Stratified Guarded First-Order Transition Systems

Christan Müller and Helmut Seidl(✉)

TU München, Boltzmannstraße 3, Garching, Germany
seidl@in.tum.de

Abstract. First-order transition systems are a convenient formalism to specify parametric systems such as multi-agent workflows or distributed algorithms. In general, any nontrivial question about such systems is undecidable. Here, we present three subclasses of first-order transition systems where every universal invariant can effectively be decided via fixpoint iteration. These subclasses are defined in terms of syntactical restrictions: negation, stratification and guardedness. While guardedness represents a particular pattern how input predicates control existential quantifiers, stratification limits the information flow between predicates. Guardedness implies that the weakest precondition for every universal invariant is again universal, while the remaining sufficient criteria enforce that either the number of first-order variables, or the number of required instances of input predicates remains bounded, or the number of occurring negated literals decreases in every iteration. We argue for each of these three cases that termination of the fixpoint iteration can be guaranteed.

Keywords: First-order transition systems · Universal invariants · Second-order quantifier elimination · Stratification · Decidability

1 Introduction

FO transition systems (FO for First-order) are a convenient tool for specifying systems where the number of agents is not known in advance. This is very useful for modeling systems like network protocols [22] or web-based workflows like conference management, banking or commerce platforms. Consider, e.g., the specification from Fig. 1 modeling parts of the review process of a conference management system as a FO transition system.

Assume that initially, all predicates with the exception of auth are false, i.e., the property $\mathcal{H}$ given by

$$\begin{aligned}\forall x_1, x_2, p, r, d.\neg\mathsf{conf}(x_1, p) \wedge \neg\mathsf{assign}(x_1, p) \wedge \\ \neg\mathsf{report}(x_1, p, r) \wedge \neg\mathsf{discuss}(x_1, x_2, p, d)\end{aligned} \tag{1}$$

holds. The predicates $A_1, \ldots, A_4$ are *input predicates* whose values either represent agents' decisions or actions from the environment. Intuitively, the transition

D. Pichardie and M. Sighireanu (Eds.): SAS 2020, LNCS 12389, pp. 113–133, 2020.
https://doi.org/10.1007/978-3-030-65474-0_6

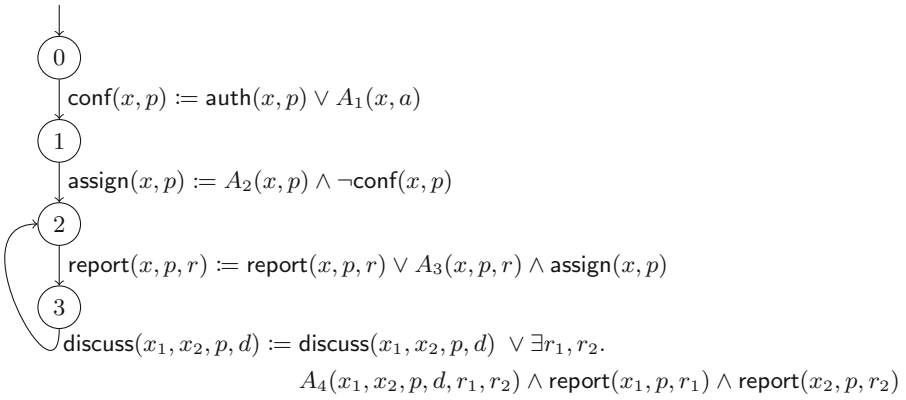

Fig. 1. A conference management system.

system works as follows: First, each PC member x possibly declares her conflict with each paper p. Then, papers p are assigned to PC members x in such a way that the conf relation is respected. Repeatedly, reports for PC members x about papers p arrive, where a subsequent discussion between PC members x_1, x_2 on some paper p is only possible if both have received a report on that paper and may update their reviews based on the discussions. Variants of this example have already been studied in [19,25].

A useful property to ensure in this example is that a discussion between x_1 and x_2 on some paper p is only possible if neither x_1 nor x_2 are authors of p:

$$\forall x_1, x_2, p, d. \neg \mathsf{discuss}(x_1, x_2, p, d) \vee \neg \mathsf{auth}(x_1, p) \wedge \neg \mathsf{auth}(x_2, p) \tag{2}$$

As FO predicate logic is undecidable, we cannot hope to find an effective algorithm for proving an invariant such as (2) for arbitrary FO transition systems. That does not exclude, though, that at least some invariants can be proven *inductive* and thus, to be valid. Also, approximation techniques may be conceived to construct *strengthenings* of given invariants which, hopefully turn out to be inductive and thus may serve as certificates for the invariants in question.

The idea of using FO predicate logic for specifying the semantics of systems has perhaps been pioneered by abstract state machines (ASMs) [6,7,14]. Recently, it has successfully been applied for the specification and verification of software-defined networks [2,20], of network protocols [23], of distributed algorithms [22]. The corresponding approach is built into the tool Ivy [18,23]. Ivy is a proof assistant for systems specified in FO logic which is carefully designed around a decidable many-sorted extension of EPR (Effectively Propositional Logic, or $\exists^*\forall^*$FO logic). In the base setting, invariants are provided manually and then checked for inductiveness by the theorem prover Z3 [8]. Some effort, though, has been invested to come up with more automatic techniques for specific settings such as threshold algorithms [4] or more general FO invariant inference [15,16]. The fundamental problem thereby is that repeated application of the

weakest precondition operator may introduce additional first-order variables, new instances of input predicates or existential quantifiers and thus result in formulas outside the decidable fragment of FO logic.

This problem also has been encountered in [10,11,19] where noninterference [13] is investigated for multi-agent workflows in the spirit of the conference management system from Fig. 1. In [19], the authors present a a symbolic verification approach where the agent capabilities as well as declassification and self-composition of the original system $\mathcal{T}$ is encoded into a FO transition system $\mathcal{T}^2$. Noninterference of the original system is thus reduced to a universal invariant of the resulting system $\mathcal{T}^2$. Further abstraction (i.e., strengthening of the encountered formulas) is applied in order to arrive at a practical algorithm which iteratively strengthens the initial invariant.

Only for rare cases, so far, decidability could be shown. In [21], Sagiv et al. show that inferring universal inductive invariants is decidable when the transition relation is expressed by formulas with unary predicates and a single binary predicate restricted by the background theory of singly-linked-lists. The same problem becomes undecidable when the binary symbol is not restricted by a background theory. In [19] on the other hand, syntactic restrictions are introduced under which termination at least of an *abstract* fixpoint iteration can be guaranteed. The abstraction thereby, consists in strengthening each occurring existential quantifier via appropriate instantiations (see also [9]). The syntactic restrictions proposed in [19] essentially amount to introducing a *stratification* on the predicates and restricting substitutions to be *stratified* and *guarded updates*. It is argued that these restrictions are not unrealistic in specifications of multi-agent systems where the computation proceeds in stages each of which accumulates information based on the results obtained in earlier stages. The example transition system from Fig. 1, e.g., is stratified: there is a mapping λ assigning a *level* $\lambda(R)$ to each predicate R so that the predicates occurring in right-hand sides which are distinct from the left-hand side have lower levels. In the example, λ could be given by

$$\{\mathsf{auth} \mapsto 0, \mathsf{conf} \mapsto 1, \mathsf{assign} \mapsto 2, \mathsf{report} \mapsto 3, \mathsf{discuss} \mapsto 4\}$$

Intuitively, stratification limits dependencies between predicates to be acyclic. Examples of *stratified guarded updates* on the other hand, are the two statements in the loop body of Fig. 1. *Guarded updates* only allow to extend predicates where the extensions constrain the use of existential quantifiers to the format $\varphi \vee \exists \bar{z}. A\bar{y}\bar{z} \wedge \psi$ for some input predicate A and quantifier-free subformulas φ, ψ.

The loop of the example thus satisfies the requirements of [19], implying that an *abstract* fixpoint iteration is guaranteed to terminate for every universal invariant. Here, we show that under the given assumptions, *no abstraction* is required: the *concrete* fixpoint iteration in question already terminates and returns the weakest inductive invariant, which happens to consist of universal formulas only. We conclude that universal invariants for the given class of FO transition systems are decidable.

Beyond that, we extend this class of FO transition systems by additionally allowing stratified guarded *resets* such as the two assignments before the loop

in Fig. 1. Guarded stratified resets are seemingly *easier* than updates, as they define their left-hand sides solely in terms of predicates of lower levels. In full generality, though, when there are both updates and resets, we *failed* to prove that universal invariants are decidable. We only succeed so—provided further (mild) restrictions are satisfied. Our results are that jointly, stratified guarded updates and resets can be allowed

- when resets refer to predicates at the highest and at the lowest level of the stratification only; or
- when all predicates of level at least 1, occur in right-hand sides only positively; or
- when all updates are not only guarded, but *strictly* guarded.

2 Basic Definitions

Assume that we are given a finite set of predicate names $\mathcal{R}$ together with a finite set of constant names $\mathcal{C}$. A *FO structure* $s = \langle I, \rho \rangle$ over a given universe $\mathcal{U}$ consists of an *interpretation* I of the predicates in $\mathcal{R}$, i.e., a mapping which assigns to each predicate $R \in \mathcal{R}$ of arity $k \geq 0$, a k-ary relation over $\mathcal{U}$, together with a valuation $\rho : \mathcal{C} \to \mathcal{U}$ which assigns to each constant name an element in $\mathcal{U}$. The *semantics* of FO (first-order) formulas as well as SO (second-order) formulas with free occurrences of predicates and variables in $\mathcal{R}$ and $\mathcal{C}$, respectively, is defined as usual. We write $s \models \varphi$ or $I, \rho \models \varphi$ to denote that φ is valid for the given interpretation I and valuation ρ as provided by s. For FO transition systems, we distinguish between the set $\mathcal{R}_{state}$ of *state predicates* and the disjoint set $\mathcal{A}$ of *input predicates*. While the values of constants as well as the interpretation of the state predicates constitute the state attained by the system, the input predicates are used to model (unknown) input from the environment or decisions of participating agents.

At each transition of a FO transition system, the system state s' after the transition is determined in terms of the system state s before the transition via a *substitution* θ. For each state predicate $R \in \mathcal{R}_{state}$, θ provides a FO formula to specify the interpretation of R after the transition in terms of the interpretation and valuation in s.

Technically, we introduce a set $\mathcal{Y} = \{y_i \mid i \in \mathbb{N}\}$ of distinct formal parameters where $\mathcal{C} \cap \mathcal{Y} = \emptyset$. For a predicate R of arity $k \geq 0$, we write $R\bar{y}$ for the literal $R(y_1, \ldots, y_k)$ and assume that each substitution θ maps each literal $R\bar{y}$, $R \in \mathcal{R}_{state}$, to some FO formula $\theta(R\bar{y})$ with predicates in $\mathcal{R}_{state} \cup \mathcal{A}$ and free variables either from $\mathcal{C}$ or occurring among the variables in $\bar{y}$. In case that $\theta(R\bar{y}) = \psi$ and $\theta(R'\bar{y}) = R'\bar{y}$ for all $R' \in \mathcal{R}_{state} \setminus \{R\}$, we also denote θ by $R\bar{y} := \psi$.

Example 1. In the example from Fig. 1, $\mathcal{R}_{state}$ consists of the predicates conf, auth, assign, report and discuss while $\mathcal{R}_{input}$ consists of the predicates $A_1 \ldots A_4$. No constants are needed, so $\mathcal{C} = \emptyset$. The edge from node 1 to 2, e.g., specifies a substitution θ that updates assign with

$$\theta(\mathsf{assign}(x, p)) = A_2(x, p) \wedge \neg \mathsf{conf}(x, p)$$

but does not change literals of predicates conf, auth, report or discuss. □

Applying θ to a FO formula φ results in the FO formula $\theta(\varphi)$ which is obtained from φ by replacing each literal $R\bar{z}$ with the FO formula $\theta(R\bar{y})[\bar{z}/\bar{y}]$. Here, $[\bar{z}/\bar{y}]$ represents the simultaneous substitution of the variables in $\bar{y}$ by the corresponding variables in $\bar{z}$.

Example 2. Consider formula φ that specifies that the author of a paper p should never be assigned to provide a review for p:

$$\varphi = \forall x, p. \neg \mathsf{assign}(x, p) \vee \neg \mathsf{auth}(x, p)$$

Applying the substitution θ from Example 1 results in

$$\theta(\varphi) = \forall x, p. \neg(A_2(x, p) \wedge \neg \mathsf{conf}(x, p)) \vee \neg \mathsf{auth}(x, p)$$

□

A FO transition system $\mathcal{T}$ (over the given sets $\mathcal{R}_{state}$ of predicates, $\mathcal{A}$ of input predicates and $\mathcal{C}$ of constant names) consists of a finite set of nodes V together with a finite set E of edges of the form $e = (u, \theta, v)$ where $u, v \in V$ and θ is a substitution of the predicates in $\mathcal{R}_{state}$. W.l.o.g., we assume that each substitution θ at some edge e always has occurrences of at most one input predicate, which we denote by A_e. For a given universe $\mathcal{U}$, a program state s attained at a program point is a FO structure for the predicates in $\mathcal{R}_{state}$ and the constants in $\mathcal{C}$ over the universe $\mathcal{U}$. Let S denote the set of all program states. A *configuration* of $\mathcal{T}$ is a pair $(v, s) \in V \times S$. A (finite) *run* τ of $\mathcal{T}$ starting in configuration (v_0, s_0) and ending at node v in state s, i.e., in configuration (v, s) is a sequence of configurations (v_i, s_i), $i = 0, \ldots, n$ where $(v_n, s_n) = (v, s)$ and for all $i = 1, \ldots, n$, there is some edge $e_i = (v_{i-1}, \theta_i, v_i) \in E$ such that for $s_{i-1} = \langle I, \rho \rangle$, $s_i = \langle I', \rho \rangle$ where for some interpretation R_i of the input predicate A_{e_i}, and every valuation $\rho_{\mathcal{Y}}$ of the formals, $I', \rho \oplus \rho_{\mathcal{Y}} \models R\bar{y}$ iff $I \oplus \{A_{e_i} \mapsto R_i\}, \rho \oplus \rho_{\mathcal{Y}} \models \theta(R\bar{y})$. Assume that we are given an initial node $v_0 \in V$ together with an initial hypothesis $\mathcal{H}$, i.e., a FO formula (with predicates in $\mathcal{R}_{state}$ and free variables only in $\mathcal{C}$) characterizing all possible initial states attained at v_0.

Example 3. According to the specification in Eq. (1) for the example transition system in Fig. 1, the single initial state is a pair of state 0 and the FO structure which interprets the relations auth, assign, report and discuss with the empty relation. □

Input predicates may take fresh interpretations whenever the substitution of the corresponding edge is executed. This should be contrasted to state predicates whose interpretations stay the same if they are not explicitly updated by the transition system. The constant interpretation of such predicates instead may be constrained by suitable background theories as provided, e.g., via conjuncts of the initial hypothesis.

Assume that Ψ assigns to each program point $v \in V$, a FO formula $\Psi[v]$. Then Ψ is a *valid invariant* (relative to the initial hypothesis $\mathcal{H}$), if every run τ

of the system starting in a configuration (v_0, s_0) with $s_0 \models \mathcal{H}$ and visiting some configuration (v, s), it holds that $s \models \Psi[v]$. Ψ is *inductive* if

$$\Psi[u] \to \theta(\Psi[v]) \qquad \text{forall } (u, \theta, v) \in E \tag{3}$$

If Ψ is inductive, then Ψ is a valid whenever

$$\mathcal{H} \to \Psi[v_0] \tag{4}$$

Indeed, it is this observation which is used in the Ivy project to verify distributed algorithms such as the Paxos protocol, essentially, by manually providing the invariant Ψ and verifying properties (3) and(4) via the theorem prover Z3 [8].

Not each valid invariant Ψ, though, is by itself inductive. If this is not yet the case, iterative *strengthenings* $\Psi^{(h)}, h \geq 0$, of Ψ may be computed as follows:

$$\begin{aligned} \Psi^{(0)}[u] &= \Psi[u] \\ \Psi^{(h)}[u] &= \Psi^{(h-1)}[u] \wedge \bigwedge_{e=(u,\theta,v)\in E} \forall A_e.\, (\theta(\Psi^{(h-1)}[v]) \quad \text{for } h > 0 \end{aligned} \tag{5}$$

For computing the next iterate in (5), universal SO quantification over the input predicate A_e is required in order to account for *every* input possibly occurring during a run at the given edge. As, e.g., noted in [25], $s \models \Psi^{(h)}[u]$ iff every run of length at most h starting in (u, s), ends in some configuration (u', s') with $s' \models \Psi[u']$. In particular, the assignment Ψ is a valid invariant iff $\mathcal{H} \to \Psi^{(h)}[v_0]$ for all $h \geq 0$. The iteration thus can be considered as computing the *weakest pre-condition* of the given invariant Ψ – as opposed to the *collecting semantics* of the FO transition system, which corresponds to the set of all configurations reachable from the set of all initial configurations $(v_0, s), s \models \mathcal{H}$. Whenever the fixpoint iteration (5) terminates, we obtain the *weakest* strengthening of the given invariant Ψ which is inductive. We have:

Lemma 1. *Let $\mathcal{T}$ be a FO transition system and let Ψ an invariant. Assume that for some $h \geq 0$, $\Psi^{(h)} = \Psi^{(h+1)}$ holds. Then $\Psi^{(h)}$ is the* weakest *inductive invariant implying Ψ. Moreover, Ψ is valid iff $\mathcal{H} \to \Psi^{(h)}[v_0]$.* □

In general, the required SO quantifier elimination may not always be possible, i.e., there need not always exist an equivalent FO formula [1], and even if SO quantifier elimination is always possible, the fixpoint iteration need not terminate. Non-termination may already occur when all involved predicates either have no arguments or are *monadic* [25]. Termination as well as effective computability can be enforced by applying *abstraction* (see, e.g., [24] for a general discussion). Applying an abstraction α amounts to computing a *sufficient* condition for the invariant Ψ to hold. Technically, an abstraction maps each occurring formula ψ to a formula $\alpha[\psi]$ (hopefully of a simpler form) so that $\alpha[\psi] \to \psi$. Subsequently, we list three examples for such strengthenings.

Example 4. Abstraction of existentials. In [19], formulas with universal SO quantifiers and universal as well as existential quantifiers are strengthened to

formulas with universal quantifiers only. The idea is to replace an existentially quantified subformula $\exists x.\varphi$ with a disjunction $\bigvee_{y \in Y} \varphi[y/x]$ where Y is the subset of constants and those universally quantified variables in whose scope φ occurs. So, the formula $\forall y_1, y_2.\exists x.R(x)$ is abstracted by $\forall y_1, y_2.R(y_1) \vee R(y_2)$. This abstraction is particularly useful, since SO universal quantifiers can be eliminated from universally quantified formulas.

Abstraction of Universals. Fixpoint iteration for universally quantified formulas still may not terminate due to an ever increasing number of quantified variables. The universally quantified variable x in an otherwise quantifier-free formula ψ in negation normal form can be removed by replacing each literal containing x with *false*. In this way, the formula $\forall x.\ (Rx \vee \neg Sy \vee Tz) \wedge (\neg Rx \vee \neg Ty)$ is strengthened to $(\neg Sy \vee Tz) \wedge \neg Ty$.

Abstraction of Conjunctions. Assume that the quantifier-free formula ψ is a conjunction of clauses. Then ψ is implied by the single clause c consisting of all literals which all clauses in ψ have in common. The formula $(Rx \vee \neg Sy \vee Tz) \wedge (Rx \vee Tz \vee \neg Tx)$, e.g., can be strengthened to $Rx \vee Tz$. □

In this paper, rather than focusing on using abstractions, we identify sufficient criteria when the concrete iteration (5) terminates without any further abstraction.

3 Stratification and Guardedness

Subsequently, we concentrate on initial conditions in the $\exists^*\forall^*$ fragment and *universal* invariants, i.e., where the invariant Ψ consists of *universal* FO formulas only. Already for this setting, non-termination of the inference algorithm may occur even without SO quantification when a single binary predicate is involved.

Example 5. Consider the FO transition system $\mathcal{T}$ over a monadic state predicate R, a binary state predicate E and a constant element a. $\mathcal{T}$ consists of a single state u with a single transition:

$$R(y) := R(y) \vee \exists z.\ E(y, z) \wedge R(z)$$

Consider the invariant $\Psi[u] = \neg R(a)$. Then for $h \geq 0$,

$$\Psi^{(h)}[u] = \neg R(a) \wedge \bigwedge_{k=1}^{h} \forall z_1, \ldots, z_k.\, \neg E(a, z_1) \vee \bigvee_{i=1}^{k-1} \neg E(z_i, z_{i+1}) \vee \neg R(z_k)$$

The weakest inductive invariant thus represents the set of elements which are *not* reachable from a via the edge relation E. This property is not expressible in FO predicate logic. Accordingly, $\Psi^{(h)}[u] \neq \Psi^{(h+1)}[u]$ must hold for all $h \geq 0$. □

Our goal is to identify useful non-trivial classes of FO transition systems where the fixpoint iteration is guaranteed to terminate. One ingredient for this definition is a stratification mapping $\lambda : \mathcal{R}_{state} \to \mathbb{N}$ which assigns to each state

predicate R a *level* $\lambda(R)$. Intuitively, this mapping is intended to describe how the information flows between predicates. Thereby, we use the convention that $\lambda(R) = 0$ only for predicates R which are never substituted, i.e., whose values stay the same throughout each run of the transition system.

We will consider substitutions which are *guarded* and *stratified*. A substitution θ is called *guarded* if it modifies at most one predicate $R \in \mathcal{R}_{state}$ at a time and is of one of the following forms:

$$Update: \quad R\bar{y} := R\bar{y} \vee \varphi \vee \exists \bar{z}.\, A\bar{y}\bar{z} \wedge \psi \tag{6}$$

$$Reset: \quad R\bar{y} := \varphi \vee \exists \bar{z}.\, A\bar{y}\bar{z} \wedge \psi \tag{7}$$

where $A \in \mathcal{R}_{input}$ is an input predicate and φ, ψ are quantifier-free FO formulas without occurrences of predicate A. If additionally, each predicate R' occurring in φ or ψ has level less than $\lambda(R)$, then θ is called *stratified*.

According to our definition, a *guarded* substitution only updates a single predicate R. We might wonder whether the single update restriction could be lifted by additionally allowing simultaneous updates of several predicates which are coupled via the same input predicate. For this extension, however, termination can no longer be guaranteed.

Lemma 2. *There exists a FO transition system $\mathcal{T}$ using stratified simultaneous guarded updates and resets, together with some universal invariant Ψ such that for each $h \geq 0$, $\Psi^{(h)}$ is universal FO definable, but $\Psi^{(h)}[u] \not\to \Psi^{(h+1)}[u]$ for some program point u.*

Proof. Consider the FO transition system $\mathcal{T}$ as shown in Fig. 2 for some binary predicate E, together with the invariant $\Psi = \{1 \mapsto \mathsf{error} \vee \neg\mathsf{hull}(a,b), 0, 2 \mapsto \top\}$ for constants a, b. Initially, the predicate hull is set to $\bot$. By executing the loop h times, either the error flag error is set to $\top$, or hull receives $k$fold compositions of E for $k = 0, \ldots, h$. Still, we can assign levels to the predicates used by $\mathcal{T}$ which meet the requirements of a stratification:

$$\lambda = \{E \mapsto 0, \mathsf{add} \mapsto 0, \mathsf{hull} \mapsto 1, \mathsf{error} \mapsto 2\}$$

For $h \geq 0$, we obtain $\Psi^{(h)}[1] =$

$$\bigwedge_{j=1}^{h} \forall y_1 \ldots y_j.\ \mathsf{error} \vee \neg\mathsf{hull}(a,b) \vee \neg\mathsf{hull}(a,y_1) \vee \bigvee_{i=1}^{j-1} \neg E(y_i, y_{i+1}) \vee \neg E(y_j, b)$$

For the required SO quantifier elimination of A_1, A_2, we note that in order to avoid error to be set to $\top$, $\mathsf{add}(x,y,z)$ must imply $\mathsf{hull}(x,y) \wedge E(y,z)$. In order to falsify the invariant at program point 1 whenever possible, thus, $A_1(x,y,z)$ should be set to $\mathsf{hull}(x,y) \wedge E(y,z)$, and $A_2(x,z,y)$ at least to $\mathsf{add}(x,y,z)$. Altogether thus, the weakest inductive invariant for program point 0 is given by $\mathsf{error} \vee \neg E^*(a,b)$ where E^* is the transitive closure of E. As transitive closure is not FO definable, we conclude that the fixpoint iteration cannot terminate. □

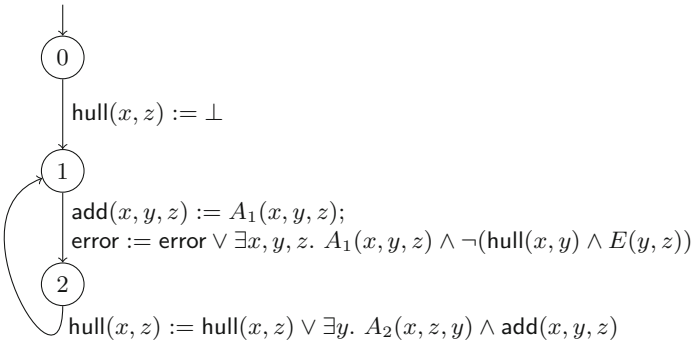

Fig. 2. FO transition system capturing transitive closure.

At the expense of slightly more complicated formulas for $\Psi^{(h)}$, the right-hand side for add could be brought into the form (6). Thus, the crucial issue which results in inexpressible weakest inductive invariants, is the use of the *same* input predicate in two simultaneous updates. In the next section, we indicate how to generally deal with SO quantifiers, once a guarded substitution has been applied.

4 Universal So Quantifier Elimination

It is well-known that universal SO quantifiers can be removed from otherwise quantifier-free formulas [12,19]. For example,

$$\forall A.\, R\bar{x} \vee A\bar{y} \vee \neg A\bar{z} \quad \longleftrightarrow \quad R\bar{x} \vee (\bar{y} = \bar{z})$$

where for $\bar{y} = (y_1, \ldots, y_k)$ and $\bar{z} = (z_1, \ldots, z_k)$, $\bar{y} = \bar{z}$ is a shortcut for the formula $(y_1 = z_1) \wedge \ldots \wedge (y_k = z_k)$. Interestingly, there are also cases where SO quantifier elimination is possible even in presence of FO existential quantifiers.

Example 6. Consider the substitution θ

$$R(y) := R(y) \vee \exists z.\, A(y, z) \wedge S(y, z)$$

In that case, $\theta(R(a) \vee \neg R(b))$ is given by

$$\begin{aligned} & \forall z_1.\, R(a) \vee \exists z.\, A(a, z) \wedge S(a, z) \vee \neg R(b) \wedge (\neg A(b, z_1) \vee \neg S(b, z_1)) \\ \longleftrightarrow\ & \forall z_1.\, (R(a) \vee \exists z.\, A(a, z) \wedge S(a, z) \vee \neg R(b)) \wedge \\ & \quad (R(a) \vee (\exists z.\, A(a, z) \wedge S(a, z)) \vee \neg A(b, z_1) \vee \neg S(b, z_1)) \end{aligned}$$

A closer inspection reveals that in this case, SO quantifier elimination of A is possible where $\forall A.\, \theta(R(a) \vee \neg R(b))$ is equivalent to

$$\begin{aligned} & \forall z_1.\, (R(a) \vee \neg R(b)) \wedge ((R(a) \vee (a = b) \wedge S(a, z_1)) \vee \neg S(b, z_1)) \\ \longleftrightarrow\ & \forall z_1.\, (R(a) \vee \neg R(b)) \wedge ((R(a) \vee (a = b) \wedge S(b, z_1)) \vee \neg S(b, z_1)) \\ \longleftrightarrow\ & \forall z_1.\, (R(a) \vee \neg R(b)) \wedge (R(a) \vee (a = b) \vee \neg S(b, z_1) \\ \longleftrightarrow\ & \forall z_1.\, R(a) \vee R(b) \wedge ((a = b) \vee \neg S(b, z_1)) \end{aligned}$$

In particular, the resulting FO formula has universal FO quantifiers only. □

The observation in Example 6 can be generalized.

Lemma 3. *1. If Ψ is of the form*

$$\bigvee_{i=1}^{n} (\exists \bar{z}.\, A\bar{y}_i\bar{z} \wedge \varphi[\bar{y}_i/\bar{y}]) \vee \bigvee_{j=1}^{m} (\forall \bar{z}.\neg A\bar{y}'_j\bar{z} \vee \neg\varphi[\bar{y}'_j/\bar{y}]) \tag{8}$$

for $n, m \in \mathbb{N}$ where φ is a FO formula without occurrences of A. Then $\forall A.\, \Psi$ is equivalent to

$$\bigvee_{j=1}^{m} (\bigvee_{i=1}^{n} \bar{y}_i = \bar{y}'_j) \vee (\forall \bar{z}.\neg\varphi[\bar{y}'_j/\bar{y}]) \tag{9}$$

2. If Ψ is of the form

$$\varphi' \vee \bigvee_{i=1}^{n} (\exists \bar{z}.\, A\bar{y}_i\bar{z} \wedge \varphi[\bar{y}_i/\bar{y}]) \vee \bigvee_{j=1}^{m} (\forall \bar{z}.\neg A\bar{y}'_j\bar{z} \vee \neg\varphi[\bar{y}'_j/\bar{y}]) \wedge \psi'_j \tag{10}$$

for $n, m \in \mathbb{N}$ where $\varphi, \varphi', \psi'_j$ all are FO formulas without occurrences of A. Then $\forall A.\, \Psi$ is equivalent to

$$\varphi' \vee \bigvee_{j=1}^{m} (\bigvee_{i=1}^{n} (\bar{y}_i = \bar{y}'_j) \vee (\forall \bar{z}.\neg\varphi[\bar{y}'_j/\bar{y}]) \wedge \psi'_j \tag{11}$$

Proof. For proving statement (1), we consider the negated formula $\exists A.\neg\Psi$ and apply Ackermann's lemma in order to remove existential SO quantification. We calculate:

$$\begin{aligned}
\exists A.\neg\Psi &\longleftrightarrow \exists \bar{z}_1 \ldots \bar{z}_m.\exists A.\forall \bar{z}.\ \textstyle\bigwedge_{i=1}^{n}\ \bigwedge_{j=1}^{m} (\neg A\bar{y}_i\bar{z} \vee \neg\varphi[\bar{y}_i/\bar{y}]) \wedge A\bar{y}'_j\bar{z}_j \wedge \varphi[\bar{y}'_j/\bar{y}, \bar{z}_j/\bar{z}] \\
&\longleftrightarrow \exists \bar{z}_1 \ldots \bar{z}_m.\ \textstyle\bigwedge_{i=1}^{n}\ \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \vee \neg\varphi[\bar{y}_i/\bar{y}, \bar{z}_j/\bar{z}]) \wedge \bigwedge_{j=1}^{m} \varphi[\bar{y}'_j/\bar{y}, \bar{z}_j/\bar{z}] \\
&\longleftrightarrow \exists \bar{z}_1 \ldots \bar{z}_m.\ \textstyle\bigwedge_{i=1}^{n}\ \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \vee \neg\varphi[\bar{y}_j/\bar{y}, \bar{z}_j/\bar{z}]) \wedge \bigwedge_{j=1}^{m} \varphi[\bar{y}'_j/\bar{y}/\bar{z}_j/\bar{z}] \\
&\longleftrightarrow \exists \bar{z}_1 \ldots \bar{z}_m.\ \textstyle\bigwedge_{i=1}^{n}\ \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \vee \neg\varphi[\bar{y}_j/\bar{y}]) \wedge \varphi[\bar{y}'_j/\bar{y}, /\bar{z}_j/\bar{z}] \\
&\longleftrightarrow \textstyle\bigwedge_{i=1}^{n}\ \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \vee \exists \bar{z}.\ \varphi[\bar{y}'_j/\bar{y}])
\end{aligned}$$

where the last formula is equivalent to the negation of formula (9). The second statement then follows from statement (1) by distributivity. □

Interestingly, the same result is obtained when the existentially quantified variables $\bar{z}$ do not occur as arguments to the input predicate A.

Lemma 4. *1. If Ψ is of the form*

$$\bigvee_{i=1}^{n} A\bar{y}_i \wedge (\exists \bar{z}.\, \varphi[\bar{y}_i/\bar{y}]) \vee \bigvee_{j=1}^{m} \neg A\bar{y}'_j \vee (\forall \bar{z}.\neg\varphi[\bar{y}'_j/\bar{y}]) \tag{12}$$

for $n, m \in \mathbb{N}$ where φ is a FO formula without occurrences of A. Then $\forall A.\, \Psi$ is equivalent to

$$\bigvee_{j=1}^{m} (\bigvee_{i=1}^{n} \bar{y}_i = \bar{y}'_j) \vee (\forall \bar{z}.\neg\varphi[\bar{y}'_j/\bar{y}]) \tag{13}$$

2. If Ψ is of the form

$$\varphi' \vee \bigvee_{i=1}^{n} A\bar{y}_i \wedge (\exists \bar{z}.\, \varphi[\bar{y}_i/\bar{y}]) \vee \bigvee_{j=1}^{m} (\neg A\bar{y}'_j \vee (\forall \bar{z}.\neg\varphi[\bar{y}'_j/\bar{y}])) \wedge \psi'_j \tag{14}$$

for $n, m \in \mathbb{N}$ where $\varphi, \varphi', \psi'_j$ all are FO formulas without occurrences of A. Then $\forall A.\, \Psi$ is equivalent to

$$\varphi' \vee \bigvee_{j=1}^{m} (\bigvee_{i=1}^{n} (\bar{y}_i = \bar{y}'_j) \vee (\forall \bar{z}.\neg\varphi[\bar{y}'_j/\bar{y}]) \wedge \psi'_j \tag{15}$$

Proof. For proving statement (1), we again consider the negated formula $\exists A.\neg\Psi$ and apply Ackermann's lemma in order to remove existential SO quantification. By introducing the shortcut Φ for $\exists \bar{z}.\, \varphi$, we calculate:

$$\begin{aligned}
\exists A.\neg\Psi &\longleftrightarrow \exists A.\ \textstyle\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{m} (\neg A\bar{y}_i \vee \neg\Phi[\bar{y}_i/\bar{y}]) \wedge A\bar{y}'_j \wedge \neg\Phi[\bar{y}'_j/\bar{y}] \\
&\longleftrightarrow \textstyle\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \vee \neg\Phi[\bar{y}_i/\bar{y}]) \wedge \bigwedge_{j=1}^{m} \Phi[\bar{y}'_j/\bar{y}] \\
&\longleftrightarrow \textstyle\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \vee \neg\Phi[\bar{y}_j/\bar{y}]) \wedge \bigwedge_{j=1}^{m} \Phi[\bar{y}'_j/\bar{y}] \\
&\longleftrightarrow \textstyle\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \vee \neg\Phi[\bar{y}_j/\bar{y}]) \wedge \Phi[\bar{y}'_j/\bar{y}] \\
&\longleftrightarrow \textstyle\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{m} ((\bar{y}_i \neq \bar{y}'_j) \wedge \Phi[\bar{y}'_j/\bar{y}])
\end{aligned}$$

where the last formula is equivalent to the negation of formula (13). Again, the second statement then follows from statement (1) by distributivity. □

In light of Lemmas 3 and 4, we introduce simplified versions of guarded updates and resets where the input predicate no longer occurs in the scope of existential quantifiers:

$$\textit{Simplified Update:} \qquad R\bar{y} := R\bar{y} \vee \varphi \vee A\bar{y} \wedge \exists \bar{z}.\, \psi \tag{16}$$

$$\textit{Simplified Reset:} \qquad R\bar{y} := \varphi \vee A\bar{y} \wedge \exists \bar{z}.\, \psi \tag{17}$$

As a first corollary, we obtain:

Corollary 1. *Assume that θ is a guarded update of the form (6) (guarded reset of the form (7)), and that θ' is the corresponding simplified update (16) (simplified reset (17)). Then for every universal FO formula Ψ,*

$$\forall A.\, \theta(\Psi) \quad \longleftrightarrow \quad \forall A.\, \theta'(\Psi)$$

□

In light of Corollary 1, we subsequently consider FO transition systems with *simplified* guarded updates and resets only.

Example 7. Consider the second update in the loop of the transition system from Fig. 1. Its simplified variant removes r_1 and r_2 from the signature of A_4:

$$\begin{aligned}
\mathsf{discuss}(x_1, x_2, p, d) := \mathsf{discuss}(x_1, x_2, p, d) \ &\vee A_4(x_1, x_2, p, d) \wedge \\
&\exists r_1, r_2.\ \mathsf{report}(x_1, p, r_1) \wedge \mathsf{report}(x_2, p, r_2)
\end{aligned}$$

Let θ_4 denote this simplified update, and consider the invariant (2) from the introduction. Application of θ_4 results in the formula

$$\begin{aligned}\forall x_1, x_2, p, d, r_1, r_2.\neg\mathsf{discuss}(x_1, x_2, p, d) \wedge \\ (\neg A_4(x_1, x_2, p, d) \vee \neg\mathsf{report}(x_1, p, r_1) \vee \neg\mathsf{report}(x_2, p, r_2)) \vee \\ (\neg\mathsf{auth}(x_1, p) \wedge \neg\mathsf{auth}(x_2, p))\end{aligned}$$

Since A_4 only occurs negatively, universal SO quantifier elimination of A_4 yields

$$\begin{aligned}\forall x_1, x_2, p, d, r_1, r_2.\neg\mathsf{discuss}(x_1, x_2, p, d) \wedge \\ (\neg\mathsf{report}(x_1, p, r_1) \vee \neg\mathsf{report}(x_2, p, r_2)) \vee \\ (\neg\mathsf{auth}(x_1, p) \wedge \neg\mathsf{auth}(x_2, p))\end{aligned}$$

□

Corollary 2. *Assume Ψ is a formula of the form (14). Then $\forall A.\,\Psi \longleftrightarrow \theta(\Psi)$ where θ is given by*

$$A\bar{y} := \bigwedge_{i=1}^{n} (\bar{y}_i \neq \bar{y}) \tag{18}$$

The definition (18) thus provides us with the *worst* adversarial strategy to defeat the proposed invariant. As another consequence of Lemma 3, we find that in presence of subsequent SO quantifier elimination, the effect of a guarded substitution of the forms (16) or (17) could also be simulated by the corresponding *nonuniform* substitutions:

$$\begin{aligned} R\bar{y} &:= R\bar{y} \vee \varphi \vee A\bar{y} \\ \neg R\bar{y} &:= \neg R\bar{y} \wedge \neg\varphi \wedge (\neg A\bar{y} \vee \forall\bar{z}.\neg\psi) \end{aligned} \tag{19}$$

and

$$\begin{aligned} R\bar{y} &:= \varphi \vee A\bar{y} \\ \neg R\bar{y} &:= \neg\varphi \wedge (\neg A\bar{y} \vee \forall\bar{z}.\neg\psi) \end{aligned} \tag{20}$$

respectively. Here, *nonuniform* means that positive and negative occurrences of literals are substituted differently. We have:

Corollary 3. *Assume that θ is a guarded substitution of the form (16) or (17). Assume that θ' is the nonuniform substitution of the corresponding form (19) or (20), respectively. Then for every universal formula Ψ,*

$$\forall A.\,\theta(\Psi) \quad\longleftrightarrow\quad \forall A.\,\theta'(\Psi)$$

□

Finally, as another important consequence of Lemma 3, we obtain:

Theorem 1. *Assume that $\mathcal{T}$ is a FO transition system with guarded (simplified) updates and resets only, and Ψ a universal FO invariant.*

1. *The iterates* $\Psi^{(h)}[u], h \geq 0$, *in* (5) *all are effectively equivalent to universal FO formulas.*
2. *The iteration terminates, i.e.,* $\Psi^{(h)} = \Psi^{(h+1)}$ *for some* $h \geq 0$, *iff for each program point* u, *the weakest strengthening of all iterates* $\Psi^{(h)}[u]$ *is FO-definable.*

Proof. Due to Lemmas 3 and 4, for each universal FO formula φ and each guarded (simplified) update or reset θ with input predicate A, $\forall A.\,(\theta\varphi)$ is equivalent to a universal FO formula. That implies statement (1). Now assume for for each $h \geq 0$ and each $v \in V$, $\Phi^{(h)}[v]$ is FO definable. Then due to the compactness theorem for FO predicate logic [5], there is some $h \geq 0$ such that $\Psi^{(h)}[v] \leftrightarrow \Psi^{(h+j)}[v]$ holds for all $v \in V$ and $j \geq 0$, iff for each $v \in V$, the conjunction $\bigwedge_{h\geq 0} \Psi^{(h)}[v]$ is again FO definable. □

Example 8. Consider again the specification from Fig. 1, and let $\theta_1, \theta_2, \theta_3$, and θ_4 denote the simplified substitutions occurring therein. Assume that Ψ equals the universal formula in (2), and we are interested in its validity at program point 2 of the transition system. The formula $\forall A_3.\,\theta_3(\forall A_4.\,\theta_4(\Psi))$ is given by

$$\begin{aligned}
&\forall A_3.\,\theta_3(\forall x_1, x_2, p, d, r_1, r_2.\neg\mathsf{discuss}(x_1, x_2, p, d) \wedge \\
&\qquad (\neg\mathsf{report}(x_1, p, r_1) \vee \neg\mathsf{report}(x_2, p, r_2)) \vee (\neg\mathsf{auth}(x_1, p) \wedge \neg\mathsf{auth}(x_2, p)) \\
\longleftrightarrow\ &\forall x_1, x_2, p, d, r_1, r_2.\neg\mathsf{discuss}(x_1, x_2, p, d) \wedge \\
&\qquad (\neg\mathsf{report}(x_1, p, r_1) \wedge \neg\mathsf{assign}(x_1, p) \vee \neg\mathsf{report}(x_2, p, r_2) \wedge \neg\mathsf{assign}(x_2, p)) \vee \\
&\qquad (\neg\mathsf{auth}(x_1, p) \wedge \neg\mathsf{auth}(x_2, p))
\end{aligned}$$

The resulting formula Ψ' already equals the fixpoint for the loop. Since the predicate assign only occurs negatively in Ψ' and conf only negatively in the right-hand side for assign, the formula $\forall A_1.\theta_1(\forall A_2.\theta_2(\Phi'))$ construction from Ψ' via the substitution θ_{assign} defined by

$$\mathsf{assign}(y_1, y_2) := \neg\mathsf{auth}(y_1, y_2)$$

This means the formula Ψ'' for the initial node of the transition system is given by

$$\begin{aligned}
&\forall x_1, x_2, p, d, r_1, r_2.\neg\mathsf{discuss}(x_1, x_2, p, d) \wedge \\
&\qquad (\neg\mathsf{report}(x_1, p, r_1) \wedge \mathsf{auth}(x_1, p) \vee \neg\mathsf{report}(x_2, p, r_2) \wedge \mathsf{auth}(x_2, p)) \vee \\
&\qquad (\neg\mathsf{auth}(x_1, p) \wedge \neg\mathsf{auth}(x_2, p))
\end{aligned}$$

By the initial condition $\mathcal{H}$ from the introduction, $\neg\mathsf{discuss}(x_1, x_2, p, d)$ holds at the initial node of the transition system, as well as $\neg\mathsf{report}(x_1, p, r_1)$ and $\neg\mathsf{report}(x_2, p, r_2)$ for all x_1, x_2, p, d, r_1, r_2. Therefore, $\mathcal{H}$ implies Ψ'', and the property Ψ at the exit of the transition system is valid. □

In this section we have shown comprehensively how to eliminate universal SO quantifiers introduced by guarded updates in a FO transition system and introduced a non-uniform variant of any guarded updates and resets which removes all possibly introduced existential FO quantifiers. In the next two sections, we will apply these results to FO transition systems which additionally are stratified.

5 Stratified Guarded Updates

In [19], termination was announced for FO transition systems with stratified guarded updates where instantiation of existential quantifiers was applied as an *abstraction* to enforce all occurring formulas to be universal. Here, we improve on that result in two respects. First, we present a proof that termination can also be guaranteed without any abstraction. Second, we generalize the setting to allow stratified guarded resets—at least at the maximal and minimal levels.

Theorem 2. *Assume that $\mathcal{T}$ is a FO transition system where each occurring substitution is stratified guarded with the restriction that resets only occur for predicates of level 1 and the maximal level L. Then for every universal invariant Ψ, the weakest inductive invariant is again universal and can effectively be computed.*

Proof. W.l.o.g., we assume that each occurring substitution is a *simplified* update or reset, i.e., either of the form (16) or (17). We show that there is some $h \geq 0$, so that $\Psi^{(h+1)} = \Psi^{(h)}$. Since by Lemma 4, $\Psi^{(h)}[u]$ is a universal formula for all $h \geq 0$ and program points u, the statement of the theorem follows.

Assume that each simplified update θ of a predicate R always is specified by means of the *same* input predicate A_R. Let Θ denote the finite set of stratified guarded substitutions occurring in $\mathcal{T}$, and Φ a universal FO formula. Let $\pi = \theta_N, \ldots, \theta_1$ be any sequence of nonuniform substitutions where for each $i = 1, \ldots, N$, $\theta_i = \theta_i'[A_i/A_R]$ holds for a fresh input predicate A_i, and a nonuniform substitution θ_i' of the form (19) corresponding to a simplified update or reset $\theta'' \in \Theta$ with left-hand side $R\bar{y}$.

Lemma 5. *There is some number V only depending on Φ and Θ so that $\pi(\Phi) = \theta_N(\ldots\theta_1(\Phi)\ldots) = \bigwedge_{h=t}^{N}(\forall \bar{z}_t.c_t)$ for clauses c_t where the number of FO variables in $\bar{z}_t$ is bounded by V. In particular, V is independent of the number N of substitutions in π.*

Given Lemma 5, the number of argument tuples $\bar{z}$ of occurring literals $A_i\bar{z}$ in any c_t is bounded. Due to Corollary 2, a bounded number of substitutions of the form (18) therefore suffices to realize SO quantifier elimination of $A_1, \ldots, A_N$ in c_t. As a consequence, the number of universal FO formulas possibly occurring in each conjunct of $\forall A_1 \ldots A_N.\, \pi(\Phi)$, and thus also the number of conjunctions of these formulas is finite. Accordingly, there must be some $h \geq 0$ so that $\Phi^{(h+1)} = \Phi^{(h)}$, and the theorem follows. It therefore remains to prove Lemma 5.

Proof (of Lemma 5). Let us first consider the case where there is no reset of predicates at maximal level L. We introduce a dedicated class of formulas g as finite conjunctions of *generalized* clauses c which are built up according to the following abstract grammar

$$
\begin{array}{lll}
g & ::= \top \;\mid\; c \wedge g & \\
c & ::= c_0 \;\mid\; A\bar{a} \vee c' \;\mid\; f_{R\bar{b}} \vee c' \;\mid\; o_{\bar{b}} \vee c' & \\
f_{R\bar{b}} & ::= \neg R\bar{b} \wedge \forall \bar{z}_R.\, \bigwedge_{n=1}^{r}(\neg A_n\bar{b} \vee c_n) & \\
o_{\bar{b}} & ::= \forall \bar{z}.\, \bigwedge_{n=1}^{r}(\neg A_n\bar{b} \vee c_n) &
\end{array}
$$

where c_0 is an ordinary clause without occurrences of input predicates, R is a predicate, A, A_n are input predicates, $\bar{a}, \bar{b}$ are sequences of arguments, $\bar{z}_R$ is a sequence of fresh variables whose length only depends on R, and formulas $o_{\bar{b}}$ where all state predicates are of level 0. A formula $f_{R\bar{b}}$ is also called *negation* tree with head $\neg R\bar{b}$, while we call a formula $o_{\bar{b}}$ a level 0 *chunk*. Moreover,

(a) All literals occurring in the generalized clauses c_n inside the conjunction within $f_{R\bar{b}}$ are of levels less than $\lambda(R)$;
(b) For any two negation trees φ_1, φ_2 with identical head $\neg R\bar{b}$, there is some formula Δ so that either $\varphi_1 = \varphi_2 \wedge \Delta$ or vice versa, $\varphi_2 = \varphi_1 \wedge \Delta$ holds.

Φ can be brought into the form $\forall \bar{z}.\ \bigwedge_{t=1}^{m} c_t$ where each c_t is an ordinary clause without occurrences of input predicates, i.e., a plain disjunction of literals and (dis-)equalities. Therefore, now consider a single generalized clause c which satisfies properties (a) and (b). We show that for each nonuniform update substitution θ of the form

$$\begin{aligned} R\bar{y} &:= R\bar{y} \vee \varphi \vee A_h\bar{y} \\ \neg R\bar{y} &:= \neg R\bar{y} \wedge \neg\varphi \wedge (\neg A_h\bar{y} \vee \neg\forall \bar{z}.\neg\psi) \end{aligned}$$

$\theta(c)$ can again be represented as a conjunction of generalized clauses satisfying properties (a) and (b), and whose free variables are all contained in the set of free variables from c and θ. Assume that c is of the form $c' \vee \bigvee_{i=1}^{s} R\bar{a}_i \vee \bigvee_{j=1}^{t} f_{R\bar{b}_j}$ where c' is a generalized clause without further top-level occurrences either of positive literals $R\bar{a}'$ or negation trees with head $\neg R\bar{b}'$ for any $\bar{a}', \bar{b}'$, and $f_{R\bar{b}_j} = \neg R\bar{b}_j \wedge \forall \bar{z}_R.\ \bigwedge_{\nu=1}^{u_j}(\neg A_{j,\nu}\bar{b}_j \vee c_{j,\nu})$ is a negation tree with head $\neg R\bar{b}_j$. Then

$$\begin{aligned} \theta(c) = \bigwedge_{c_1,\ldots,c_s \in \mathcal{C}} \bigwedge_{J \subseteq [1,t]} \bigwedge_{j \in J, \bar{c}_j \in \bar{\mathcal{C}}} \theta(c') \vee \\ \bigvee_{i=1}^{s} R\bar{a}_i \vee A\bar{a}_i \vee c_i[\bar{a}_i/\bar{y}] \ \vee \bigvee_{j \in J} \bar{c}_j[\bar{b}_j/\bar{y}] \ \vee \\ \bigvee_{j \notin J} \neg R\bar{b}_j \ \wedge (\forall \bar{z}_R.\ A\bar{b}_j \vee \psi[\bar{b}_j/\bar{y}]) \ \wedge \bigwedge_{\nu=1}^{u_j}(\neg A_{j,\nu}\bar{b}_j \vee \psi_{j,\nu}) \end{aligned}$$

where $\mathcal{C}$ and $\bar{\mathcal{C}}$ are the sets of clauses in the normal forms of φ and $\neg\varphi$, respectively. The resulting formula can indeed be represented as a conjunction of generalized clauses satisfying property (a). Concerning property (b), we observe that for every fresh negative literal property (b) trivially holds, while for existing negation trees, this property is preserved. If on the other hand, θ is a reset of a predicate at level 1, $\theta(c)$ is a conjunction of generalized clauses where some negation trees have been replaced by level 0 chunks. In particular, properties (a) and (b) still hold.

Assume now that we are given a generalized clause c satisfying properties (a) and (b). Then c is called *flat up to level* i, if the roots of all negation trees occurring in c with a nonempty conjunction, have level at most i, and for every predicate R of level i and every possible argument tuple $\bar{b}$, there is at most one negation tree with head $\neg R\bar{b}$. For a generalized clause c which is flat up to level i, we define the transformation $\mathsf{flatten}_i$ as follows. Assume that c is of the form

$$c' \vee \bigvee_{j=1}^{t} \neg R_j\bar{b}_j \wedge \forall \bar{z}_j.\ \bigwedge_{\nu=1}^{u_j}(\neg A_{j,\nu}\bar{b}_j \vee c_{j,\nu})$$

where the $\neg R_j \bar{b}_j$ represent all occurrences of negated literals of level i. Then

$$c \quad \longleftrightarrow \quad \bigwedge_{J=\{j_1<\ldots<j_k\}\subseteq[1,t]} \bigwedge_{\nu_1\in[1,u_{j_1}],\ldots,\nu_k\in[1,u_{j_k}]} (\forall \bar{z}_{j_1} \ldots \bar{z}_{j_k}.\, c' \vee \bigvee_{j\notin J} R_j \bar{b}_j \vee \bigvee_{l=1}^{k} \neg A_{j_l,\nu_l} \bar{b}_j \vee c_{j_l,\nu_l})$$

In each quantified clause $\forall \bar{z}_{j_1} \ldots \bar{z}_{j_k}.\, c'$ in the conjunction, all occurring negation trees have level less than i. Now due to property (2), c' can be simplified so that for each negated literal $R'\bar{b}$ where R' is of level $i-1$, there is at most one negation tree. The resulting conjunction of quantified clauses is denoted by $\mathsf{flatten}_i\, c$.
To compute a bound on the number of possible argument variables, let us introduce the following structural parameters:

v	—	the number of variables occurring in Φ
L	—	the number of levels of predicates
r	—	maximal arity of a predicate
m	—	maximal number predicates at some level i
l	—	maximal length of $\bar{z}$ in subformulas $\forall \bar{z}.\neg\psi$ occurring in the substitutions from Θ

Successive application of $\mathsf{flatten}_L, \ldots, \mathsf{flatten}_1$ allows us to construct for a generalized clause c satisfying properties (a) and (b), an equivalent conjunction of formulas $\forall \bar{z}'.\, c'$ where c' is disjunction of literals, (dis-)equalities and level 0 chunks $o_{\bar{b}}$ only, and $\bar{z}'$ is the list of globally bound variables occurring freely in c'.

For $i = L, \ldots, 1$, we inductively determine a bound V_i to the number of distinct FO variables possibly occurring as arguments of literals at level i in a clause c'. For $i = L$, we can set $V_L = v$, since the only literals at level L occurring in c' already must have occurred in Φ. Therefore, assume that $i < L$ and a bound V_{i+1} has already been found. Then V_i can be bound as follows: Given the number V_{i+1}, the number of literals of predicates at level $i+1$ can be bound by $m \cdot V_{i+1}^r$. For each of these literals, a fresh list of variables of length at most l can be provided. Accordingly,

$$V_i = V_{i+1} + l \cdot m \cdot V_{i+1}^r \leq (1 + l \cdot m) \cdot V_{i+1}^r$$

Altogether, this means that the total number of variables possibly occurring in literals of c' (outside of level 0 chunks) at level at least 0 is bounded by

$$V \leq \begin{cases} (1 + l \cdot m)^L \cdot v & \text{if } r = 1 \\ (1 + l \cdot m)^{\frac{r^L - 1}{r-1}} \cdot v^{r^L} & \text{if } r > 1 \end{cases} \tag{21}$$

Now given that there is a bound V_1 to the number of variables possibly occurring as arguments of predicates at level 1, there is also only a bounded number O of non-equivalent subformulas $o_{\bar{b}}$ (after SO quantifier elimination) in any of the generalized clauses from $\mathsf{flatten}_1(\ldots \mathsf{flatten}_L(c')\ldots)$. Accordingly, $V_0 + O \cdot l$ bounds the number of variables occurring in equalities, disequalities and literals of predicates at level 0.

Let us finally also consider the case when additionally resets of predicates at maximal level L occur. Such a reset for a predicate R takes effect at most once. It thus introduces one fresh list of universally quantified variables for each occurrence $\neg R\bar{b}$ of the negated the negated literal at most once where we w.l.o.g. may even assume that the list of outside universal quantifiers of the negation tree for that literal can be reused. Thus, no further universal quantifiers are introduced. Altogether, therefore, the number of FO variables in quantified clauses $\forall \bar{z}'.c'$ contained in $\pi(\Phi)$ remains bounded. This completes the proof of Lemma 5. □

We remark that Theorem 2 remains true if there are predicates R' with stratified guarded updates as well as resets also at non-extremal levels—given that neither their updates nor their resets introduce FO variables, i.e., the variable lists $\bar{z}$ in (6) and (7) ((16) and (17)) are empty. In general, though, the proof technique of Theorem 2 cannot easily be extended to FO transition systems with arbitrary resets of the form (7), since then conjunctions of the form $o_{\bar{b}}$ with non-empty lists of quantified variables may also occur at higher levels—where it is no longer clear how to prove that their number is finite.

6 Allowing Guarded Stratified Resets

We would like to extend Theorem 2 from the last section to FO transition systems which additionally have resets at arbitrary levels. We succeed in doing so in two special cases (see Theorems 3 and 4, respectively). Let us call an update *strictly* guarded it it is of the form:

$$R\bar{y} := R\bar{y} \vee A\bar{y} \wedge \exists \bar{z}.\, \psi \tag{22}$$

for some predicate R and quantifier-free FO formula ψ without occurrences of A. Furthermore, let us call an update or reset θ *positive* if all predicates only occur positively in the right-hand side.

Theorem 3. *Consider a FO transition system $\mathcal{T}$ where all substitutions are stratified, guarded, and all substitutions of predicates not of level 0 are positive. Then for every universal invariant Ψ, the weakest inductive invariant is again universal and can effectively be computed.*

Proof. Let Θ denote the set of substitutions occurring in $\mathcal{T}$. As in the proof of Theorem 2, let $\pi = \theta_N, \ldots, \theta_1$ be any sequence of nonuniform substitutions where for each $i = 1, \ldots, N$, $\theta_i = \theta_i'[A_i/A_R]$ holds for a fresh input predicate A_i, and a nonuniform substitution θ_i' of the form (19) corresponding to an update or reset substitution $\theta'' \in \Theta$ with left-hand side $R\bar{y}$. Let $\bigwedge_{j=1}^{M}(\forall \bar{z}_j.\, c_j)$ denote the conjunction of quantified generalized clauses for $\pi(\Phi)$—now possibly also with subformulas $o_{\bar{b}}$ holding predicates of level > 0. Then each FO variable x occurring in a positive literal $A_i\bar{a}$ in any c_j, already occurs in Φ. In light of Corollary 2, it therefore suffices to use only a globally bounded number of input predicates in each c_j. If the number of predicate symbols is bounded, then also the number of generalized clauses as well as the number of non-equivalent formulas $\forall A_1 \ldots A_N.\, \pi(\Phi)$—implying that for every universal invariant Ψ, $\Psi^{(h+1)} = \Psi^{(h)}$ for some $h \geq 0$. From that, the statement of the theorem follows. □

The proof argument for Theorem 3 cannot easily be extended to unrestricted stratified guarded substitutions. In presence of *negated* literals in substitutions, it is no longer the case that the arguments of positive literals $R\bar{a}$ occurring in $\pi(\Phi)$ have already occurred in Φ, so for the next result we have to rely on a different proof strategy.

Theorem 4. *Consider a FO transition system $\mathcal{T}$ where all substitutions are guarded and stratified. Assume furthermore that all updates are* strictly *guarded. Then for every universal invariant Ψ, the weakest inductive invariant is again universal and can effectively be computed.*

Proof. For this proof, it is convenient to use the notation $\Phi \ni \forall \bar{x}.\, c$ for a universal FO formula Φ, a clause c, and a list $\bar{x}$ of distinct variables so that for the prenex CNF $\forall \bar{z}.\, c_1 \wedge \ldots \wedge c_m$ of Φ, c occurs among the c_j, and $\bar{x}$ is the subsequence of variables in $\bar{z}$ which occur in c. We rely on the following technical lemma:

Lemma 6. *Assume that c is a clause and θ a stratified reset or stratified strictly guarded update with input predicate A which substitutes a predicate R with $\lambda(R) = s$. Let c' be a clause with $\forall A.\, \theta(c) \ni \forall \bar{x}.\, c'$ where $\bar{x}$ is the list of newly introduced variables in c'. Then either $c = c'$ and $\bar{x}$ is empty, or the number of literals at level s of c' is less than the corresponding number of c.*

Proof. Assume that the clause c is of the form

$$c_0 \vee R\bar{y}_1 \vee \ldots \vee R\bar{y}_n \vee \neg R\bar{y}'_1 \vee \cdots \vee \neg R\bar{y}'_m$$

where c_0 does not contain the predicate R. If θ is a reset, all literals containing R are eliminated. Therefore, the assertion of the lemma trivially holds. Now assume that θ is a strictly guarded update, i.e., of the form (22). Then by Lemma 3,

$$\begin{aligned}\forall A.\, \theta(R_i\bar{y}) &\longleftrightarrow c_0 \vee \textstyle\bigvee_{j=1}^{m} \neg R\bar{y}'_j \wedge (\bigvee_{i=1}^{n}(\bar{y}_i = \bar{y}'_j) \vee \neg\psi[\bar{y}'_j/\bar{y}]) \\ &\longleftrightarrow \textstyle\bigwedge_{J\subseteq[1,m]} \forall \bar{z}_J.\, (c_0 \vee \bigvee_{j\notin J} \neg R\bar{y}'_j \vee \\ &\qquad\qquad \textstyle\bigvee_{j\in J}\bigvee_{i=1}^{n}(\bar{y}_i = \bar{y}'_j) \vee \neg\psi[\bar{y}'_j/\bar{y}, \bar{z}_j/\bar{z}])\end{aligned}$$

where $\bar{z}_j$ is a fresh list of FO variables of the same length as $\bar{z}$, and $\bar{z}_J$ is the concatenation of all lists $\bar{z}_j, j \in J$. In particular for $J = \emptyset$, $\bar{z}_J$ is empty and the corresponding clause equals c. If on the other hand $J \neq \emptyset$, the number of negated literals occurring in the clause has decreased. □

By Lemma 6, the number of literals at level s therefore either decreases, or the clause stays the same. Let Θ denote a finite set of stratified guarded substitutions where all updates in Θ are strictly guarded, and let c_0 denote any clause. Consider a sequence $(\theta_t, \forall \bar{x}_t.c_t), t \geq 1$, where for all $t \geq 1$, $\theta_t \in \Theta$ with some input predicate A_t, and $\forall A_t.\, (\theta_t c_{t-1}) \ni \forall \bar{x}_t.\, c_t$ holds. We claim that then there is some $t' \geq 1$ so that $c_{t'} = c_{t''}$ and $\bar{x}_{t''}$ is empty for all $t'' > t'$.

In order to prove that claim, we introduce for $t \geq 1$, the vector $v_t = (v_{t,L}, \ldots, v_{t,1}) \in \mathbb{N}^L$ where L is the maximal level of a predicate in $\mathcal{R}_{state}$, and $v_{t,i}$ is the number of literals with predicates of level i. By Lemma 6, it holds for all $t \geq 0$, that either $c_t = c_{t+1}$ and $\bar{z}_t$ is empty, or $v_t > v_{t+1}$ w.r.t. the lexicographic order on $\mathbb{N}^L$. Since the lexicographical ordering on $\mathbb{N}^L$ is well-founded, the claim follows. We conclude that the set of quantified clauses $\forall \bar{z}.c$ with $\Psi^{(h)}[u] \ni \forall \bar{z}.c$ for any u and h, is finite. From that, the statement of the theorem follows. □

Theorem 4 leaves open the case of transition systems with stratified guarded resets and stratified guarded updates of which some are not strictly guarded. To these, the presented proof technique cannot be easily extended. The reason is that a non-strictly guarded update θ for some predicate R, when applied to some clause c, may result in a quantified clause $\forall \bar{z}.\, c'$ with $\forall A.\theta(c) \ni \forall \bar{z}.\, c'$ so that neither $c = c'$ holds nor does the number of literals $\neg R\bar{b}$ decrease.

7 Conclusion

We have investigated FO transition systems where all substitutions are either guarded updates or guarded resets. For these, we observed that the exact weakest pre-condition of a universal FO formula is again a universal FO formula, thus allowing us to realize a fixpoint computation of iterated strengthening for proving the validity of universal invariants. In order to identify sub-classes of FO transition systems where termination can be guaranteed, we relied on a natural notion of stratification. Here, we were able to prove termination (and thus decidability) for three interesting sub-classes of stratified guarded FO transition systems. However, it remains as an open question whether termination can be proven for *all* FO transition systems with stratified guarded updates and resets.

The results of our paper can immediately be applied to the multi-agent workflow language as considered in [19] for analyzing noninterference in presence of declassification and agent coalitions. There, transformations are presented to encode noninterference properties as invariants of the *self-composition* of the given workflow [3,17]. At least for the case of *stubborn agents* [11], i.e., agents who do not participate in adversarial coalitions, the given transformation preserves both guardedness and the stratification. The same also holds true if the size of adversarial coalitions is bounded. For these cases, our novel decidability results therefore translate into decidability of noninterence.

References

1. Ackermann, W.: Untersuchungen über das Eliminationsproblem der mathematischen Logik. Math. Ann. **110**, 390–413 (1935)
2. Ball, T., et al.: Vericon: towards verifying controller programs in software-defined networks. In: ACM Sig-plan Notices number 6, vol. 49, pp. 282–293. ACM (2014)
3. Barthe, G., Crespo, J.M., Kunz, C.: Product programs and relational program logics. J. Log. Algebraic Methods Program. **85**(5), 847–859 (2016). https://doi.org/10.1016/j.jlamp.2016.05.004

4. Berkovits, I., Lazić, M., Losa, G., Padon, O., Shoham, S.: Verification of threshold-based distributed algorithms by decomposition to decidable logics. In: Dillig, I., Tasiran, S. (eds.) CAV 2019. LNCS, vol. 11562, pp. 245–266. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-25543-5_15
5. Börger, E., Grädel, E., Gurevich, Y.: The Classical Decision Problem. Perspectives in Mathematical Logic. Springer, Heidelberg (1997)
6. Börger, E., Stärk, R.: History and survey of ASM research. In Abstract State Machines: A Method for High-Level System Design and Analysis, pp. 343–367. Springer, Heidelberg (2003). ISBN: 978-3-642-18216-7. https://doi.org/10.1007/978-3-642-18216-7_9
7. Böorger, E., Stäark, R.: Tool support for ASMs. In: Abstract State Machines: A Method for High-Level System Design and Analysis, pp. 313–342. Springer, Heidelberg (2003). ISBN: 978-3-642-18216-7, https://doi.org/10.1007/978-3-642-18216-7_8
8. de Moura, L., Bjørner, N.: Z3: an efficient SMT solver. In: Ramakrishnan, C.R., Rehof, J. (eds.) TACAS 2008. LNCS, vol. 4963, pp. 337–340. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78800-3_24
9. Feldman, Y.M.Y., Padon, O., Immerman, N., Sagiv, M., Shoham, S.: Bounded quantifier instantiation for checking inductive invariants. Logical Methods Comput. Sci. **15**, 3 (2019). https://doi.org/10.23638/LMCS-15(3:18)2019
10. Finkbeiner, B., Müller, C., Seidl, H., Zalinescu, E.: Verifying security policies in multi-agent work OWS with loops. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS 2017), pp. 633–645. IEEE (2017). https://doi.org/10.1145/3133956.3134080
11. Finkbeiner, B., Seidl, H., Müller, C.: Specifying and verifying secrecy in workflows with arbitrarily many agents. In: Artho, C., Legay, A., Peled, D. (eds.) ATVA 2016. LNCS, vol. 9938, pp. 157–173. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46520-3_11
12. Gabbay, D.M., Schmidt, R., Szalas, A.: Second Order Quantifier Elimination: Foundations. Computational Aspects and Applications, College Publications (2008)
13. Goguen, J.A., Meseguer, J.: Security policies and security models. In: IEEE Symposium on Security and Privacy, Oakland, CA, USA, April 26–28, 1982. IEEE Computer Society (1982). https://doi.org/10.1109/SP.1982.10014
14. Gurevich, Y.: Evolving algebras 1993: Lipari guide. arXiv preprint arXiv:1808.06255 (2018)
15. Karbyshev, A., Bjørner, N., Itzhaky, S., Rinetzky, N., Shoham, S.: Property-directed inference of universal invariants or proving their absence. J. ACM (JACM) **64**(1), 7 (2017)
16. Koenig, J.R., Padon, O., Immerman, N., Aiken, A.: [n. d.] Firstorder quantified separators. In: Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2020) (2020, to appear)
17. Kovács, M., Seidl, H., Finkbeiner, B.: Relational abstract interpretation for the verification of 2-hypersafety properties. In: Sadeghi, A.-R., Gligor, V.D., Yung, M. (eds.) 2013 ACM SIGSAC Conference on Computer and Communications Security, CCS 2013, Berlin, Germany, November 4–8, 2013, pp. 211–222. ACM (2013). https://doi.org/10.1145/2508859.2516721
18. McMillan, K.L., Padon, O.: Deductive verification in decidable fragments with ivy. In: Podelski, A. (ed.) SAS 2018. LNCS, vol. 11002, pp. 43–55. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99725-4_4

19. Müller, C., Seidl, H., Zalinescu, E.: Inductive invariants for noninterference in multi-agent work flows. In: 31st IEEE Computer Security Foundations Symposium, (CSF 2018), pp. 247–261. IEEE (2018). https://doi.org/10.1109/CSF.2018.00025
20. Padon, O., Immerman, N., Karbyshev, A., Lahav, O., Sagiv, M., Shoham, S.: Decentralizing SDN policies. In: ACM SIGPLAN Notices, vol. 50, no. 1, pp. 663–676. ACM (2015)
21. Padon, O., Immerman, N., Shoham, S., Karbyshev, A., Sagiv, M.: Decidability of inferring inductive invariants. In: Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016. ACM, 217–231 (2016). https://doi.org/10.1145/2837614.2837640
22. Padon, O., Losa, G., Sagiv, M., Shoham, S.: Paxos made EPR: decidable reasoning about distributed protocols. In: Proceedings of the ACM Programming Language, 1, OOPSLA, 108:1–108:31 (2017). https://doi.org/10.1145/3140568
23. Padon, O., McMillan, K.L., Panda, A., Sagiv, M., Shoham, S.: Ivy: safety verification by interactive generalization. ACM SIG- PLAN Notices **51**(6), 614–630 (2016)
24. Ranzato, F.: Decidability and synthesis of abstract inductive invariants. CoRR, abs/2004.03170. arXiv:2004.03170 (2020). https://arxiv.org/abs/2004.03170
25. Seidl, H., Müller, C., Finkbeiner, B.: How to win first-order safety games. In: Beyer, D., Zufferey, D. (eds.) VMCAI 2020. LNCS, vol. 11990, pp. 426–448. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-39322-9_20