A Parametrical Similarity Saturation Based Decision Procedure for a Fragment of FTL

Regimantas Pliuškevičius

Institute of Mathematics and Informatics,
Akademijos 4, Vilnius 2600, LITHUANIA,
regis@ktl.mii.lt

Abstract. A simple saturation based decision procedure for a fragment of a first-order linear temporal logic with function symbols and temporal operators Next and Always is presented. The main part of the presented decision procedure consists of two decidable procedures which allow us to construct a fully formal inductive proof. The soundness and completeness of the proposed decision procedure are proved.

1 Introduction

First-order temporal logic (FTL, in short) is a very expressive language. Unfortunately, FTL is incomplete, in general [14]. But it becomes complete [7, 15] after adding an ω-type rule. The sequent version of this infinitary rule of inference is of the following shape:

$$\frac{\Gamma \to \Delta, A; \Gamma \to \Delta, \bigcirc A; \ldots; \Gamma \to \Delta, \bigcirc^k A; \ldots}{\Gamma \to \Delta, \Box\, A}\ (\to \Box_\omega).$$

The rule of inference $(\to \Box_\omega)$ expresses the semantics of the temporal operator Always (formula $\Box\, A$ means:"A is true now and will be true in all future states of time"). Some fragments of the first-order linear temporal logic are finitary complete and/or decidable. In these cases instead of the ω-type rule $(\to \Box_\omega)$ one can use the following finitary rule:

$$\frac{\Gamma \to \Delta, I; I \to \bigcirc I; I \to A}{\Gamma \to \Delta, \Box\, A}\ (\to \Box).$$

This rule corresponds to the induction axiom: $A \wedge \Box(A \supset \bigcirc A) \supset \Box\, A$. The formula I is called an *invariant formula* and has a close relation with invariant formulas in Hoare logic and dynamic logic.

Recently in [6] the decidability of so-called monodic fragments of FTL (without function symbols) has been proved. The aim of this paper is to

present a simple deduction-based decision procedure $PSSat$ for a non-monodic fragment of FTL with function symbols. The objects of consideration of the proposed procedure $PSSat$ are so-called parametrical similar sequents (PS-sequents, in short). The PS-sequents are of the following simple form $\Sigma, \Box\Omega \rightarrow \Box^{\circ}A$, where $\Box^{\circ} \in \{\varnothing, \Box\}$, Σ consists of atomic formulas, $\Box\Omega$ consists of the so-called kernel formulas. PS-sequents are somewhat specialization of Fisher's normal form [4]. The exact notion of PS-sequents is presented in Definition 5 (see Section 2). As in [8-13] the proposed decision procedure $PSSat$ is based on the saturation method. Usually the saturation (see, e.g., [5]) means computing the closure of a given set of sequents under a set R of rules. The presence of the existential quantifier and function symbols in kernel formulas force us to use the so-called parametrical similarity saturation.

The proposed decision procedure $PSSat$ consists of a preliminary step and the main part. In the preliminary step we try automatically to reduce a given PS-sequent to so-called semi-saturated PS-sequent.

The main part of $PSSat$ consists of two decidable procedures $Re^k(S)$ and $HRe^k(S)$. These procedures allow us to construct a fully formal inductive proof. The computation of $Re^k(S)$ provides the base case and then we can use $HRe^k(S)$ to prove the inductive step. The procedures $Re^k(S)$ and $HRe^k(S)$ effectively replace the ω-type rule $(\rightarrow \Box_{\omega})$ and the finitary invariant-type rule $(\rightarrow \Box)$. Both procedures do not contain any logical rules and instead of temporal rules contain two so-called separation rules (IS) and (GIS) which introduce some substitution operation. These separation rules allow us to construct a derivation of any PS- sequent in a form of rather simple tree. The separation rule (GIS) allows us to construct an inductive hypothesis automatically.

As in [8–13], the notions of calculus and deduction-based decision procedure are identical.

The paper is organized as follows. In Section 2, a sound and ω-complete (for PS-sequents) loop-free infinitary calculus G_{ω} is presented. In section 3, the preliminary step and the main part of $PSSat$ are described and decidability of $PSSat$ is established. In section 4, a decidable invariant calculus $PSIN$ is presented. It is stated that $G_{\omega} \vdash S \iff PSSat \vdash S \iff PSIN \vdash S$. From this fact the soundness and completeness of the calculi $PSSat$ and $PSIN$ are obtained. In section 5, conclusions, related works and future investigations are shortly discussed.

2 Loop-free infinitary calculus G_ω

An inference rule (i) of some sequent calculus is called a loop rule if the premise (premises) of the rule (i) contains the main formula (or a modification of the main formula) of the rule. Loop rules may be a reason of severe problems in the proof search and make it difficult to find decision procedures. The proposed decision procedures $PSSat$ are justified using loop-free infinitary calculus G_ω.

We assume that all the predicate symbols are flexible (i.e., change their value in time), and functions are rigid (i.e., with time-independent meanings). For simplicity we consider only two-place predicate symbols and only one-place function symbols. A term and a formula are defined in the usual way. An atomic formula is an expression of the form $P(t_1, t_2)$, where P is a predicate symbol, t_1, t_2 are terms.

In the first-order linear temporal logic, we have that $\bigcirc(A \odot B) \equiv \bigcirc A \odot \bigcirc B(\odot \in \{\supset, \wedge, \vee\})$ and $\bigcirc\sigma A \equiv \sigma\bigcirc A(\sigma \in \{\neg, \Box, \forall x, \exists x\})$. Relying on these equivalences, we can consider occurrences of the Next operator $\bigcirc$ only entering the formula $\bigcirc^k P(t_1, t_2)$ (k-time Next atomic formula $P(t_1, t_2)$). For the sake of simplicity, we "eliminate" the Next operator and the formula $\bigcirc^k P(t_1, t_2)$ is abbreviated as $P^k(t_1, t_2)$ (i.e., as an atomic formula with the index k).

The main tool for eliminating the loop rules is a transformation of some formulas and/or joining of some rules. We used this idea to construct an infinitary calculus with loop-free rules.

To construct a loop-free infinitary calculus, let us introduce PS^*-sequents. First we introduce kernel formulas.

Definition 1 (kernel formula). *A formula $\Box A$ is a kernel formula, if $\Box A = \Box\forall x_1 x_2(E(x_1, x_2) \supset \exists y P^1(\bar{f}(x_1), y))$, where $\bar{f}(x_1) = f_1(f_2 \ldots (f_n(x_1$ $)) \ldots)$, f_i $(1 \leq i \leq n)$ is a function symbol.*

Definition 2 (PS^*-sequent, induction-free PS^*-sequent, non-repeating conditions). *A sequent S is a PS^*-sequent, if $S = \Sigma, \Box\Omega \rightarrow \Box^\circ A$, where Σ consists of atomic formulas; $\Box\Omega$ consists of kernel formulas; $A = \bigvee_{i=1}^{m} \exists^\circ \bar{y}_i E_i^{k_i}(\bar{y}_i)$ $(\exists^\circ \bar{y}_i \in \{\varnothing, \exists \bar{y}_i\}$, $\bar{y}_i = y_{i_1}, y_{i_2}$; $k_i \geq 0)$; $\Box^\circ \in \{\varnothing, \Box\}$, if $\Box^\circ = \varnothing$ then S is an induction-free PS^*-sequent. Each PS^*-sequent must satisfy the following*

Non-repeating conditions:

(a) let $\Box\forall x_1 x_2(E_i(x_1, x_2) \supset \exists y P_i^1(\bar{f}(x_1), y)) \in \Box\Omega$ and $\Box\forall z_1 z_2(E_j(z_1,$ $z_2) \supset \exists u P_j^1(\bar{g}(z_1), u)) \in \Box\Omega$, then $\forall ij$ $(E_i \neq E_j$ and $P_i \neq P_j)$;

(b) if $Q(t_1,t_2) \in \Sigma$, *then* $Q(p_1,p_2) \notin \Sigma$ *for each terms* t_i, p_i *(*$i \in \{1,2\}$*).*

Remark 1. Atomic formulas from Σ correspond to so-called "start" formulas, kernel formulas correspond to "next" (or "step") formulas, and the succedent formula $\Box A$ corresponds to so-called "sometimes" formulas from Fisher's normal form [4].

The shape of PS^*-sequents allow us to construct a loop-free infinitary calculus. First we introduce the following operation.

Definition 3 (operation $(+)$**).** *Let* $\Sigma, \Box\Omega \to \Box^\circ A$ *be a* PS^**-sequent and* $E(t_1,t_2) \in \Sigma$. *Let* $\Box\forall x_1x_2(E(x_1,x_2) \supset \exists y P^1(\bar{f}(x_1,y))) \notin \Box\Omega$, *then* $(E(t_1,t_2))^+ = \varnothing$. *Let* $\Box\forall x_1x_2(E(x_1,x_2) \supset \exists y P^1(\bar{f}(x_1,y))) \in \Box\Omega$, *then* $(E(t_1,t_2))^+ = P(\bar{f}(x^*),b)$ *(where* x^* *is a new variable and the value of this variable is the term* t_1*;* b *is a new constant, called an eigen-constant of the operation* $(+)$*). A substitution* $x^* \leftarrow t_1$, *is written alongside to the result of the operation* $(+)$ *and it is used to find the similarity substitution (see Lemma 7 and Algorithm (SS)). Let* $\Sigma = E_1, \ldots, E_n$, *then* $(\Sigma)^+ = (E_1)^+, \ldots, (E_n)^+$.

Example 1. Let $\Sigma = R(a,b), P(a,c), E(e,d)$; $\Box\Omega = \Box\forall x_1x_2(E(x_1,x_2) \supset \exists y P^1(f(x_1),y))$, $\Box\forall z_1,z_2\ (P(z_1,z_2) \supset \exists u E^1(g(z_1),u))$. Then $(\Sigma)^+ = E(g(x^*_{11}),b_1),\ P(f(x^*_{21}),b_2)$; $\{x^*_{11} \leftarrow a,\ x^*_{21} \leftarrow e\}$, where b_1, b_2 are eigen-constants of the operation $(+)$.

Definition 4 (calculi G_ω**,** G**).** *A calculus* G_ω *is defined by the following postulates.*

Axiom:

$$\Gamma, E_i^{k_i}(\bar{f}_1(a_1), \bar{f}_2(a_2)) \to \bigvee_{j=1}^{n} \exists z_{j1}z_{j2} E_j^{k_j}(\bar{g}_1(z_{j1}), \bar{g}_2(z_{j2})),$$

where $\exists j$ *such that* $E_j^{k_j} = E_i^{k_i}$ *and the term* $\bar{f}_k(a_k)$ *(*$k \in \{1,2\}$*) is an instance of the term* $\bar{g}_k(z_{ik})$ *(for example,* $f(g(a))$ *is the instance of the term* $f(z)$*).*

Rules:

$$\frac{\Gamma \to A; \Gamma \to A^1; \ldots; \Gamma \to A^k; \ldots}{\Gamma \to \Box A}\,(\to \Box_\omega)$$

$$\frac{(\Sigma)^+, \Box\Omega \to A^{-1}}{\Sigma, \Box\Omega \to A}\,(IS),$$

where $\Sigma, \Box\Omega \to A$ *is an induction-free* PS^**-sequent which is not an axiom;* $A = \bigvee_{i=1}^{m} \exists^\circ \bar{y}_i E_i^{k_i}(\bar{y}_i)$; A^{-1} *denotes the formula which is obtained from* A,

replacing the formula $E_i^{k_i}(\bar{y}_i)$ *by* $E_i^{k_i-1}(\bar{y}_i)$*, moreover, if* $k_i - 1 < 0$ *then the* i*-th disjunctive component is omitted.*

A calculus G *is obtained from* G_ω *by dropping the rule* $(\rightarrow \Box_\omega)$*.*

Remark 2. The axiom replaces the loop-type rule $(\rightarrow \exists)$. The rule (IS) incorporates the logical rules $(\supset\rightarrow)$, $(\exists\ \rightarrow)$, the traditional loop-type rules $(\forall\ \rightarrow)$, $(\Box\ \rightarrow)$ and the following rule $\dfrac{\Gamma \rightarrow A}{\Sigma, \Gamma^1 \rightarrow A^1}\,(+1)$, which is admissible in G_ω and which corresponds to the traditional rule for the Next operation $\dfrac{\Gamma \rightarrow A}{\Sigma, \bigcirc\Gamma \rightarrow \bigcirc A}$.

Lemma 1. *The calculus* G *is decidable.*

Proof. Follows from invertibility in G of the rule (IS) and decidability of the axiom.

Theorem 1 (soundness and ω**-completeness of** G_ω**).** *Let* S *be a* PS^**-sequent then* $\forall M \Vdash S \iff G_\omega \vdash S$*.*

Proof. Analogously as in [13].

This ends the description of the loop-free infinitary calculus G_ω .

3 Description of decidable saturation based decision procedure *PSSat*

In this section, we describe a preliminary step and the main part of the proposed decision procedure $PSSat$. The procedure $PSSat$ will be applied to so-called periodic PS^*-sequents (PS-sequents, in short). This periodic condition essentially simplifies saturation procedures $Re^k(S)$ and $HRe^k(S)$ (see below) of $PSSat$.

Definition 5 (PS**-sequent).** *A* PS^**-sequent* $S = \Sigma, \Box\Omega \rightarrow \Box^\circ A$ *is a* PS*-sequent, if the kernel formulas from* $\Box\Omega$ *satisfy the following*

Periodic condition:

$$\begin{aligned}
\Box\Omega = \ & \Box\forall x_{11}x_{12}(E_1(x_{11}, x_{12}) \supset \exists y_1 E_2^1(\bar{f}_1(x_{11}), y_1)),\\
& \Box\forall x_{21}x_{22}(E_2(x_{21}, x_{22}) \supset \exists y_2 E_3^1(\bar{f}_2(x_{21}), y_2)),\\
& \qquad\qquad \cdots\cdots\\
& \Box\forall x_{n1}x_{n2}(E_n(x_{n1}, x_{n2}) \supset \exists y_n E_{n+1}^1(\bar{f}_n(x_{n1}), y_n))\\
& \textit{and} \quad E_1 = E_{n+1}.
\end{aligned}$$

Let us define the generalized integrated separation rule (GIS) which is applied to any non-induction-free PS-sequent.

Definition 6 (generalized integrated separation rule: (GIS), successful applications of (GIS)). *Let $S = \Sigma, \Box\Omega \to \Box B$ be a PS-sequent and $(\Sigma)^+$ means the same as in the definition of (IS). Then the generalized integrated separation rule (GIS) is as follows:*

$$\frac{\Sigma, \Box\Omega \to B; (\Sigma)^+, \Box\Omega \to \Box B}{\Sigma, \Box\Omega \to \Box B}\,(GIS).$$

If the left premise of (GIS), i.e., the sequent $S_1 = \Sigma, \Box\Omega \to B$ is such that $G \vdash S_1$, we say that a bottom-up application of (GIS) is successful.

The notation $(GIS)(S) = S_2$ means that after successful bottom-up applications of (GIS), we get a sequent S_2 as the right premise of (GIS).

Remark 3. The rule (GIS) incorporates the rules (IS), $(+1)$ and the following rule

$$\frac{\Gamma \to A; \Gamma \to \Box A^1}{\Gamma \to \Box A}(\to \Box^1).$$

Using the fact that $G_\omega \vdash \Box A \equiv A \wedge \Box A^1$ and from the admissibility of (cut) in G_ω we get that the rule $(\to \Box^1)$ is admissible and invertible in G_ω.

Lemma 2. *The rule (GIS) is admissible and invertible in the calculus G_ω.*

Proof. Using admissibility and invertibility of $(\to \Box^1)$, admissibility of $(+1)$ and applying analogous reasonings as in [13].

Now we present a preliminary step of the calculus $PSSat$. The aim of this preliminary step is to obtain a so-called semi-saturated PS-sequent. To define the preliminary step, let us introduce the following notions.

Definition 7 (rank of PS-sequent: $r(S)$). *Let $S = \Sigma, \Box\Omega \to \Box B$ be a PS-sequent and $E = Q(\bar{f}(a), c)$ be any atomic formula from Σ. Let us define the rank of E (in symbols: $r(E)$): $r(E) = 0$, if $\Box\forall x_1 x_2(P(x_1, x_2) \supset \exists y Q^1(\bar{f}(x_1), y)) \in \Box\Omega$, otherwise, $r(E) = 1$. Let $S = E_1, \ldots, E_n, \Box\Omega \to \Box B$, then $r(S) = \sum_{i=1}^{n} r(E_i)$.*

Example 2. Let $\Box\Omega = \Box\forall x_1 x_2(P(x_1, x_2) \supset \exists y Q^1(f(x_1), y))$. Then if $E = Q(f(a), c)$, then $r(E) = 0$, and if $E = Q(a, c)$, then $r(E) = 1$ or if $E = Q(f(a), g(c))$, then $r(E) = 1$.

Definition 8. *(semi-saturated PS-sequent). Let S be a PS-sequent, then S is a semi-saturated PS-sequent if $r(S) = 0$.*

Definition 9 (preliminary step of $PSSat$, successful preliminary step). *Let $r(S) > 0$, then the preliminary step of PSSat consists of a bottom-up application of (GIS), by means of which the semi-saturated PS-sequent is obtained. If $r(S) = 0$, then the preliminary step is missing. The preliminary step is successful, if the bottom-up application of (GIS) is successful.*

Lemma 3. *The problem of constructing a semi-saturated PS-sequent S^* from an arbitrary PS-sequent S is decidable.*

Proof. Follows from the decidability of the calculus G.

Example 3. Let $\Sigma, \Box\Omega$ be the same as in Example 1, $A = \exists u_1 v_1 E(u_1, v_1)$. Then $S = \Sigma, \Box\Omega \to \Box A$ and $r(S) = 3$. Bottom-up applying (GIS) to S we get that preliminary step is successful and the semi-saturated PS-sequent is as follows $S^* = (\Sigma)^+, \Box\Omega \to \Box A$, where $(\Sigma)^+$ is the same as in Example 1.

Now we are going to define the basic procedure of $PSSat$. The procedure will be called as k-th resolvent of the semi-saturated PS-sequent S (in short: $Re^k(S)$).

First we define a halting parameter for $Re^k(S)$, namely, a similarity index.

Definition 10. *(similarity index: $p(S)$)). Let $S = \Sigma, \Box\Omega \to \Box A$ be a PS-sequent and $|\Box\Omega|$ be the number of kernel formulas in $\Box\Omega$, then $p(S) = |\Box\Omega|$.*

Now we define $Re^k(S)$, the parametrical part of $Re^k(S)$ and existential closure of the parametrical part of $Re^k(S)$.

Definition 11. *(k-th resolvent: $Re^k(S)$, favourable procedure $Re^k(S)$, parametrical part of $Re^k(S)$), existential closure of the of $Re^k(S)$). Let S be a semi-saturated PS-sequent. Then $Re^0(S) = S$. Let $Re^k(S) = S_k = \Sigma, \Box\Omega \to \Box A$, then $Re^{k+1}(S)$ is defined in the following way:*

1. Let us bottom-up apply the rule (GIS) to S_k, and S_{k1}, S_{k2} be the left and right premises of the application of (GIS).

2. If $G \not\vdash S_{k1}$, then $Re^{k+1}(S) = \bot$ (false) and the calculation of $Re^{k+1}(S)$ is stopped.

3. Let $G \vdash S_{k1}$ (it means that the bottom-up application of (GIS) is successful). Then $Re^{k+1}(S) = S_{k2} = (\Sigma)^+, \Box\Omega \to \Box A$; $(\Sigma)^+$ is called a parametrical part of $Re^{k+1}(S)$. Let $(\Sigma)^+ = E_1(\bar{f}_1(a_1), b_1), \ldots, E_m(\bar{f}_m(a_m), b_m)$ ($b_1, \ldots, b_m$ are the eigen-constants of the operation $(+)$) be a parametrical part of $Re^{k+1}(S)$. Then $\exists x_1 x_2\, E_1(\bar{f}_1(x_1), x_2), \ldots, \exists x_1 x_2\, E_m(\bar{f}_m(x_1), x_2)$ is an existential closure of the parametrical part of $Re^{k+1}(S)$.

4. If $Re^{k+1}(S) = S_{k2}$ and $k + 1 = |\Box\Omega|$, then the calculation of $Re^{k+1}(S)$ is finished.

The notation $Re^k(S) \neq \perp$ $(k \leq p(S))$, where $p(S)$ is the similarity index of S, means that all the bottom-up applications of (GIS) in the calculation of $Re^k(S)$ are successful. In this case, we say that $Re^k(S)$ is favourable for the semi-saturated PS-sequent S.

Lemma 4. *The problem of calculation of $Re^k(S)$ is decidable, i.e. for any semi-saturated PS-sequent S the computation of $Re^k(S)$ always terminates.*

Proof. Follows from the decidability of G and the definition of $Re^k(S)$.

Lemma 5 (composition of $Re^k(S)$). *Let $Re^n(S) = S_n, Re^m(S_n) = S^*$ and $Re^n(S) \neq \perp$ $(n \leq p(S))$, $Re^m(S_n) \neq \perp$ $(m \leq p(S))$. Then $Re^l(S) = S^*$, where $l = n + m$.*

Proof. By induction on l.

Lemma 6 (decomposition of $Re^k(S)$). *Let $Re^{n+m}(S) = S^*$ and $Re^{n+m}(S) \neq \perp$ $(k \leq p(S))$, then for each n and m there exists a sequent S_n such that $Re^n(S) = S_n$ and $Re^m(S_n) = S^*$.*

Proof. By induction on $n + m$.

Now we can state the main property (so-called basic loop property) of the procedure $Re^k(S)$. First we define some notions.

Definition 12 (parametrically identical PS-sequents). *Two formulas are parametrically identical (in symbols: $A \approx A^*$) if A, A^* differ only by the corresponding occurrences of eigen-constants of the operation $(+)$. Two PS-sequents S_1 and S_2 are parametrically identical (in symbols: $S_1 \approx S_2$) if S_1, S_2 consist of parametrically identical formulas.*

Definition 13 (similarity substitution σ). *Let S be a semi-saturated PS-sequent, then a substitution σ is called similarity substitution if $S\sigma \approx Re^k(S)$, where $k = p(S)$.*

Lemma 7 (basic loop property). *Let S be a semi-saturated PS-sequent, $Re^l(S) \neq \bot$ $(l \leq p(S))$ and $Re^k(S) = S^*$, where $k = p(S)$. Then $S\sigma \approx S^*$.*

Proof. Using decomposition and composition properties of $Re^l(S)$, and applying induction on $p(S)$.

From the proof of Lemma 7 a way for constructing the similarity substitution σ can be extracted. Below an algorithm for generating the similarity substitution σ is presented.

Algorithm (SS) (algorithm for constructing the similarity substitution). Let $S = Q_1(\bar{g}_1(x_1^*), b_{11}), \ldots, Q_r(\bar{g}_r(x_r^*), b_{1r}), \Box\Omega \to \Box A$ be a semi-saturated PS-sequent and $p(S) = k$. From Lemma 7 it follows that $Re^k(S) = S_k = Q_1(\bar{g}_1(x_{k1}), b_{k1},), \ldots, Q_r(\bar{g}_r(x_{kr}), b_{kr}), \Box\Omega \to \Box A$. The algorithm (SS) consists of three steps.

(1) Let σ_1 be a sequence of substitutions obtained during the construction of $Re^k(S) = S_k$.

(2) In σ_1 replace all the intermediate variables between x_i^* and x_{ki} $(1 \leq i \leq r)$ by the corresponding values. Continue these transformations until a sequence σ^* containing r substitutions of the shape $x_{ki} \leftarrow \bar{g}_{ki}(\ldots \bar{g}_i(x_i^*)\ldots)$ $(1 \leq i \leq r$ and $\bar{g}_{ki}$ consists of the function symbols from $\bar{g}_i)$ is obtained.

(3) In σ^* replace the variables x_{ki} by the variables x_i^* $(1 \leq i \leq r)$. Then, the desired similarity substitution σ has the shape $\sigma := \{x_1^* \leftarrow \bar{g}_{k1}(\ldots \bar{g}_1(x_1^*)\ldots), \ldots, x_r^* \leftarrow \bar{g}_{kr}(\ldots \bar{g}_r(x_r^*)\ldots)\}$.

Lemma 8. *The algorithm (SS) is correct, it terminates and calculates the unique similarity substitution.*

Proof. Follows from description of the Algorithm (SS).

Example 4. Let S be the following semi-saturated PS-sequent: $S = Q(f(x_1^*, b_1)), R(g(x_2^*, b_2)),\ E(h(x_3^*, b_3)),\ \Box\Omega \to \Box A$, where $\Box\Omega = \Box\forall x_1 z_1 (E(x_1, z_1) \supset \exists y_1 Q^1(f(x_1), y_1)),\ \Box\forall x_2 z_2 (Q(x_2, z_2) \supset \exists y_2 R^1(g(x_2), y_2)),\ \Box\forall x_3 z_3 (R(x_3, z_3) \supset \exists y_3 E^1(h(x_3), y_3))$; $A = \exists u_1 v_1 E(u_1, v_1)$. In order to verify that S satisfies the basic loop property let us construct $Re^k(S)$. Since $|\Box\Omega| = 3$, $k = 3$. It is easy to verify that $Re^i(S) \neq \bot$ $(i \in \{1, 2, 3\})$. Therefore we indicate only a temporal premise of the rule (GIS) and substitutions generated by means of the operation $(+)$:

$$S_3 = Q(f(x_{13}), b_{13}),\ R(g(x_{23}), b_{23}),\ E(h(x_{33}), b_{33}),\ S_0;$$
$$\{x_{13} \leftarrow h(x_{12}),\ x_{23} \leftarrow f(x_{22}),\ x_{33} \leftarrow g(x_{32})\}$$

$$\frac{\begin{array}{l} E(h(x_{12}), b_{12}),\ Q(f(x_{22}), b_{22}),\ R(g(x_{32}), b_{32}),\ S_0; \\ \qquad\qquad \{x_{12} \leftarrow g(x_{11}),\ x_{22} \leftarrow h(x_{21}),\ x_{32} \leftarrow f(x_{31})\} \end{array}}{\begin{array}{l} R(g(x_{11}), b_{11}),\ E(h(x_{21}), b_{21}),\ Q(f(x_{31}), b_{31}),\ S_0; \\ \qquad\qquad \{x_{11} \leftarrow f(x_1^*),\ x_{21} \leftarrow g(x_2^*),\ x_{31} \leftarrow h(x_3^*)\} \end{array}}\ ,$$
$$S = Q(f(x_1^*), b_1),\ R(g(x_2^*), b_2),\ E(h(x_3^*), b_3),\ S_0$$

where $S_0 = \Box\Omega \rightarrow \Box A$.

Let us now construct a similarity substitution σ. Let σ_1 be the sequence of the substitutions obtained during the construction of $Re^3(S) = S_3$, i.e., $\sigma_1 = \{x_{11} \leftarrow f(x_1^*),\ x_{21} \leftarrow g(x_2^*),\ x_{31} \leftarrow h(x_3^*),\ x_{12} \leftarrow g(x_{11}),\ x_{22} \leftarrow h(x_{21}),\ x_{32} \leftarrow f(x_{31}),\ x_{13} \leftarrow h(x_{12}),\ x_{23} \leftarrow f(x_{22}),\ x_{33} \leftarrow g(x_{32})\}$. Let us eliminate the intermediate variables x_{i1} and x_{i2} $(1 \leq i \leq 3)$. First, let us eliminate the variables x_{i2} $(1 \leq i \leq 3)$, i.e., replace the variables x_{i2} by the corresponding values of these variables. So, instead of the sequence σ_1 we get $\sigma_2 = \{x_{11} \leftarrow f(x_1^*),\ x_{21} \leftarrow g(x_2^*),\ x_{31} \leftarrow h(x_3^*),\ x_{13} \leftarrow h(g(x_{11})),\ x_{23} \leftarrow f(h(x_{21})),\ x_{33} \leftarrow g(f(x_{31}))\}$. In the same manner, let us eliminate the variables x_{i1} $(1 \leq i \leq 3)$. So, instead of the sequence σ_2 we get $\sigma_3 = \{x_{13} \leftarrow h(g(f(x_1^*))),\ x_{23} \leftarrow f(h(g(x_2^*))),\ x_{33} \leftarrow g(f(h(x_3^*)))\}$. Now, by adding $x_i^* = x_{i3}$ $(1 \leq i \leq 3)$ to σ_3 we get the desired similarity substitution $\sigma = \{x_1^* \leftarrow h(g(f(x_1^*))),\ x_2^* \leftarrow f(h(g(x_2^*))),\ x_3^* \leftarrow g(f(h(x_3^*)))\}$. So, $S\sigma \approx S_3$ and S satisfies the basic loop property.

Lemma 9. *Let S be a semi-saturated PS-sequent. Then the problem of testing that S satisfies the basic loop property is decidable.*

Proof. Follows from decidability of G, the Algorithm (SS) and definition of $Re^k(S)$.

Remark 4. Let S satisfy the basic loop property. Then the PS- sequent $S\sigma$ is an inductive hypothesis which is generated using the procedure $Re^k(S)$ automatically.

Now we define hypothetical k-th resolvent ($HRe^k(S)$, in symbols). The halting parameter for $HRe^k(S)$ is the same as for $Re^k(S)$, namely, $k = p(S)$, i.e., the similarity index of S. First let us construct the substitution σ^n in the following way. Let $\sigma = \{x_1^* \leftarrow \bar{f}_1(x_1^*), \ldots, x_m^* \leftarrow \bar{f}_m(x_m^*)\}$, then $\sigma^n = \{x_1^* \leftarrow \bar{f}_1^n(x_1^*), \ldots, x_m^* \leftarrow \bar{f}_m^n(x_m^*)\}$, where $\bar{f}_i(x_i^*) = \varnothing$; $\bar{f}_i^n(x_i^*) = \bar{f}_i(\bar{f}_i^{n-1}(x^*))$, therefore $\sigma^0 = \varnothing$ and $\sigma^1 = \sigma$. For example, if $\sigma = x^* \leftarrow f(g(x^*))$, then $\sigma^2 = x^* \leftarrow f(g(f(g(x^*))))$.

Definition 14 (hypothetical k-th resolvent: $HRe^k(S)$). *Let S be a semi-saturated PS-sequent, σ be the similarity substitution, and m an arbitrary natural number, then $HRe^0(S) = S\sigma^m$. Let $HRe^k(S) = S_k$, then $HRe^{k+1}(S)$ is defined in the same way as $Re^{k+1}(S)$.*

Lemma 10 (hypothetical loop property). *Let S be a semi-saturated PS-sequent and $HRe^k(S) \neq \bot$ $(k \leq p(S))$. Let m be an arbitrary natural number, σ be the similarity substitution, $HRe^\circ(S) = S\sigma^m = S^*$ and $HRe^l(S) = S^{**}$, where $l = p(S)$. Then $S^*\sigma \approx S^{**}$.*

Lemma 11. *Let S be a semi-saturated PS-sequent. Then the problem of testing that S satisfies the hypothetical loop property is decidable.*

Proof. Follows from decidability of G, the Algorithm (SS) and definition of $HRe^k(S)$.

Definition 15 (calculus $PSSat$, PS-sequent derivable in $PSSat$). *A calculus $PSSat$ consists of preliminary step and procedures $Re^k(S)$, and $HRe^k(S)$. A PS-sequent S is derivable in $PSSat$ (in symbols, $PSSat \vdash S$) if (1) $GIS(S) = S^*$, where $r(S^*) = 0$; (2) S^* satisfies the basic and hypothetical loop properties; otherwise $PSSat \nvdash S$.*

Theorem 2. *The calculus $PSSat$ is decidable for the class of PS-sequents.*

Proof. Follows from Lemmas 3, 9, and 11.

Example 5. (a) Let S be the same semi-saturated PS-sequent as in Example 4. In Example 4 the similarity substitution σ was constructed and we get that S satisfies the basic loop property, i.e., $S\sigma \approx Re^3(S)$. Assume that $HRe^0(S) = S\sigma^m$. Analogously as in Example 4 we get that $HRe^3(S) \approx S\sigma^{m+1}$. Therefore S satisfies the hypothetical loop property. Hence, using Example 4, we get $PSSat \vdash S$.

(b) Let $S = E(a,c), \Box\Omega \rightarrow \Box A$, where $\Box\Omega = \forall xz(E(x,z) \supset \exists y E^1(f(x),y))$; $A = \exists u_1 E(a,u_1) \vee \exists u_2 E(f(a),u_2)$. It is easy to verify that PS-sequent S can be reduced (using the preliminary step of $PSSat$) to the semi-saturated PS-sequent $S^* = E(f(x^*),b_1), \Box\Omega \rightarrow \Box A$ and $x^* \leftarrow a$. Let us verify (using the procedure $Re^k(S^*)$) that S^* satisfies the basic loop property. Since $|\Box\Omega| = 1$, $k = 1$. It is easy to verify that the bottom-up application of (GIS) in the calculation of $Re^1(S^*)$ is successful. Therefore the construction of $Re^1(S^*)$ is as follows:

$$\frac{Re^1(S^*) = E(f(x_1),b_2), \Box\Omega \rightarrow \Box A \quad \{x_1 \leftarrow f(x^*)\}}{S^* = E(f(x^*)),\ b_1),\ \Box\Omega \rightarrow \Box A}$$

Therefore the similarity substitution $\sigma = \{x^* \leftarrow f(x^*)\}$, $S^*\sigma \approx Re^1(S^*)$ and S^* satisfies the basic loop property. Let us try to calculate $HRe^1(S^*)$. Let $HRe^0(S^*) = S^*\sigma^m$. Since $G \nvdash E(f^{m+1}(x_1), b_2), \Box\Omega \to A$, $HRe^1(S) = \bot$. Therefore $PSSat \nvdash S$.

4 Foundation of the calculus *PSSat*

In order to justify the decidable calculus $PSSat$, we introduce a so-called invariant calculus $PSIN$. First we introduce some simple calculi.

Definition 16 (calculi *Log*, G^+ and G^{++}). *A calculus Log is obtained from the calculi G replacing the rule (IS) by the traditional invertible logical rules $(\to \wedge)$, $(\to \vee)$.*

Calculus G^+ are obtained from the calculi G by adding

1) the axiom $\Gamma, \Box A \to \Delta, \Box A^1$;

2) the traditional invertible logical rules $(\exists \to)$, $(\wedge \to)$, $(\vee \to)$, $(\to \wedge)$, $(\to \vee)$.

Calculus G^{++} are obtained from the calculus G^+ by dropping the axiom $\Gamma, \Box A \to \Delta, \Box A^1$.

Lemma 12. *The calculi $J \in \{Log,\ G^+, G^{++}\}$ are decidable.*

Proof. Follows from decidability of G and invertibility of logical rules $(\exists \to)$, $(\wedge \to)$, $(\vee \to)$, $(\to \wedge)$, $(\to \vee)$.

Definition 17 (invariant calculus $PSIN$). *An invariant calculus $PSIN$ is obtained from the calculus G^+ by adding the following rule:*

$$\frac{\Sigma, \Box\Omega \to I;\ I \to I^1;\ I \to A}{\Sigma, \Box\Omega \to \Box A}\ (\to \Box)$$

The rule $(\to \Box)$ satisfies the following conditions:

(1) the conclusion of $(\to \Box)$, i.e., the PS-sequent $\Sigma, \Box\Omega \to \Box A$ satisfies the basic loop property;

(2) $I = \bigvee_{i=0}^{r-1}(\exists\Sigma_i)^\wedge \wedge (\Box\Omega)^\wedge$; $\exists\Sigma_i$ is the existential closure of the parametrical parts of $Re^i(S)$; Σ_i is the parametrical part of $Re^k(S)$, $k \in \{1, \ldots, r\}$, $r = |\Box\Omega|$, i.e., the number of kernel formulas in $\Box\Omega$); $\Gamma^\wedge$ is the conjunction of formulas from Γ;

(3) the left premise of $(\to \Box)$, i.e., the sequent $S_1 = \Sigma, \Box\Omega \to I$, is such that $Log \vdash S_1$;

(4) the middle premise of $(\to \Box)$, i.e., the sequent $S_2 = I \to I^1$, is such that $G^+ \vdash S_2$;

(5) the right premise of $(\to \Box)$*, i.e., the sequent* $S_3 = I \to A$*, is such that* $G^{++} \vdash S_3$*.*

Lemma 13. *The problem of finding the invariant formula* I *in the rule* $(\to \Box)$ *is decidable.*

Proof. Follows from Lemmas 4 and 9.

Lemma 14. *The calculus* $PSIN$ *is decidable for the class of* PS*-sequents.*

Proof. Follows from invertibility of the rule $(\to \Box)$ and Lemmas 12, 13.

Example 6. (a) Let S be the same semi-saturated PS-sequent as in Example 4. Using the Example 4 we get that the invariant formula I is of the following form

$$\begin{aligned} I = &(\exists x_1 x_2 Q(f(x_1), x_2) \wedge \exists y_1 y_2 R(g(y_1), y_2)) \wedge \exists z_1 z_2 E(h(z_1), z_2)) \\ &\vee (\exists u_1 u_2 R(g(u_1), u_2) \wedge \exists v_1 v_2 E(h(v_1), v_2)) \wedge \exists w_1 w_2 Q(f(w_1), w_2)) \\ &\vee (\exists x_3 x_4 E(h(x_3), x_4) \wedge \exists y_3 y_4 Q(f(y_3), y_4)) \wedge \exists z_3 z_4 R(g(z_3), z_4)) \wedge \Box\Omega. \end{aligned}$$

It is easy to verify that

$$Log \vdash Q(f(x_1^*), b_1), R(g(x_2^*), b_2)), E(h(x_3^*), b_3)), \Box\Omega \to I \quad (1)$$
$$G^+ \vdash I \to I^1 \quad (2)$$
$$G^{++} \vdash I \to A \quad (3)$$

Applying $(\to \Box)$ to (1), (2), (3) we get $PSIN \vdash S$.

(b) Let S^* be the semi-saturated PS-sequent obtained in Example 5(b). Using Example 5(b) we get that the invariant formula I is of the following form: $I = \exists z_1 z_2 E(f(z_1), z_2) \wedge \Box\Omega$. It is easy to verify that

$$Log \vdash E(f(x^*), b_1), \Box\Omega \to I \quad (1)$$
$$G^+ \vdash I \to I^1 \quad (2)$$

But $G^{++} \nvdash I \to A$. Therefore $PSIN \nvdash S^*$.

Theorem 3. *Let* S *be a* PS*-sequent. Then* $PSSat \vdash S \iff PSIN \vdash S \iff G_\omega \vdash S$*.*

Proof. Analogously as in [8, 12].

Theorem 4 (soundness and completeness of calculi $PSSat$ **and** $PSIN$**).** *Let* S *be a* PS*-sequent. Then* $\forall M \models S$ *iff* $I \vdash S$ *where* $I \in \{PSSat, PSIN\}$*.*

Proof. Follows from Theorems 1, 3.

5 Conclusions, related works and future investigations

In this paper we present the deduction-based decision procedure $PSSat$ for PS-sequents. The procedure is based on parametrical similarity saturation. The main feature of the proposed procedure is the automatic verification of the loop properties (see Lemmas 7, 10).

The main part of $PSSat$ consists of two decidable procedures $Re^k(S)$ and $HRe^k(S)$. These procedures allows us to construct a fully formal inductive proof. The computation of $Re^k(S)$ provides the base case and then we can use $HRe^k(S)$ to prove the inductive step. The procedure $Re^k(S)$ allows us to construct an inductive hypothesis automatically.

The construction of automated reasoning procedures for FTL becomes topical for various applications of computer science and artificial intelligence. Unfortunately, investigations on deductive reasoning procedures for FTL are not sufficiently developed. Let us indicate some works.

- Degtyarev, Fisher [3] have presented a resolution-like procedure for some fragments of FTL. This resolution procedure is different from our saturation-based procedure.
- An interesting deductive temporal procedure, called STeP system, has been developed at the Stanford University [2]. The STeP system uses some interactive tools.
- A resolution based proof procedure of Abadi and Manna [1] is not applicable for automated reasoning in the FTL, because it demands an unrestricted cut rule. The same situation is observed in Gentzen and Hilbert style systems of Szalas [16].
- The project (guided by Prof. M. Fisher) "Analysis and mechanization of decidable first-order temporal logic" in Liverpool University (UK) jointly with London Imperial College is worked out.

The presented decision procedure can be generalized for more complex fragments of FTL (e.g. including branching kernel formulas without non-repeating and non-periodic conditions) which will be considered in the next papers.

References

[1] M. Abadi, Z. Manna. Nonclausal deduction in first-order temporal logic. *ACM Journal*, **37**(2), 279–317, 1990.

[2] N.S. Bjorner, Z. Manna, H.P. Sipma, and T.E. Uribe. Deductive verification of real-time systems using STeP. *Lecture Notes in Computer Science*, **1231**, 22–43, 1997.

[3] A. Degtyarev, M. Fisher. Towards first-order temporal resolution. *Lecture Notes in Computer Science*, **2174**, 18–32, 2001.

[4] M. Fisher. A normal form for temporal logics and its applications in theorem proving and execution. *Journal of Logic and Computation*, **7**(4), 429–456, 1997.

[5] H. Ganzinger. Saturation-based theorem proving: past successes and future potential. *Proceedings of 13th Intern. Conference on Automated Deduction*, New Jersey, USA, 1996.

[6] I. Hodkinson, F. Wolter, and M. Zakharyaschev. Decidable fragments of first-order temporal logics. *Annals of Pure and Applied Logic*, **106**, 85–134, 2000.

[7] H. Kawai. Sequential calculus for a first-order infinitary temporal logic. *Zeitchr. für Math. Logic and Grundlagen der Math.*, **33**, 423–432, 1987.

[8] R. Pliuškevičius. The saturated tableaux for linear miniscoped Horn-like temporal logic. *Journal of Automated Reasoning*, **13**, 51–67, 1994.

[9] R. Pliuškevičius. A deductive decision procedure for a restricted FTL. *Abstracts of Seventh Workshop on Automated Reasoning*, London, 2000.

[10] R. Pliuškevičius. On ω-decidable and decidable deductive procedures for a restricted FTL with Unless. *Proc. of International Workshop on First-order Theorem Proving*, St. Andrews, UK, 194–205, 2000.

[11] R. Pliuškevičius. Deduction-based decision procedure for a clausal miniscoped fragment of FTL. *Lecture Notes in Artificial Intelligence*, **2083**, 107–120, 2001.

[12] R. Pliuškevičius. On the completeness and decidability of the Horn-like fragment of the first-order linear temporal logic. *Lithuanian Math. J.*, **41**(4), 373-383, 2001.

[13] R. Pliuškevičius. Effective replaceability of the omega-rule for restricted sequents of the first-order linear temporal logic. *Lithuanian Math. J.*, **41**(3), 266-281, 2001.

[14] A. Szalas. Concerning the semantic consequence relation on first-order temporal logic. *Theoretical Computer Science*, **47**, 329–334, 1986.

[15] A. Szalas. A complete axiomatic characterization of first-order temporal logic of linear time. *Theoretical Computer Science*, **54**, 199–214, 1987.

[16] A. Szalas. Temporal logic of programs: a standard approach. In: *Time and Logic: A computational approach*, A. Szalas, A. Bolc (Eds.), UCL Press LTD, London, 1–50, 1995.