M. Stenger, L. Tentrup, Synthesis from hyperproperties, Acta Inform. 57 (2020) 137–163, https://doi.org/10.1007/s00236-019-00358-2.
[6] P. Galliani, Inclusion and exclusion dependencies in team semantics: on some logics of imperfect information, Ann. Pure Appl. Log. 163 (2012) 68–84.
[7] J.O. Gutsfeld, A. Meier, C. Ohrem, J. Virtema, Temporal team semantics revisited, in: C. Baier, D. Fisman (Eds.), LICS '22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel, August 2 - 5, 2022, ACM, 2022, pp. 44:1–44:13.
[8] J.O. Gutsfeld, M. Müller-Olm, C. Ohrem, Automata and fixpoints for asynchronous hyperproperties, Proc. ACM Program. Lang. 5 (2021) 1–29, https://doi.org/10.1145/3434319.
[9] M. Hannula, J. Kontinen, J. Virtema, H. Vollmer, Complexity of propositional logics in team semantic, ACM Trans. Comput. Log. 19 (2018) 2:1–2:14.
[10] L. Hella, A. Kuusisto, A. Meier, J. Virtema, Model checking and validity in propositional and modal inclusion logics, J. Log. Comput. 29 (2019) 605–630, https://doi.org/10.1093/logcom/exz008.
[11] J. Kontinen, J. Müller, H. Schnoor, H. Vollmer, Modal independence logic, in: R. Goré, B.P. Kooi, A. Kurucz (Eds.), Advances in Modal Logic 10, Invited and Contributed Papers from the Tenth Conference on "Advances in Modal Logic," Held in Groningen, The Netherlands, August 5-8, 2014, College Publications, 2014, pp. 353–372, http://www.aiml.net/volumes/volume10/Kontinen-Mueller-Schnoor-Vollmer.pdf.
[12] J. Kontinen, M. Sandström, On the expressive power of teamltl and first-order team logic over hyperproperties, in: WoLLIC, Springer, 2021, pp. 302–318.
[13] J. Kontinen, M. Sandström, J. Virtema, A remark on the expressivity of asynchronous teamltl and hyperltl, in: FoIKS, Springer, 2024, pp. 275–286.
[14] A. Krebs, A. Meier, J. Virtema, A team based variant of CTL, in: F. Grandi, M. Lange, A. Lomuscio (Eds.), 22nd International Symposium on Temporal Representation and Reasoning, TIME 2015, Kassel, Germany, September 23-25, 2015, IEEE Computer Society, 2015, pp. 140–149.

[15] A. Krebs, A. Meier, J. Virtema, M. Zimmermann, Team semantics for the specification and verification of hyperproperties, in: I. Potapov, P. Spirakis, J. Worrell (Eds.), MFCS 2018, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2018, pp. 10:1–10:16.
[16] M. Lück, Axiomatizations of team logics, Ann. Pure Appl. Log. 169 (2018) 928–969.
[17] M. Lück, On the complexity of team logic and its two-variable fragment, in: MFCS, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, pp. 27:1–27:22.
[18] M. Lück, On the complexity of linear temporal logic with team semantics, Theor. Comput. Sci. (2020).
[19] M. Lück, Team logic: axioms, expressiveness, complexity, Ph.D. thesis, University of Hanover, Hannover, Germany, 2020, https://www.repo.uni-hannover.de/handle/123456789/9430.
[20] J. McLean, Proving noninterference and functional correctness using traces, J. Comput. Secur. 1 (1992) 37–58, https://doi.org/10.3233/JCS-1992-1103.
[21] N. Piterman, A. Pnueli, Temporal logic and fair discrete systems, in: E.M. Clarke, T.A. Henzinger, H. Veith, R. Bloem (Eds.), Handbook of Model Checking, Springer, 2018, pp. 27–73.
[22] A. Pnueli, The temporal logic of programs, in: 18th Annual Symposium on Foundations of Computer Science, IEEE Computer Society, 1977, pp. 46–57.
[23] M.N. Rabe, A Temporal Logic Approach to Information-Flow Control, Ph.D. thesis, Saarland University, 2016.
[24] A.W. Roscoe, CSP and determinism in security modelling, in: Proceedings of the 1995 IEEE Symposium on Security and Privacy, Oakland, California, USA, May 8–10, 1995, IEEE Computer Society, 1995, pp. 114–127.
[25] A.P. Sistla, E.M. Clarke, The complexity of propositional linear temporal logics, J. ACM 32 (1985) 733–749, https://doi.org/10.1145/3828.3837.
[26] J. Väänänen, Dependence Logic, Cambridge University Press, 2007.
[27] J. Virtema, Complexity of validity for propositional dependence logics, Inf. Comput. 253 (2017) 224–236, https://doi.org/10.1016/j.ic.2016.07.008.
[28] J. Virtema, J. Hofmann, B. Finkbeiner, J. Kontinen, F. Yang, Linear-time temporal logic with team semantics: expressivity and complexity, CoRR, arXiv:2010.03311 [abs], https://arxiv.org/abs/2010.03311, arXiv:2010.03311, 2020.
[29] J. Virtema, J. Hofmann, B. Finkbeiner, J. Kontinen, F. Yang, Linear-time temporal logic with team semantics: expressivity and complexity, in: M. Bojanczyk, C. Chekuri (Eds.), 41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2021, December 15-17, 2021, Virtual Conference, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, pp. 52:1–52:17.
[30] S. Zdancewic, A.C. Myers, Observational determinism for concurrent program security, in: 16th IEEE Computer Security Foundations Workshop (CSFW-16 2003), 30 June - 2 July 2003, Pacific Grove, CA, USA, IEEE Computer Society, 2003, p. 29.