UNIVERSITA' DEGLI STUDI DI MILANO

Dipartimento di Scienze dell'Informazione

RAPPORTO INTERNO N° RI 336-11

Automated Termination in Model Checking Modulo Theories (extended version)

Alessandro Carioni, Silvio Ghilardi, Silvio Ranise

Automated Termination in Model Checking Modulo Theories (extended version)

Alessandro Carioni[1] and Silvio Ghilardi[2] and Silvio Ranise[3]
[1]Dipartimento di Tecnologie dell'Informazione, Università degli Studi di Milano (Italy)
[2]Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano (Italy)
[3]FBK (Fondazione Bruno Kessler), Trento, (Italy)

November 25, 2011

Abstract

We use a declarative SMT-based approach to model-checking of infinite state systems to design a sufficient procedure for automatically establishing the termination of backward reachability by using well-quasi-orderings. Besides showing that our procedure succeeds in many instances of problems covered by general termination results, we argue that it could predict termination also on single problems outside the scope of applicability of such general results.

This Technical Report is the extended version of a paper published in the proceedings of the 5th International Workhop on Reachability Problems (RP '11).

1 Introduction

Infinite state model checking is nowadays a mature field and many successful attempts have been made to verify disparate problems, using various kinds of methodologies. Still, termination of search (both forward and backward) is a major problem and it is sometimes difficult to predict *in advance* whether a given problem will be solved in a finite amount of time or some source of divergence is hidden somewhere and appropriate techniques (such as acceleration or abstraction) should be employed in order to make the search terminating. In the literature, several results for the termination of backward reachability are known covering entire classes of infinite state systems [2, 5, 6, 10, 12]. In most cases, following the seminal work in [1], these results are based on the use of well-quasi-orders on configurations, which are (finite) symbolic representations of infinite sets of backward reachable states. Thus, their applicability crucially rely on the human ability to reformulate a given specification so that it fits in one of the classes of systems for which termination is guaranteed.

In this paper, we propose an *automated* technique capable (when successful) of predicting termination from the static analysis of a given verification problem that is amenable to backward reachability. We develop our ideas in the model checking modulo theories framework [13, 17] where array-based (guarded assignment) transition systems are used to symbolically specify a wide range of systems by using certain classes of first-order formulae and background theories. There are two main ingredients. The former is that of a *wqo-theory* W, which is the declarative counterpart of a well-quasi-order on configurations used in the main arguments for termination in [1]. The second ingredient is the standard notion in first-order logic textbooks (see, e.g., [11]) of *syntactic interpretation*. Then, we cook these two ingredients together to design a method for establishing the termination of backward reachability on a given verification problem by checking for the *existence* of a syntactic translation from W, satisfying certain conditions. Such conditions refer to a search space *restricted to formulae describing small models* and the possibility of such restriction is indeed the essential content of our main result (Theorem 4.6 below). Interestingly, the conditions of Theorem 4.6 can all be checked trough proof obligations that can be efficiently discharged by using SMT solving techniques. We shall turn to an informal discussion on our main issue at the beginning of Section 4.

2 Preliminaries

We assume the usual syntactic (e.g., signature, variable, term, ground term, atom, literal, formula, and sentence) and semantic (e.g., structure, sub-structure, embedding, assignment, truth, and validity) notions of many-sorted first-order logic (see, e.g., [9, 11]). The equality symbol $=$ is included in all signatures considered below. We use $L, M, \dots$ for literals and $\phi, \psi, \dots$ for formulae. A signature is *relational* iff it does not contain function symbols and it is *quasi-relational* iff the only function symbols it contains are constants. A formula is *open* (or *quantifier-free*) iff it does not contain quantifiers; it is *universal* (resp. *existential*) iff it is obtained from an open formula by prefixing it a finite sequence of universal (resp. existential) quantifiers. If $\phi(\underline{x})$ is a formula with free variables included in the tuple $\underline{x} = x_1, \dots, x_n$ and $\underline{a} = a_1, \dots, a_n$ is a (sort-conforming) tuple of elements of the support $|\mathcal{M}|$ of a structure $\mathcal{M}$, we write $\mathcal{M} \models \phi(\underline{a})$ to denote that $\phi(\underline{x})$ is valid in $\mathcal{M}$ under the assignment $\{x_1 \mapsto a_1, \dots, x_n \mapsto a_n\}$.

SMT. Following [21], a *theory* T is a pair $(\Sigma, \mathcal{C})$, where Σ is a signature and $\mathcal{C}$ is a class of Σ-structures; the structures in $\mathcal{C}$ are the *models* of T. Below, let $T = (\Sigma, \mathcal{C})$. A Σ-formula ϕ is *T-satisfiable* if there exists a Σ-structure $\mathcal{M}$ in $\mathcal{C}$ such that ϕ is true in $\mathcal{M}$ under a suitable assignment to the free variables of ϕ (in symbols, $\mathcal{M} \models \phi$); it is *T-valid* (in symbols, $T \models \varphi$) if its negation is T-unsatisfiable. Two formulae φ_1 and φ_2 are *T-equivalent* if $\varphi_1 \leftrightarrow \varphi_2$ is *T-valid.* The *quantifier-free satisfiability modulo the theory T ($SMT(T)$) problem* amounts to establishing the T-satisfiability of quantifier-free Σ-formulae. A theory is said to be *syntactically specified* if we are given a set of Σ-sentences (called the *axioms* of T): in this case, the class $\mathcal{C}$ of the models of T is formed by the Σ-structures in which all the axioms of T are true. A theory T is *universal* iff it is syntactically specified and its axioms are universal sentences.

Diagrams. Given a (finite, in our applications) Σ-structure $\mathcal{M}$, take a free variable x_a for every a in the support of $\mathcal{M}$ and call $\underline{x}$ the set of all x_a (varying a). The *Σ-diagram $\delta_{\mathcal{M}}$ of $\mathcal{M}$* [9] is the set of all Σ-literals $L(\underline{x})$ such that $\mathcal{M}, \mathtt{a} \models L$, where $\mathtt{a}$ is the assignment mapping x_a to a. By abuse of notation, we shall confuse the variable x_a with the element a. Intuitively, we can view $\delta_{\mathcal{M}}$ as a sort of 'multiplication table' of the structure $\mathcal{M}$. Notice that the diagram of a finite structure is also finite and can be seen as the formula obtained by the conjunction of its literals.

Interpretations. Informally, an interpretation $(-)^*$ of a Σ-theory T into a Σ'-theory T' is a mapping from the expressions of T to the expressions of T' which preserves the validity of sentences. We will consider a special class of interpretations, generalizations of our definition exist—see, e.g., [11]—but we do not need them here. Formally, $(-)^*$ is a mapping associating (i) a sort S^* of Σ' with each sort S of Σ, (ii) a Σ'-formula $R^*(x_1, \dots, x_n)$ with each predicate symbol R of Σ in such a way that the variables $x_1, \dots, x_n$ occurring free in R^* match the translations of the arity sorts of R (implicitly we assume that identity of sort S is translated into identity of sort S^*), and (iii) a Σ'-term $f^*(x_1, \dots, x_n)$ with each function symbol f of Σ (with the same condition on $x_1, \dots, x_n$ as for predicate symbols). Then, $(-)^*$ can be extended inductively to formulae in the obvious way: $(R(t_1, \dots, t_n))^* = R^*(t_1^*/x_1, \dots, t_n^*/x_n)$ for atomic formulae and $(A \wedge B)^* = A^* \wedge B^*, (\neg A)^* = \neg A^*, (\forall x A)^* = \forall x A^*$ for non-atomic formulae. The last requirement that $(-)^*$ is supposed to satisfy is the following: (iv) for every sentence ϕ in the signature of T, if $T \models \phi$ then $T' \models \phi^*$. Notice that, if T is specified syntactically, it is sufficient to check (iv) only for the axioms of T. In this paper, we shall limit ourselves to *quantifier-free* translations, i.e. to translations in which the formulae R^* mentioned in (ii) above are quantifier-free.

3 Array-based systems and backward reachability

Array-based transition systems [13, 15, 17, 18] have been proved useful for the verification of several classes of infinite state systems, such as broadcast protocols, lossy channel systems, timed networks, parametric and distributed systems. The Model Checker Modulo Theories (MCMT) tool [19] implements symbolic backward reachability for array-based systems (its executable, several benchmark problems, the documentation, and related papers can be downloaded at `http://homes.dsi.unimi.it/~ghilardi/mcmt`). The state variables of array-based systems are arrays "connecting" the theories T_I and T_E: the former describes the topology and the latter the data structures of the system. We fix for the whole paper the following conventions:

(T1) $T_I = (\Sigma_I, \mathcal{C}_I)$ is a relational mono-sorted theory whose unique sort is named $\mathtt{INDEX}$), its $SMT(T_I)$-problem is decidable, and $\mathcal{C}_I$ is closed under substructures (meaning that $\mathcal{C}_I$ contains every substructure of any $\mathcal{M} \in \mathcal{C}_I$);

(T2) T_E is a multi-sorted theory whose sorts are $\mathtt{ELEM}_1, ..., \mathtt{ELEM}_S$ for some $S \geq 1$ and its $SMT(T_E)$-problem is decidable;

(T3) A_I^E is a compound theory, intended to describe 'arrays with indexes in T_I and elements in T_E.' Formally, the signature of A_I^E contains the sort symbols of $\Sigma_I \cup \Sigma_E$, together with a new sort symbol $\mathtt{ARRAY}_\ell$ for each sort $\mathtt{ELEM}_\ell$ of Σ_E; it contains also all the function and predicate symbols in $\Sigma_I \cup \Sigma_E$ together with a new function symbol $\_[\_]_\ell \ : \ \mathtt{ARRAY}_\ell, \mathtt{INDEX} \longrightarrow \mathtt{ELEM}_\ell$ for each sort $\mathtt{ELEM}_\ell$ of Σ_E. The models $\mathcal{M}$ of A_I^E are the structures whose Σ_I- and Σ_E-reducts are models of T_I and T_E, respectively; the sort $\mathtt{ARRAY}_\ell$ is interpreted as the set of total functions $\mathtt{INDEX}^\mathcal{M} \longrightarrow \mathtt{ELEM}_\ell^\mathcal{M}$ and the symbol $\_[\_]_\ell$ as function application.

So, $a[i]_\ell$ denotes the element of sort $\mathtt{ELEM}_\ell$ stored in the array a of sort $\mathtt{ARRAY}_\ell$ at index i; the subscript ℓ is dropped whenever it is clear from the context.

Definition 3.1. An *array-based (transition) system (for (T_I, T_E))* is a triple $\mathcal{S} = (a, I, \tau)$ where (i) $a = a_1, \ldots, a_s$ is a tuple of free constants of array sorts (these are to be thought as *state variables* storing data of sorts $\mathtt{ELEM}_1, \ldots, \mathtt{ELEM}_s$, respectively); (ii) $I(a)$ is the *initial* formula; (iii) $\tau(a, a')$ is the *transition* formula, where a' contains the renamed copies of the variables in a. The formula I is assumed to be a $\forall^I$-formula and the formula τ is a disjunction $\bigvee_{h=1}^r \tau_h$ of guarded assignments in functional form.

To give the definition of a $\forall^I$-formula and of a guarded assignment in functional form, we preliminarily introduce the following notational convention and definitions. Below, d, e range over variables of a sort $\mathtt{ELEM}_\ell$ of Σ_E, $i, j, k, z, \ldots$ over variables of sort $\mathtt{INDEX}$. An underlined variable name abbreviates a tuple of variables of unspecified (but finite) length and, if $\underline{i} := i_1, \ldots, i_n$, the notation $a[\underline{i}]$ abbreviates the $s{*}n$-tuple of terms $a_1[i_1], \ldots, a_s[i_1], \ldots, a_1[i_n], \ldots, a_s[i_n]$. Possibly sub/super-scripted expressions of the form $\phi(\underline{i}, \underline{e}), \psi(\underline{i}, \underline{e})$ denote *quantifier-free* $(\Sigma_I \cup$

Σ_E)*-formulae in which at most the variables* $\underline{i} \cup \underline{e}$ *occur.* Also, $\phi(\underline{i}, \underline{t}/\underline{e})$ (or simply $\phi(\underline{i}, \underline{t})$) abbreviates the substitution of the Σ-terms $\underline{t}$ for the variables $\underline{e}$. Thus, for instance, $\phi(\underline{i}, a[\underline{i}])$ denotes the formula obtained by replacing $\underline{e}$ with $a[\underline{i}]$ in the quantifier-free formula $\phi(\underline{i}, \underline{e})$.

Given a theory T (in our case, T will be A_I^E), a *T-partition* is a finite set $C_1(\underline{x}), \ldots, C_n(\underline{x})$ of quantifier-free formulae (with free variables contained in the tuple $\underline{x}$) such that $T \models \forall \underline{x} \bigvee_{i=1}^n C_i(\underline{x})$ and $T \models \bigwedge_{i \neq j} \forall \underline{x} \neg (C_i(\underline{x}) \wedge C_j(\underline{x}))$. The formulae $C_1, \ldots, C_k$ are called the *components* of the T-partition. A *case-definable extension* $T' = (\Sigma', \mathcal{C}')$ of a theory $T = (\Sigma, \mathcal{C})$ is obtained from T by applying (finitely many times) the following procedure: (i) take a T-partition $C_1(\underline{x}), \ldots, C_n(\underline{x})$ together with Σ-terms $t_1(\underline{x}), \ldots, t_n(\underline{x})$; (ii) let Σ' be $\Sigma \cup \{F\}$, where F is a "fresh" function symbol (i.e. $F \not\in \Sigma$) whose arity is equal to the length of $\underline{x}$; (iii) take as $\mathcal{C}'$ the class of Σ'-structures $\mathcal{M}$ whose Σ-reduct is a model of T and such that $\mathcal{M} \models \bigwedge_{i=1}^n \forall \underline{x} \; (C_i(\underline{x}) \to F(\underline{x}) = t_i(\underline{x}))$. Thus a case-definable extension T' of a theory T contains finitely many additional function symbols, called case-defined functions. By abuse of notation, below, we shall identify T with its case-definable extensions T'.

A formula $\forall \underline{i}.\phi(\underline{i}, a[\underline{i}])$ is a *$\forall^I$-formula*, one of the form $\exists \underline{i}.\phi(\underline{i}, a[\underline{i}])$ is an *$\exists^I$-formula*, and a sentence $\exists a \; \exists \underline{i} \; \forall \underline{j} \; \psi(\underline{i}, \underline{j}, a[\underline{i}], a[\underline{j}])$ is an *$\exists^{A,I}\forall^I$-sentence.* A *guarded assignment in functional form* is a formula of the form

$$\exists e \exists \underline{k} \, (\phi_L(e, \underline{k}, a[\underline{k}]) \wedge a' = \lambda j.F(e, \underline{k}, a[\underline{k}], j, a[j])) \tag{1}$$

where: (i) $F = F_1, \ldots, F_s$ is a tuple of case-defined functions; (ii) the existentially quantified data variable e ranges over a sort $\mathtt{ELEM}_\ell$ such that T_E admits quantifier elimination with respect to quantified variables of sort $\mathtt{ELEM}_\ell$.

Given an array-based system $\mathcal{S} = (a, I, \tau)$ and an $\exists^I$-formula $U(a)$ describing a set of *unsafe* states (also called *error* states), the symbolic backward reachability procedure iteratively computes the set of backward reachable states $BR(a)$ as follows. (Below, we give a very high-level description of the symbolic backward reachability procedure implemented in MCMT. For a description of the techniques and heuristics used to make the procedure effective in practice, the reader is pointed to [15–18]. In particular, [17] reports an extensive experimental evaluation of the tool.) Preliminarily, define (for $n \geq 0$) the *n-pre-image* of a formula $K(a)$ as $Pre^0(\tau, K) := K$ and $Pre^{n+1}(\tau, K) := Pre(\tau, Pre^n(\tau, K))$, where $Pre(\tau, K) := \exists a'.(\tau(a, a') \wedge K(a'))$. Intuitively, $Pre^n(\tau, U)$ describes the set of backward reachable states in $n \geq 0$ steps. At the n-th iteration, the *backward reachability procedure* computes the formula $BR^n(\tau, U) := \bigvee_{i=0}^n Pre^i(\tau, U)$ representing the set of states which are backward reachable from the states in U with at most n steps. While computing $BR^n(\tau, U)$, the procedure also checks whether the system is unsafe by establishing if the formula $I \wedge Pre^n(\tau, U)$ is A_I^E-satisfiable (*safety test*) or whether a fix-point has been reached by checking if $(BR^n(\tau, U) \to BR^{n-1}(\tau, U))$ is A_I^E-valid or, equivalently, if the formula $BR^n(\tau, U) \wedge \neg BR^{n-1}(\tau, U)$ is A_I^E-unsatisfiable (*fix-point test*).

To mechanize this procedure, it is mandatory to identify a class of formulae for representing sets of backward reachable states which is closed under pre-image computation and such that the safety and fix-point checks are decidable. Using $\exists^I$-formulae for representing unsafe states, this is indeed the case as stated in the following theorem, a corollary of more general results in [13, 17].

Theorem 3.2. *Let $\mathcal{S} = (a, I, \tau)$ be an array-based system; we have that*

(RF1) *if K is an $\exists^I$-formula, then $Pre(\tau, K)$ is equivalent to an (effectively computable) $\exists^I$-formula;*

(RF2) *if the set U of unsafe states is represented by an $\exists^I$-formula, then the A_I^E-satisfiability checks for safety and fix-point of the backward reachability procedure are effective.*

As shown in [7, 8, 13, 17, 18], this theorem allows the automated verification of reachability properties for several classes of systems (e.g., parameterised systems, timed networks, or fault-tolerant algorithms). For several of these problems, a (declarative reformulation of) the *approximate model technique* (see, e.g., [4]) is required as explained in [14]. **(RF2)** is a special case of the decidability of the A_I^E-satisfiability problem for $\exists^{A,I}\forall^I$-sentences in [17]. Assumptions (**T1**) on T_I are essential for **(RF2)** because, if they are dropped, undecidability of $\exists^{A,I}\forall^I$-sentences arises [17]. The proof of **(RF2)** in [17] consists of a decision procedure integrating quantifier-free SMT solving and quantifier instantiation. Powerful heuristics [16] are also crucial for implementation.

An important refinement of the backward reachability procedure above is to exploit invariants whenever they are available. An invariant $J(a)$ for the array-based system $\mathcal{S} = (a, I, \tau)$ is a $\forall^I$-formula such that (a) $A_I^E \models I(a) \to J(a)$ and (b) $A_I^E \models Pre(\tau, J) \to J(a)$. The requirement that $J(a)$ is a $\forall^I$-formula allows us to reduce conditions (a) and (b) to the A_I^E-satisfiability of $\exists^{A,I}\forall^I$-sentences, which is decidable because of (**T1**). Techniques for invariant synthesis are discussed in [17] and are also implemented in MCMT. Whenever an invariant J is known, we can replace A_I^E with $A_I^E \cup \{J\}$ in our satisfiability tests (e.g. for fix-point in the backward reachability procedure). The presence of invariants in such tests is often crucial either to greatly speed up the performances of MCMT or to obtain termination (see again [17] for details).

3.1 Closure under pre-image computation

The proof of **(RF1)** (Theorem 3.2) in, e.g., [13, 17] consists of simple logical manipulations (this is a distinguishing feature of our approach). Here, we briefly discuss a variant of such proofs, whose details are needed to state the main result of this paper (see Theorem 4.6 below).

Let $\mathcal{S} = (a, I, \tau)$ be an array-based system, where τ is a finite disjunction of formulae $\tau_1, ..., \tau_r$ of the form (1). We consider only $Pre(\tau_h, K)$ since $Pre(\tau, K)$ is easly seen to be equivalent to the disjunction of $Pre(\tau_1, K), ..., Pre(\tau_r, K)$. Let us now focus on the definition of $F = F_1, ..., F_s$ in (1). Without loss of generality (since partitions admit common refinements), we assume that the A_I^E-partition $\{C_1(e, \underline{k}, a[\underline{k}], j, a[j]), \ldots, C_m(e, \underline{k}, a[\underline{k}], j, a[j])\}$ is the same for each F_l ($l \in \{1, ..., s\}$). Thus, each case-defined function F_l can be written as

$$F_l(j, a[j]) \quad := \quad \texttt{case of}\{C_1(j, a[j]) : t_{l1}(j, a[j]); \; \cdots \; C_m(j, a[j]) : t_{lm}(j, a[j])\}, \qquad (2)$$

for $l = 1, \ldots, s$. (According to the definition of case-definable extension, the logical reading of the `case of` construct is the conjunction of the formulae $\bigwedge_{z=1}^{m} \forall \underline{j}.(C_z(j, a[j]) \to F_l(j, a[j]) = t_{lz})$ for each $l = 1, ..., s$.) Notice that $F_l, C_1, \ldots, C_m, t_{l1}, \ldots, t_{lm}$ depend not only on $j, a[j]$ but also on $e, \underline{k}, a[\underline{k}]$; to simplify notation, we omit these dependences in the rest of this paper. If $K(a)$ is the $\exists^I$-formula $\exists \underline{i}\, \psi(\underline{i}, a[\underline{i}])$, then $Pre(\tau_h, K)$ is logically equivalent to the formula obtained from

$$\phi_L \wedge \psi(\underline{i}, F[\underline{i}]) \qquad (3)$$

by prefixing it with the existential quantifiers $\exists \underline{i}\, \exists \underline{k}\, \exists e$; here the notation $F[\underline{i}]$ abbreviates the $(n{*}s)$-tuple of terms $F_l(i_z, a[i_z])$, varying $l = 1, \ldots, s$ and $z = 1, \ldots, n$ when $\underline{i} = i_1, \ldots, i_n$. We can further manipulate the formula (3) in order to eliminate the defined symbols $F_1, ..., F_s$. To do this, we consider the functions $f : \underline{i} \to \{1, \ldots, m\}$ and rewrite the formula (3) as the disjunction (varying f) of the formulae

$$\begin{aligned} \tau_h[\psi, f] := \quad & \phi_L \wedge C_{f(i_1)}(i_1, a[i_1]) \wedge \cdots \wedge C_{f(i_n)}(i_n, a[i_n]) \wedge \\ & \wedge \psi(\underline{i}, t_{1f(i_1)}(i_1, a[i_1]), \ldots, t_{sf(i_n)}(i_n, a[i_n])) \; . \end{aligned}$$

At this point, it is clear that $Pre(\tau_h, K)$ is equivalent to $\bigvee_f \exists \underline{i}\, \exists \underline{k}\, \exists e\, \tau_h[\psi, f]$, where the functions f indexing the disjunction will be called *case-marking functions* and their purpose is to mark each index in $\underline{i}$ with the case that formally applies to it. This concludes the proof of (**RF1**).

4 Wqo-theories, QE-degrees, and Termination

The assumptions of Theorem 3.2 guarantee that the backward reachability procedure described in the previous section can be mechanized but they are not sufficient to guarantee termination. This is because the symbolic representation of pre-images (in our case, $\exists^I$-formulae) may not be expressive enough to represent a fix-point of the set of backward reachable states. Termination can be achieved only under additional assumptions. The classical (non declarative) method for obtaining termination (see, e.g., [1, 5, 6]) consists in endowing the states of the system with a preorder relation $\preceq$ and in assuming that (1) pre-images are monotonic w.r.t. $\preceq$ and (2) $\preceq$ is a well-quasi-ordering (wqo). Now, (1) implies that the pre-image of an upward-closed (w.r.t. $\preceq$) set is still upward-closed and (2) implies that every upward-closed set can be characterized by a finite set of minimal (w.r.t. $\preceq$) elements. Thus, starting from an upward-closed set U of states, the iterative computation of the backward reachable configurations from U necessarily terminates because the fixpoint is upward closed and hence its minimal elements are rechable in finitely many steps. Obviously, this requires that relevant upward-closed sets can be effectively represented and manipulated. Our goal here is to recast this argument in our declarative framework underlying the backward reachability procedure of Section 3. Roughly, our plan is as follows: without loss of generality (see [17], Section 4), states of the system can be identified with the values assigned to the array constants a in a model $\mathcal{M}$ having finitely many generators of sort `INDEX`. Since we represent backward reachable states by $\exists^I$-formulae, we replace $\preceq$ by the embeddability relation between such finitely generated models. The fact that existential formulae are preserved by super-structures guarantees that they describe upward closed sets of states. However, it would be too strong to require embeddability among finitely generated models to be a wqo: instead, we make an abstraction of the system, through a wqo-theory W and a syntactic translation into A_I^E. This will replace embeddability between finitely generated models of A_I^E by *embeddability between the abstract states*, i.e. between the finitely generated models of W. The definition of a wqo theory just says that embeddability between such abstract states is a wqo. The key step of our plan consists of checking whether every pre-image P_i computed by the backward reachability procedure is a translation of an existential formula β_i in W for $i \geq 0$ where $P_0 := U$. The sequence $\beta_0, \beta_1, ...$ is finite because it describes increasingly larger upsets of a wqo, hence the sequence $P_0, P_1, ...$ must be finite too. In fact, if there exists $b \geq 1$ such that β_b is a fix-point, i.e. $W \models \beta_b \rightarrow \beta_{b-1}$, by using the syntactic translation we must also have that $A_I^E \models P_b \rightarrow P_{b-1}$ where P_b and P_{b-1} are the translations of β_b and β_{b-1}, respectively. Hence P_b is a fixed point of our backward reachability procedure. Thus we must find a condition that guarantees that the backward reachability procedure generates only $\exists^I$-formulae which are translations of existential formulae of W: to this aim, we reduce the general case to finitely many cases involving formulae of the kind $\exists e\, \exists \underline{i}\, \exists \underline{k}\, \tau_h[\psi, f]$, *where* ψ *is the translation of the diagram of a "small" model of* W. Finally, since the elimination of the existentially quantified data variable e is required, we need also assumptions on the quantifier elimination algorithm.

Wqo-theories. A wqo $(P, \leq)$ is a set P endowed with a binary reflexive and transitive relation $\leq$ such that for every infinite sequence $p_1, p_2, \ldots$ of elements from P there are $i < j$

such that $p_i \leq p_j$.

Definition 4.1. A wqo-theory is a universal theory whose finitely generated models[1] are a well-quasi-order with respect to the embeddability relation.

Simple examples of wqo-theories can be obtained by taking vector spaces on a fixed field, torsion-free abelian groups, etc. The reason why we get a wqo in these cases is that an embedding always exists whenever the dimension is lower. Examples of wqo-theories which are more relevant to this paper can be obtained by re-interpreting declaratively some special cases of Kruskal theorem or Higman lemma, as sketched in the following example.

Example 4.2. Consider a signature contaning one sort, finitely many 0-ary and 1-ary predicates and a single binary predicate $\leq$ (besides equality). We get a wqo theory W by Higmann lemma (see Appendix A for details) if we syntactically specify W through the following set of axioms

$$\forall x\,(x \leq x), \quad \forall x, y, z\,(x \leq y \wedge y \leq z \to x \leq z), \text{ and } \forall x, y\,(x \leq y \vee y \leq x)$$

(the axioms say that $\leq$ is to be interpreted as a total pre-order relation).

QE-degree. A theory T admits *quantifier elimination* (relative to a sort S) iff for every formula $\varphi(\underline{x})$ containing only quantified variables of sort S, there exists a quantifier-free formula $\varphi'(\underline{x})$ such that $T \models \forall \underline{x}(\varphi(\underline{x}) \leftrightarrow \varphi'(\underline{x}))$.

A set $\mathcal{P}$ of Σ-predicates is said to be T*-representative* for a Σ-theory T iff for every Σ-literal $L(\underline{x})$ one can compute a formula $\psi(\underline{x})$ which is a positive combination (i.e. a disjunction of conjunctions) of atoms whose root predicate symbol is in $\mathcal{P}$ such that $T \models \forall \underline{x}\,(L \leftrightarrow \psi)$. The atoms whose root symbol is a predicate in $\mathcal{P}$ are called T*-representative literals.* The set of T-representative literals is denoted by $\mathcal{L}_{\mathcal{P}}$. Notice that T-representative literals are closed under taking substitutions of terms for variables, by definition.

Definition 4.3. Let T be a theory eliminating quantifiers for a sort S. We say that T has QE-degree N with respect to a set of representative predicates $\mathcal{P}$ iff for every finite family $\{L_i\}_{i \in I}$ of literals from $\mathcal{L}_{\mathcal{P}}$, the formula

$$\exists x\,(\bigwedge_{i \in I} L_i) \leftrightarrow \bigwedge_{I_0 \subseteq_N I} \exists x\,(\bigwedge_{i \in I_0} L_i) \tag{4}$$

is T-valid (here the variable x is of sort S and $I_0 \subseteq_N I$ means that I_0 is a subset of I having cardinality at most N).

[1] Recall that a model $\mathcal{M}$ is finitely generated iff there is a finite subset X of the support $|\mathcal{M}|$ of $\mathcal{M}$ such that for every $c \in |\mathcal{M}|$, there are $\underline{b} \subseteq X$ and a term $t(\underline{x})$ such that $\mathcal{M} \models t(\underline{b}) = c$.

Notice that the left-to-right implication in (4) is universally valid. If we allow N to be ∞, then it is clear that any quantifier eliminating theory has a QE-degree with respect to any representative set of predicates $\mathcal{P}$. In this paper, we are interested only in the cases in which the QE-degree is finite.

Example 4.4. Real linear arithmetic is the the set of sentences in the signature $\{0, 1, -, +, <, =\}$ which are true in $\mathbb{R}$ (endowed with the natural interpretation). We shall adopt the well-known Fourier-Motzkin quantifier elimination procedure for this theory. For simplicity, suppose we also have a constant for each rational number in the language; we take as representative literals the positive literals, i.e. those of the kind $t = u$ or $t < u$ (thus $\{=, <\}$ is the set of representative predicates). We show that *we have QE-degree equal to 2*. In order to eliminate the variable x from a conjunction of a finite set of representative literals, we first rewrite such literals in such a way that they are of the following four kinds: (i) $x = t$; (ii) $x < u$; (iii) $v < x$; (iv) literals not involving x (here t, u, v are linear polynomials not containing x). In case there are literals of the kind (i), a simple replacement eliminates x (thus formula (4) trivially holds with $N = 2$). In case there are no literals of the kind (i), the elimination procedure produces as output the conjunction of all literals of the kind (iv), together with the literals of the kind $v < u$: the latter are equivalent to $\exists x\,(x < u \wedge v < x)$ and this shows that formula (4) holds with $N = 2$. It can be easily shown that the QE-degree is still equal to 2 if we add $\leq$ to the language and to set of representative predicates, but the same is not true anymore if we add $\neq$.[2]

Example 4.5. Real (resp. integer) difference logic is the the set of sentences in the signature $\{0, \mathit{succ}, \mathit{pred}, <, \leq, =\}$ which are true in $\mathbb{R}$ (resp. $\mathbb{Z}$), where $\mathit{succ}, \mathit{pred}$ are interpreted as the operation of adding/subtracting 1. Notice that atoms of this theory can be rewritten as $x - y \bowtie n$, where $n \in \mathbb{Z}$, $\bowtie \in \{=, <, \leq\}$ and x, y can possibly be 0. The argument in the previous example can be adapted to the present cases, thus yieldying again QE-degree 2 (in the integer case, the predicate $<$ can be dropped from the set of representative predicates, because $x - y < n$ can be replaced by $x - y \leq n - 1$).

We remark that Definition 4.3 does not allow negative literals to be T-representatives. However, there is an obvious way to circumvent this limitation (when needed) by expanding the signature in an inessential way. For instance, if we want negated equations to be T-representative, it is sufficient to expand the signature with a new binary predicate Neq, to let the formula $\forall x \forall y (Neq(x, y) \leftrightarrow x \neq y)$ be T-valid (e.g., by adding it to the axioms of T), and then to include Neq into the set $\mathcal{P}$ of representative predicates. Because of this, we

[2] In fact, if you include $\neq$ among representative predicates, formula (4) does not hold anymore. To get a counterexample, compute both sides of (4) in the case of $\exists x\,(x \neq b \wedge x > d \wedge x < e)$.

prefer to speak of 'T-representative literals' rather than of 'T-representative atoms'. Finally, notice that, in a multi-sorted context, we might be interested in quantifier elimination over just one sort (e.g., the sort representing time, in timed networks); in this case, it is convenient to take all predicates not involving such sort *and their negations* as representative, by using the above trick.

4.1 Automated termination

Let us fix from now on a wqo theory W and an array-based system $\mathcal{S} = (a, I, \tau)$ (built up over theories T_I, T_E). We make the following extra assumptions:

(E1) T_E has QE-degree N for all sorts occurring in τ as sorts of an existentially quantified data variable (this is the variable e in (1));

(E2) for each disjunct of the form (1) in τ, ϕ_L and the partition components in F are conjunctions of representative literals;

(E3) W has a finite relational signature Σ_W with unique sort S and there is a syntactic interpretation $(-)^*$ from W into A_I^E such that $S^* = \mathtt{INDEX}$.

The fact that a translation map $(-)^*$ is a syntactic interpretation from W into A_I^E is decidable since the axioms for W are universal and their translations modulo A_I^E generate $\exists^{A,I}\forall^I$-sentences, whose satisfiability is decidable. As a consequence, if the signature of A_I^E is finite (which is always the case in practical examples), the syntactic interpretations $(-)^*$ can be *enumerated*. In theory, this guarantees the possibility to find the right translation (if it exists) for the termination argument of Theorem 4.6 below to work, relatively to the wqo W, the array-based system $\mathcal{S}$, and the set of unsafe states U under consideration. In practice, however, the search for the right translation could be driven by user provided hints.

We use $\alpha(\underline{i}), \beta(\underline{i}), \dots$ to denote quantifier-free Σ_W-formulae in which at most the variables $\underline{i}$ occur. We say that an $\exists^I$-formula $\exists \underline{i}\ \psi(\underline{i}, a[\underline{i}])$ is a *translation* iff ψ is equivalent (modulo A_I^E) to a formula α^* for some $\alpha(\underline{i})$. Being a translation is clearly a decidable notion: this is because the signature of W is finite and relational, so the search space for the suitable $\alpha(\underline{i})$ is finite (the required validity tests fall within the decidability result for $\exists^{A,I}\forall^I$-sentences).

The following Theorem (whose proof is delayed to Appendix B) contains the main result of the paper:

Theorem 4.6. *Assume (**E1**)-(**E3**). Suppose that the $\exists^I$-formula $U(a)$ describing unsafe states is a translation and let M be the maximum arity of the predicate symbols in Σ_W. The backward reachability procedure terminates if the following conditions are satisfied by every finite model $\mathcal{M}$ of W ($\sharp\mathcal{M}$ indicates the cardinality of the support of $\mathcal{M}$):*

(i) *if $\sharp\mathcal{M} \leq M$, then the the translation $\delta^*_{\mathcal{M}}$ of the diagram of $\mathcal{M}$ is A_I^E-equivalent to a conjunction of representative literals $L_1^{\mathcal{M}} \wedge \cdots \wedge L_{k_{\mathcal{M}}}^{\mathcal{M}}$;*

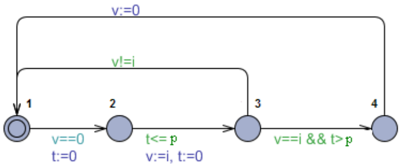
(ii) *if $\sharp\mathcal{M} \leq M$, then for every substructure $\mathcal{M}[\underline{i}_0]$ of $\mathcal{M}$, for every $r = 1, \ldots, k_{\mathcal{M}}$, we have that if $L_r^{\mathcal{M}}$ is of the kind $L(\underline{i}_0, a[\underline{i}_0])$ (i.e. if it mentions at most the elements $\underline{i}_0$ of the support of $\mathcal{M}[\underline{i}_0]$), then $A_I^E \models \delta^*_{\mathcal{M}[\underline{i}_0]} \to L_r^{\mathcal{M}}$;*

(iii) *if $\sharp\mathcal{M} \leq M * N$, then the formulae obtained after the elimination of the quantifier $\exists e$ from the formulae $\exists e\, \tau_h[\delta^*_{\mathcal{M}}, f]$ (varying τ_h among the disjuncts of τ and f under the suitable case-marking functions) are all translations.*

In practice, N and M have very small values, typically $N = M = 2$. Thus, only small models must be inspected in order to apply the theorem. When there is no existentially quantified data variable in all transitions (i.e. variable e does not occur neither in the ϕ_L's nor in the F's of (1)), the proof of Theorem 4.6 shows that we can assume $N = 1$ and condition (i) is not needed. In general, condition (i) might not be made fully automated; however, if it holds, an enumerative search can always effectively checks it. In practice, the situation is simple because the following straightforward procedure succeeds and is sufficient to guarantee (i). Consider $\delta^*_{\mathcal{M}}(\underline{i})$: if we replace in it all literals by their Boolean combinations of representative literals and put the result in disjunctive normal form, we get a formula of the form $\theta_1(\underline{i}, a[\underline{i}]) \vee \cdots \vee \theta_k(\underline{i}, a[\underline{i}])$ where the θ_j's are conjunctions of representative literals. Condition (i) of Theorem 4.6 is certainly guaranteed if $k = 1$ or if all θ_j but one are A_I^E-inconsistent. Condition (ii) is technical but usually holds trivially (we can figure that only pathological examples may violate it). The significant condition to verify is just (iii); it can be effectively checked because "being a translation" is decidable.

Theorem 4.6 covers termination of the backward reachability procedure described in Section 3 for several classes of systems including broadcast protocols [10, 12], lossy channels systems [5], timed networks with integer clocks [3], and timed networks with a single real clock [6].

The complexity of the procedure of Theorem 4.6 is hard to evaluate, because too many parameters contribute to it (not only M, N, but also the size of τ, of Σ_W, and of the translated atoms, the complexity of quantifier elimination, the number and the complexity of the involved SMT tests, etc.); however, it is arguable that we are well below the lower bounds known for deciding problems which are nevertheless covered by Theorem 4.6 (see e.g. [20]). Thus it might be convenient to apply our termination test before directly running backward search.



Legend: t is the local clock value, v is a shared variable, p is a positive parameter

Figure 1: Automaton for one process of the Fischer's protocol

5 Examples

In this Section we show how to apply Theorem 4.6 to concrete classes of problems. The application is rather trivial for cases like broadcast protocols and lossy channels systems, on the basis of their formalization as array-based systems (such formalization is just an exercise which is fully worked out in Section 7 of the extended version of [13]). We concentrate here to timed examples, where the application of Theorem 4.6 is less trivial. We first analyze a single benchmark and then show that Theorem 4.6 applies to the whole class of timed networks with a single real clock [6]. We shall also informally introduce some useful *heuristics* optimizing the automated verification of conditions (i)-(iii) of Theorem 4.6.

5.1 The Fischer protocol

The goal of the Fischer protocol[3] is to ensure mutual exclusion in a network of processes, using a clock and a shared variable v. Each process has a local clock and a control state variable ranging over $\{1, 2, 3, 4\}$. Each process is identified by a natural number > 0 and can read/update a shared variable whose values is either 0 or the index of one of the processes. A process wishing to enter the critical section 4 starts in 1. If $v = 0$, the process goes to 2 and resets its local clock. From 2, the process can go to state 3 if the clock is $< p$ time unit (where p is a positive parameter), sets v to its own index, and again resets its clock. From 3, the process can go to 4 if the clock is $> p$ time unit and v is still equal to the index of the process performing the transition. When exiting 4, the process sets v to 0. The set of unsafe states, i.e. those states violating the mutual exclusion property, can be characterized by the presence of at least two processes entering 4 at the same time. This specification is depicted in Figure 1. Before applying Theorem 4.6, we make few observations to simplify the technical development. As implemented in MCMT, the shared variable v is modelled as a

[3] Notice that this benchmark cannot be considered a special example of a timed networks with a single real clock in the sense of [6] (for more than one reason: there is a symbolic parameter p in the specification and an integer valued shared register is used). An ad hoc reformulation is needed for the Fischer protocol to fit the class of timed networks (see again [6]). Thus, this can be seen as a case where our results go beyond the classes of examples where termination is known.

constant array which is updated uniformly so that the invariant $\forall i\,\forall j\,(v[i] = v[j])$ holds. The invariant will be taken into consideration during backward search as explained in Section 3. For simplicity, below, we do not indicate the redundant dependency on the index i and write just v instead of $v[i]$. A second remark concerns *processes identifiers (id's)*, which are integers and the sort of integers is not the same as the sort `INDEX`. Formally, we view the id's as an array $id : \texttt{INDEX} \to \mathbb{Z}$ that is never updated; we implicitly add the conjunct $\forall i\,\forall j\,(id[i] = id[j] \to i = j)$ to the initial formula I and use this too as an invariant (this invariant just says that different processes have different id's); again, for simplicity however, below we write just i for $id[i]$. Since the id's are positive, we also implicitly add the conjunct $\forall i\,(id[i] > 0)$ to the initial formula and use it as an invariant. A third remark is about the use of *additional trivial invariants* that are a substantial part of the specification of the problem and do not capture any deep insight into the system. In our case, for local clocks, stored in the real-valued array t, we assume the invariant $\forall i\,(t[i] \geq 0)$ in order to specify that clock values are non-negative.

The theory A_I^E is composed of the theory of pure equality for T_I and the theory T_E is the union of three theories $T_{E_1}, T_{E_2}, T_{E_3}$, where T_{E_1} is linear real arithmetic, T_{E_2} is the theory of the single finite structure $Q = \{1, 2, 3, 4\}$ (the signature of T_{E_2} has just four constants and the identity predicate), and T_{E_3} is linear integer arithmetic. Since the specification contains the positive parameter p, the theory T_{E_1} is extended with a further constant p constrained by the axiom $p > 0$. Quantifier elimination is assumed only for T_{E_1} and we take as representative predicates all the predicates of T_{E_2}, T_{E_3} together with their negations; we pick just $<, \leq, =$ as representative predicates from T_{E_1} (we get QE-degree 2 with this choice, see Example 4.4). The following array-based system (a, I, τ) formalizes the Fischer protocol:

- a is a 4-tuple of array variables $\langle l, t, v, id \rangle$, whose target sorts are those of T_{E_1}, T_{E_2} and T_{E_3} (twice), respectively (l is the array of locations, t is the array of clocks, v is the shared register, and id is the array of the id's);

- the formula I is $\forall i\,(l[i] = 1 \wedge t[i] = 0 \wedge v = 0)$, which constrains the initial locations to be equal to 1 and that clocks and the shared variable to 0;

- τ is the disjunction of the time elapsing transition τ_1 and the discrete transitions $\tau_2, ..., \tau_6$ listed below above (identical updates of the array id are not displayed for the sake of conciseness):

$$
\begin{array}{lcl}
\tau_1 & := & \exists c. \Big(\; c > 0 \quad \wedge \quad v' = v \wedge l' = l \wedge t' = \lambda j.(t[j] + c) \; \Big) \\
\tau_2 & := & \exists i. \left(\begin{array}{l} l[i] = 1 \wedge v = 0 \;\; \wedge v' = v \wedge \\ l' = \lambda j.\ (\mathsf{if}\ j = i\ \mathsf{then}\ 2\ \mathsf{else}\ l[j]) \wedge \\ t' = \lambda j.\ (\mathsf{if}\ j = i\ \mathsf{then}\ 0\ \mathsf{else}\ t[j]) \end{array} \right) \\
\tau_3 & := & \exists i. \left(\begin{array}{l} l[i] = 2 \wedge t[i] \leq p \;\; \wedge v' = i \;\wedge \\ l' = \lambda j.\ (\mathsf{if}\ j = i\ \mathsf{then}\ 3\ \mathsf{else}\ l[j]) \wedge \\ t' = \lambda j.\ (\mathsf{if}\ j = i\ \mathsf{then}\ 0\ \mathsf{else}\ t[j]) \end{array} \right) \\
\tau_4 & := & \exists i. \left(\begin{array}{l} l[i] = 3 \wedge \;\; v \neq i \;\; \wedge v' = v \wedge t' = t \;\wedge \\ l' = \lambda j.\ (\mathsf{if}\ j = i\ \mathsf{then}\ 1\ \mathsf{else}\ l[j]) \end{array} \right) \\
\tau_5 & := & \exists i. \left(\begin{array}{l} l[i] = 3 \wedge \;\; v = i \;\; \wedge t[i] > p \;\; \wedge v' = v \wedge t' = t \;\wedge \\ l' = \lambda j.\ (\mathsf{if}\ j = i\ \mathsf{then}\ 4\ \mathsf{else}\ l[j]) \end{array} \right) \\
\tau_6 & := & \exists i. \left(\begin{array}{l} l[i] = 4 \;\; \wedge v' = 0 \;\; \wedge t' = t \;\wedge \\ l' = \lambda j.\ (\mathsf{if}\ j = i\ \mathsf{then}\ 1\ \mathsf{else}\ l[j]) \end{array} \right)
\end{array}
$$

This completes the array-based specification for the Fischer protocol. Finally, the $\exists^I$-formula U describing the set of unsafe states in which the mutual exclusion property for location 4 is violated is $\exists i_1 \, \exists i_2 \, (i_1 \neq i_2 \wedge l[i_1] = 4 \wedge l[i_2] = 4)$.

To apply Theorem 4.6, we use the theory W in Example 4.2, relatively to the set of predicates indicated below, together with their syntactic translations (recall that a unary predicate must be translated as a formula containing one free index variable, a binary predicate as a formula containing two free index variables, etc).

- The unary predicates $Q_1, \ldots, Q_4$ are translated as the formulae $l[i] = 1, \ldots, l[i] = 4$, respectively;
- the unary predicates $P_{=0}, P_{>0}, P_{<p}, P_{=p}, P_{>p}$ are translated as $t[i] = 0, t[i] > 0, t[i] < p, t[i] = p, t[i] > p$, respectively;
- the unary predicate F is translated as $v = i$;
- the 0-ary predicate f is translated as $v = 0$;[4]
- the binary predicate $\leq$ is translated as $t[i_1] \leq t[i_2]$.

[4] To be precise, since v is formally an array constant, we should use again a unary predicate f and translate it as $v[i] = 0$.

Clearly, the translations of the axioms of W (namely reflexivity, transitivity and linearity axioms for $\leq$) are valid modulo A_I^E. The unsafety formula is a translation because $i_1 \neq i_2 \wedge l[i_1] = 4 \wedge l[i_2] = 4$ is the translation of $i_1 \neq i_2 \wedge Q_4(i_1) \wedge Q_4(i_2)$. It remains to check conditions (i)-(ii)-(iii) of Theorem 4.6.

For (i), since we have $M = 2$, we need to inspect the translation of the diagrams of the models of W having at most two elements.

An example. Consider the model $\mathcal{M}$ having $\{i_1, i_2\}$ as support in which the interpretation of all unary predicates is empty, except for $Q_1, P_{>0}, P_{<p}, P_{=p}$ whose extensions are $\{i_1, i_2\}, \{i_1\}, \{i_1\}, \{i_2\}$, respectively; in addition we have that the interpretation of $\leq$ is $\{< i_1, i_2 >, < i_1, i_1 >, < i_2, i_2 >\}$ and that f is false in $\mathcal{M}$. To build the diagram, we must write down all ground atomic formulae and their negations which are true in $\mathcal{M}$ (in the language expanded with names i_1, i_2 for the elements from the support of $\mathcal{M}$); then, we must take the conjunction of these literals, translate this conjunction and see whether it is equivalent to a conjunction of representative literals modulo A_I^E. It is clear that most literals are redundant: for instance, after translation, $\neg Q_2(i_1)$ becomes subsumed by $Q_1(i_1)$ (in fact $Q_1(i_1)^*$ is $l[i_1] = 1$ and this entails $l[i_1] \neq 2$). Notice also that negative literals involving unary predicates different from F are all redundant after translation. In conclusion, the translation of the diagram of our $\mathcal{M}$ is equivalent to

$$\begin{aligned} & i_1 \neq i_2 \;\wedge\; l[i_1] = 1 \;\wedge\; l[i_2] = 1 \;\wedge\; t[i_1] > 0 \;\wedge \\ & \wedge\; t[i_1] < p \;\wedge\; t[i_2] = p \;\wedge\; v \neq i_1 \;\wedge\; v \neq i_2 \;\wedge\; v \neq 0 \end{aligned}$$

(notice that $i_1 \leq i_2$ and $i_2 \not\leq i_1$ are also true in $\mathcal{M}$, but their translations are subsumed). This is a conjunction of representative literals (recall that all literals from T_{E_2} and T_{E_3} are representative, including negative ones).

The number of the models we have to consider (disregarding those whose diagram translation is inconsistent) is 2^{12}: the task of analysing them is not infeasible, especially with an automatic support, however the following heuristics makes it much easier. Any diagram we must examine is a conjunction of the following kinds of literals

$$\begin{gathered} k \neq k',\;\; Q_i(k),\;\; \neg Q_i(k),\;\; P_{*0}(k),\;\; \neg P_{*0}(k),\;\; f,\;\; \neg f, \\ F(k),\;\; \neg F(k),\;\; P_{\natural p}(k),\;\; \neg P_{\natural p}(k),\;\; k \leq k',\;\; k \not\leq k', \end{gathered}$$

where $k, k' \in \{i_1, i_2\}$, $i \in \{1, 2, 3, 4\}$, $* \in \{=, >\}$ and $\natural \in \{=, <, >\}$. The translation of all these literals is (equivalent to) a representative literal, except for the negative literals $\neg P_{\natural p}(k)$,[5] whose translation is however equivalent to the translation of a disjunction of positive literals (for instance, the translation of $\neg P_{<p}(k)$ is equivalent to the translation of $P_{=p}(k) \vee P_{>p}(k)$): now either one of these positive literal is in the diagram or none of them is there (which means that the negations are all there). In the former case, we have subsumption ($\neg P_{\natural p}(k)$ can be removed), in the latter case the translation of the whole diagram is inconsistent and is equivalent to the representative literal $i_1 \neq i_1$. Thus condition (i) of Theorem 4.6 holds.

[5] Notice that the translation of $\neg P_{=0}(k)$ is equivalent to $t[k] > 0$ because of the invariant saying that clocks are nonnegative.

Condition (ii) is trivial: it holds because if we take a representative literal L from the translation of the diagram of a model with two elements such that L mention just one of them (say i_1), then L is also a representative literal from the translation of the diagram of the submodel whose support is just $\{i_1\}$.

It remains to check condition (iii). This involves the analysis of the models of W having at most four elements; disregarding those having a diagram whose translation is manifestly inconsistent, we still have some millions of models to be examined. Instead of examining them one by one, we design a powerful heuristics. All representative literals coming from the translations of the diagrams of the four-elements models are included in the following list:

$$(L) \qquad \begin{array}{l} k \neq k', l[k] = 1, l[k] = 2, l[k] = 3, l[k] = 4, v = k, v \neq k, v = 0, v \neq 0, \\ t[k] = 0, t[k] > 0, t[k] < p, t[k] = p, t[k] > p, t[k] \leq t[k'], t[k] < t[k'], \end{array}$$

where $k, k' \in \{i_1, i_2, i_3, i_4\}$ - the literal $t[k] < t[k']$ is the translation of $k' \not\leq k$. (Notice that to get the list above, it is sufficient to consider models on a support with at most two elements, because there are no function symbols and at most binary predicates in Σ_W, so any model has a diagram which is the conjunction of the diagrams of all submodels of cardinality at most two.) If we succeed in proving that the formulae $\exists e\, \tau_1[L_1 \wedge L_2, f]$ and $\tau_h[L_1 \wedge L_2, f]$ $(2 \leq h \leq 6)$ are translations for every pair of literals L_1, L_2 coming from the above list (with possibly $L_1 \equiv L_2$), we actually proved more than what is required by condition (iii) (the limitation to at most two literals is due to the fact the QE-degree is 2).

Consider first discrete transitions $\tau_2 - \tau_6$: here there is no existential quantification over data, the formula $\tau_h[L_1 \wedge L_2, f]$ adds to $L_1 \wedge L_2$ a guard (which involves a further variable i but) which is a translation, adds also some literals (coming from case markings) that are equalities or negated equalities between index variables and finally replaces some $l[k]$ by a numerical value and some $t[k]$ by 0 (it possibly replaces v by 0 or by i). These replacements applied to literals from the list (L) always produce a literal from that list or an equality between index variables or a tautology or a contradiction (all this modulo A_I^E enriched with our invariants). Since the literals from (L) are translations, we are done.

It remains to analyze the time elapsing transition τ_1 where $\exists e\, \tau_1[L_1 \wedge L_2, f]$ does not have case-marking function and is $\exists e\, (e > 0 \wedge L_1^{+e} \wedge L_2^{+e})$ where L_i^{+e} is L_i after the substitution of the terms $t[k]$ with $t[k] + e$. The relevant cases to be analyzed are 28 and in all of them we get that $\exists e\, \tau_1[L_1 \wedge L_2, f]$ is a translation. For example, $\exists e\, (e > 0 \wedge t[k_1] + e > 0 \wedge t[k_2] + e = p)$ gives $p > t[k_2] \wedge t[k_1] + p - t[k_2] > 0$ which is equivalent to $p > t[k_2]$ (i.e. to $P_{<p}(k_2)^*$), taking into account the invariant saying that clocks are non-negative.

Thus, all conditions from Theorem 4.6 have been checked and we can predict termination of backward search for Fischer protocol. We emphasize that *the above arguments can be fully* *mechanized: they consist just in satisfiability checks that can be automatically generated and quickly discharged by suitable tools.*

5.2 Timed networks

We directly give here the formalization of timed networks (with a single real clock) as array-based systems.[6] We use as T_I the pure equality theory; for elements, we need a theory T_E which is the union of two theories T_{E_1}, T_{E_2}: the theory T_{E_1} is linear real arithmetic, the theory T_{E_2} is the enumeratated datatype theory of a finite set $Q = \{q_0, q_1, \dots, q_n\}$ (i.e. the signature of T_{E_2} has one constant for each element of Q and Q itself is the only model of T_{E_2}).

A *timed network* is an array-based system (I, a, τ) based on the combined theory A_I^E, where:

- a is a pair of array variables $\langle l, c\rangle$, whose target sorts are the sorts of T_{E_1} and T_{E_2}, respectively (l is the array of locations and c is the array of clocks);
- the formula I is $\forall i\,(l[i] = q_0 \wedge c[i] = 0)$ (this formula says that the initial locations are all equal to a fixed $q_0 \in Q$ and that clocks are all initialized to 0);
- τ contains the time elapsing transition τ_0 as well as a finite number of discrete transitions τ_h $(h = 1, \dots, t)$; the time elapsing transition is

  $$\exists e\,(e > 0 \wedge l' = l \wedge c' = \lambda j\,(c'[j] = c[j] + e))$$

  whereas the discrete transitions have the form

  $$\exists i \left(\begin{array}{l} l[i] = q_h \ \wedge\ c[i] \bowtie_h n_h \wedge \\ c' = \lambda j\,(\texttt{if } j = i \texttt{ then } op_h(c[j]) \texttt{ else } c[j]) \wedge \\ l' = \lambda j\,(\texttt{if } j = i \texttt{ then } q_{h'} \texttt{ else } l[j]\,) \end{array} \right)$$

  where $q_h, q_{h'} \in Q$, $n_h \in \mathbb{N}$, $\bowtie_h \in \{<, \leq, =, >, \geq\}$ and $op_h(c[j])$ is either 0 (i.e. a reset) or $c[j]$.

To formulate a *safety problem* for a given timed network, we need to specify a set of undesired states by an $\exists^I$-formula $U(a)$: for instance, if we want mutual exclusion at location q_c, we can take as U the formula

$$\exists i_1\,\exists i_2\,(i_1 \neq i_2 \wedge l[i_1] = q_c \wedge l[i_2] = q_c).$$

[6] For exposition clarity, we make a couple of simplifications with respect to the definition from [6]: we disregard the controller state and we limit the number of existentially quantified index variables considered in a transition to 1. Both limitations can be easily dropped: for controller's states, it is sufficient to use a constant array (see the previous Subsection), for existential index variables it is sufficient to use a vector notation.

Our point is now to apply Theorem 4.6 to *statically* prove termination. The following Remark is important to understand our point:

Remark 5.1. Theorem 4.6 gives an algorithm that, once implemented and run against a *single* problem specification, can predict in advance (in successful cases) termination of backward search. Thus, for instance, if we run the algorithm against the specification of the Fischer protocol of Subsection 5.1, it will provide us the desired termination certification. However, Theorem 4.6 itself is not a result concerning entire classes of problems, because classes of problems rely on formal *templates* involving unknown parameters (e.g timed networks described above contains various parameters, like the number of the transitions, the number of the locations, the constants mentioned in the guards, etc.) and there is nothing in the statement of Theorem 4.6 that can support such templates. So, what we can do here in this Subsection is to supply a general *meta-argument* that shows *why the algorithm of Theorem 4.6 succeeds on every single reachability problem for timed networks.*

In order to apply Theorem 4.6 to a timed network (with a single real clock) (I, a, τ) as above, we use the theory W of Example 4.2, relatively to the following set of unary predicates (we let N to be the maximum constant appearing in the guards of the τ_h):

1. Q_k for any $k = 1 \dots n$;

2. $P_k^=$ for any $k = 0 \dots N$;

3. $P_k^>$ for any $k = 0 \dots N-1$;

4. P_N.

The syntactic translation $(\_)^*$ we use is:

1. $Q_k(i)^* := (l[i] = q_k)$ for any $k = 1 \dots n$;

2. $P_k^=(i)^* := (c[i] = k)$ for any $k = 0 \dots N$;

3. $P_k^>(i)^* := (k < c[i] \wedge c[i] < k+1)$ for any $k = 0 \dots N-1$;

4. $P_N(i)^* := (c[i] > N)$;

5. We finally let $(i_1 \leq i_2)^*$ to be

$$\begin{aligned} & P_N(i_2)^* \vee \bigvee_{k=0\dots N} P_k^=(i_1)^* \vee \\ & \vee \bigvee_{k_1,k_2=0\dots N-1} \Big(P_{k_1}^>(i_1)^* \wedge P_{k_2}^>(i_2)^* \wedge (c[i_1] - c[i_2] \leq k_1 - k_2) \Big) \end{aligned}$$

The translation of $\leq$ follows closely Definition 5.2 of [6]: informally, we have that $(i_1 \leq i_2)^*$ holds iff either i_2's clock is larger than N or the fractional part of the clock of i_1 is smaller than the fractional part of the clock of i_2.

We first notice that the translation is suitable, i.e. that the translation of the axioms of W

$$\forall x\,(x \leq x), \quad \forall x, y, z\,(x \leq y \wedge y \leq z \rightarrow x \leq z), \quad \forall x, y\,(x \leq y \vee y \leq x).$$

are A_I^E-valid. We omit the straightforward detailed verification.

Next, we consider condition (i) from Theorem 4.6. We pick $<, \leq, =$ as representative predicates for real linear arithmetic (for T_{E_1} all literals are representative); we get QE-degree equal to 2 by this choice, see Example 4.4. We apply the heuristics already successfully used for the Fischer protocol and we notice that the diagram of a two-element model contains only literals from the following list

$$j \neq j', \;\; Q_h(j), \;\; \neg Q_h(j), \;\; P_k^=(j), \;\; \neg P_k^=(j),$$
$$P_l^>(j), \;\; \neg P_l^>(j), \;\; P_N(j), \;\; \neg P_N(j), \;\; j \leq j', \;\; j \not\leq j',$$

where $j, j' \in \{i_1, i_2\}$, $h \in \{1, \ldots, n\}$, $l \in \{0, \ldots, N-1\}$ and $k \in \{0, \ldots, N\}$. The translation of all these literals are conjunction of representative literals, except the following ones

$$\neg Q_h(j), \qquad \neg P_k^=(j), \qquad \neg P_l^>(j), \qquad \neg P_N(j), \qquad j \leq j', \qquad j \not\leq j'.$$

Now we have that

(i) the translation of $\neg Q_h(j)$ is equivalent to $\bigvee_{h' \neq h} Q_{h'}(j)^*$;

(ii) the translation of $\neg P_k^=(j)$ is equivalent to $\bigvee_{l \neq k, l \leq N} P_l^=(j)^* \vee \bigvee_{l<N} P_l^>(j)^* \vee P_N(j)^*$;

(iii) the translation of $\neg P_l^>(j)$ is equivalent to

$$\bigvee_{0 \leq l' \leq N} P_{l'}^=(j)^* \vee \bigvee_{l' \neq l, 0 \leq l' < N} P_{l'}^>(j)^* \vee P_N(j)^*;$$

(iv) the translation of $\neg P_N(j)$ is equivalent to

$$\bigvee_{0 \leq l' \leq N} P_{l'}^=(j)^* \vee \bigvee_{0 \leq l' < N} P_{l'}^>(j)^* \vee P_N(j)^*;$$

(v) the translation of $j \leq j'$ is equivalent to

$$P_N(j')^* \vee \bigvee_{0 \leq k \leq N} P_k^=(j)^* \vee \bigvee_{0 \leq k_1, k_2 < N} \left(P_{k_1}^>(j)^* \wedge P_{k_2}^>(j')^* \wedge \left(c[j] - c[j'] \leq k_1 - k_2\right)\right);$$

(vi) the translation of $j \not\leq j'$ is equivalent to

$$\neg P_N(j')^* \wedge \bigwedge_{0 \leq k \leq N} \neg P_k^=(j)^* \wedge \bigwedge_{0 \leq k_1, k_2 < N} \big(\neg P_{k_1}^>(j)^* \vee \neg P_{k_2}^>(j')^* \vee \big(c[j] - c[j'] > k_1 - k_2\big)\big)\,;$$

(where the translated negative literals $\neg P_N(j')^*, \neg P_k^=(j)^*, \neg P_{k_1}^>(j)^*, \neg P_{k_2}^>(j')^*$ should be further replaced by the formulae indicated in (ii)-(iii)-(iv)).

Now, *within a diagram*, keeping in mind that a diagram always contains either a literal or its negation, we have that

(i) the translation of $\neg Q_h(j)$ simplifies either to $\top$ or to $\bot$;

(ii) the translation of $\neg P_k^=(j)$ simplifies either to $\top$ or to $\bot$;

(iii) the translation of $\neg P_l^>(j)$ simplifies either to $\top$ or to $\bot$;

(iv) the translation of $\neg P_N(j)$ simplifies either to $\top$ or to $\bot$;

(v) the translation of $j \leq j'$ simplifies either to $\top$ or to $\bot$ or to $c[j] - c[j'] \leq k_1 - k_2$ for some positive $k_1, k_2 < N$;

(vi) the translation of $j \not\leq j'$ simplifies either to $\top$ or to $\bot$ or to $c[j] - c[j'] > k_1 - k_2$ for some positive $k_1, k_2 < N$.

So we get in any case a conjunction of representative literals and condition (i) of Theorem 4.6 holds.

Condition (ii) of Theorem 4.6 is a trivial statement, so it remains to check condition (iii). Again, we follow the same schema used in the case of Fischer protocol and we list the set of representative literals occurring in the translations of the diagram of four element models. These are the following:

$$j \neq j', \qquad l[j] = h, \qquad c[j] \bowtie k, \qquad c[j] - c[j'] \Join k_1 - k_2 \tag{5}$$

where $j, j' \in \{i_1, i_2, i_3, i_4\}$, $h \in \{0, \dots, n\}$, $\bowtie \in \{<, >, =\}$, $\Join \in \{\leq, >\}$, $k \in \{0, \dots, N\}$ and $k_1, k_2 \in \{0, \dots, N-1\}$. Following the heuristics applied in the case of the Fischer protocol, we should prove that the formulae

$$\exists e\, \tau_1[L_1 \wedge L_2, f], \qquad \tau_h[L_1 \wedge L_2, f] \quad (h > 1) \tag{6}$$

are all translations for every L_1, L_2 of the kind specified in (5). However, this attempt fails, basically because it is not true here that all literals in (5) are translations. A slight modification of our heuristics replaces the list (5) by a so-called *adequate set S of conjunctions of* *literals.* This is a set of conjunction of representative literals such that for every representative literal L occurring in the translation $\delta^*_{\mathcal{M}}$ of a diagram of a model $\mathcal{M}$ whose support has at most 4 elements, there is $C \in S$ such that L occurs in C and $\delta^*_{\mathcal{M}}$ is of the kind $C \wedge C'$ for some C' (notice that in order to identify an adequate S it is sufficient to inspect two-elements models only, because predicates of W have at most arity 2). In our case, an adequate S is given by the following list of conjunctions of literals:

(1) $i_1 \neq i_2$;

(2) $l[i] = q_k$;

(3) $c[i] \bowtie k$, where $\bowtie \in \{<, >, =\}$ and $k \in \{0, \dots, N\}$;

(4) $c[i_1] - c[i_2] \bowtie k_1 - k_2 \wedge k_1 < c[i_1] \wedge c[i_1] < k_1 + 1 \wedge k_2 < c[i_2] \wedge c[i_2] < k_2 + 1$ where $\bowtie \in \{\leq, >\}$ and $k_1, k_2 \in \{0, \dots, N-1\}$.

The reason why this set is adequate should be clear by inspecting (iii)-(iv) above: in fact, in a diagram the translations of $i_1 \leq i_2$ and of $i_1 \not\leq i_2$ either simplify to $\top, \bot$ or to a conjunction of the kind (4) above. So *it remains to check that the formulae*

$$\exists e\, \tau_1[C_1 \wedge C_2, f], \qquad \tau_h[C_1 \wedge C_2, f] \quad (h > 1) \tag{7}$$

are translations for all $C_1, C_2 \in S$. This is indeed true and can be verified mechanically (once N and the τ_h are fixed); manually, it is much easier to show a more general 'smooth' result that can be stated as follows:

Proposition 5.2. *Let us call a* bounded constraint *a conjunction of formulae of the following kinds: (a)* $\top$*; (b)* $\bot$*; (c)* $i_1 = i_2$*; (d)* $i_1 \neq i_2$*; (e)* $l[i] = q_k$*; (f)* $c[i] \lhd k$*, where* $\lhd \in \{<, \leq, >, \geq, =\}$*,* $k \in \{0, \dots, N\}$*; (g)* $c[i_1] - c[i_2] \lhd k \wedge c[i_1] \lhd k_1 \wedge c[i_2] \lhd k_2$ *where* $\lhd \in \{<, \leq, \}$*,* $k \in \{-N, \dots, N\}$ *and* $k_1, k_2 \in \{0, \dots, N\}$*. A bounded constraint* $K(\underline{i})$ *enjoys the following properties:*

(i) *it is a translation (i.e. there is* $\alpha(\underline{i})$ *such that* K *is equivalent to* α^**);*

(ii) *if* K' *is obtained from* K *by replacing a term of the kind* $c[i]$ *occurring in* K *by 0, then* K' *is a bounded constraint;*

(iii) *if* K' *is obtained from* K *by replacing a term of the kind* q_h *occurring in* K *by* q'_h*, then* K' *is a bounded constraint;*

(iv) *if* K' *is* $\exists e\,(e > 0 \wedge K^{+e})$*, where* K^{+e} *is obtained from* K *by replacing all terms of the kind* $c[i]$ *occurring in* K *by* $c[i] + e$*, then* K' *is equivalent to a disjunction of bounded constraints.*

The proof of the above Proposition is a straightforward computation. We just mention the two relevant cases, requiring little symbolic manipulation.

- How to check the statement (i) in the sub-case K is $c[i_1]-c[i_2] \lhd k \wedge n_1 < c[i_1] < n_1+1 \wedge n_2 < c[i_2] < n_2+1$ for some $n_1, n_2 \in \{0, \ldots, N-1\}$ (notice that case (g) can be easily reduced to a disjunction of such subcases). We have that $k \geq n_1 - n_2 + 1$ or $k = n_1 - n_2$ or $k \leq n_1 - n_2 - 1$. If $k \geq n_1 - n_2 + 1$, since $c[i_1] - c[i_2] < n_1 - n_2 + 1$, then $c[i_1] - c[i_2] \lhd k$. Thus this is redundant, and $n_1 < c[i_1] < n_1 + 1 \wedge n_2 < c[i_2] < n_2 + 1$ is a translation of $P^{>}_{n_1}(i_1) \wedge P^{>}_{n_2}(i_2)$. If $k \leq n_1 - n_2 - 1$, since $c[i_1] - c[i_2] > n_1 - n_2 - 1$, and by hypothesis $c[i_1] - c[i_2] \lhd k \leq n_1 - n_2 - 1$, then the formula is inconsistent. If $k = n_1 - n_2$, the formula is equivalent to $c[i_1] - c[i_2] \lhd n_1 - n_2 \wedge n_1 < c[i_1] < n_1 + 1 \wedge n_2 < c[i_2] < n_2 + 1$, which is the translation of $P^{>}_{n_1}(i_1) \wedge P^{>}_{n_2}(i_2) \wedge i_1 \leq i_2$ (in case $\lhd$ is $\leq$) or of $P^{>}_{n_1}(i_1) \wedge P^{>}_{n_2}(i_2) \wedge i_2 \not\leq i_1$ (in case $\lhd$ is $<$).
- How to check the statement (iv) in case K is $c[i_1] < k_1 \wedge c[i_2] > k_2$. We have that $\exists e\, (e > 0 \wedge K^{+e})$ is equivalent to $c[i_1] < k_1 \wedge c[i_1] - c[i_2] < k_1 - k_2$; the latter is equivalent to

$$(c[i_1] < k_1 \wedge c[i_2] \leq k_2 \wedge c[i_1] - c[i_2] < k_1 - k_2) \vee$$
$$\vee (c[i_1] < k_1 \wedge c[i_2] > k_2 \wedge c[i_1] - c[i_2] < k_1 - k_2)$$

and finally to

$$(c[i_1] < k_1 \wedge c[i_2] \leq k_2 \wedge c[i_1] - c[i_2] < k_1 - k_2) \vee (c[i_1] < k_1 \wedge c[i_2] > k_2)$$

which is a disjunction of bounded constraints.

6 Conclusions

We identified a sufficient condition for the termination of a symbolic backward reachability procedure encompassing many results from the literature in a uniform and declarative framework. We believe that the statement of Theorem 4.6 could be seen as a paradigm for a declaratively-oriented approach to termination; the statement itself needs to be further investigated and exploited in connection to more examples of wqo-theories and syntactic translations arising from encoding termination arguments based on Kruskal theorem.

An interesting direction for future work consists in applying the methods of this paper in connection to abstraction techniques: our results could be profitably employed to predict whether a proposed abstraction of a system yields a terminating search.

References

[1] P. A. Abdulla, K. Cerans, B. Jonsson, and Y.-K. Tsay. General decidability theorems for infinite-state systems. In *Proc. of LICS*, pages 313–321, 1996.

[2] P. A. Abdulla, G. Delzanno, N. B. Henda, and A. Rezine. Regular model checking without transducers. In *TACAS*, volume 4424 of *LNCS*, pages 721–736, 2007.

[3] P. A. Abdulla, J. Deneux, and P. Mahata. Multi-clock timed networks. In *Proc. of LICS'04, the 18th IEEE Int. Symp. on Logic in Computer Science*, 2004.

[4] Parosh Aziz Abdulla. Forcing monotonicity in parameterized verification: From multisets to words. In *Proc. of SOFSEM '10*, pages 1–15. Springer-Verlag, 2010.

[5] Parosh Aziz Abdulla and Bengt Jonsson. Verifying programs with unreliable channels. *Information and Computation*, 127(2):91–101, 1996.

[6] Parosh Aziz Abdulla and Bengt Jonsson. Model checking of systems with many identical timed processes. *Theoretical Computer Science*, pages 241–264, 2003.

[7] F. Alberti, S. Ghilardi, E. Pagani, S. Ranise, and G. P. Rossi. Brief Announcement: Automated Support for the Design and Validation of Fault Tolerant Parameterized Systems—a case study. In *DISC 10*, number 6343 in LNCS, pages 392–394, 2010.

[8] A. Carioni, S. Ghilardi, and S. Ranise. MCMT in the Land of Parametrized Timed Automata. In *Proc. of VERIFY 10*, 2010.

[9] Chen-Chung Chang and Jerome H. Keisler. *Model Theory.* North-Holland, Amsterdam-London, third edition, 1990.

[10] G. Delzanno, J. Esparza, and A. Podelski. Constraint-based analysis of broadcast protocols. In *Proc. of CSL*, volume 1683 of *LNCS*, pages 50–66, 1999.

[11] Herbert B. Enderton. *A Mathematical Introduction to Logic.* Academic Press, New York-London, 1972.

[12] J. Esparza, A. Finkel, and R. Mayr. On the verification of broadcast protocols. In *Proc. of LICS*, pages 352–359. IEEE Computer Society, 1999.

[13] S. Ghilardi, E. Nicolini, S. Ranise, and D. Zucchelli. Towards SMT Model-Checking of Array-based Systems. In *Proc. of IJCAR*, LNCS, 2008.

[14] S. Ghilardi and S. Ranise. A Note on the Stopping Failures Models. 2009. Unpublished Draft, available from MCMT web site.

[15] S. Ghilardi and S. Ranise. Goal Directed Invariant Synthesis for Model Checking Modulo Theories. In *(TABLEAUX 09)*, LNAI, pages 173–188. Springer, 2009.

[16] S. Ghilardi and S. Ranise. Model Checking Modulo Theory at work: the integration of Yices in MCMT. In *AFM 09 (co-located with CAV09)*, 2009.

[17] S. Ghilardi and S. Ranise. Backward reachability of array-based systems by SMT-solving: termination and invariant synthesis. *LMCS*, 6(4), 2010.

[18] S. Ghilardi, S. Ranise, and T. Valsecchi. Light-Weight SMT-based Model-Checking. In *Proc. of AVOCS 07-08*, ENTCS, 2008.

[19] Silvio Ghilardi and Silvio Ranise. MCMT: A Model Checker Modulo Theories. In *Proc. of IJCAR 2010*, volume 6173 of *LNCS*, pages 22–29. Springer, 2010.

[20] Schnoebelen Philippe. Verifying lossy channel systems has nonprimitive recursive complexity. *Information Processing Letters*, 83(5):251–261, 2002.

[21] S. Ranise and C. Tinelli. The SMT-LIB Standard: Version 1.2. Technical report, Dep. of Comp. Science, Iowa, 2006. Available at `http://www.SMT-LIB.org/papers`.

A Wqo-theories

In this Section we justify the claim we made in the Example 4.2, where we showed a family of wqo theories W (these are also the wqo theories we used in Section 5). Recall that the signature Σ_W of W contains a binary predicate '$\leq$' (constrained by reflexivity, transitivity and linearity axioms),[7] besides a finite set $\mathcal{P}$ of unary predicates (0-ary predicates can also be added, dealing with the related extension is easy and we leave it to the reader).

We let σ be the set of all the multisets of subsets of $\mathcal{P}$: an element of σ is a thus function $M : \wp(\mathcal{P}) \longrightarrow \mathbb{N}$, where $\mathbb{N}$ is the set of natural numbers. We say that a multiset M is smaller than another multiset N iff $M(H) \leq N(H)$ holds for all $H \in \wp(\mathcal{P})$. We let σ^* be the set of finite words from σ; σ^* is ordered by the relation $v \preceq w$ that holds iff (i) w is equal to $M_1 \cdots M_n$; (ii) v is equal to $N_1 \cdots N_k$; (iii) for some $1 \leq i_1 < \cdots < i_k \leq n$, we have $N_1 \leq M_{i_1}, \dots, N_k \leq M_{i_k}$. It follows from Higman's results (see Lemma 7.2 in [6]) that $< \sigma^*, \preceq >$ is a wqo.

To show that W is a wqo theory, we proceed as follows: (1) we associate with every finitely generated model $\mathcal{M}$ of W some $\theta(\mathcal{M}) \in \sigma^*$; (2) we show that if $\theta(\mathcal{M}) \preceq \theta(\mathcal{N})$ holds, then $\mathcal{M}$ embeds into $\mathcal{N}$. From (1) and (2), it is immediate to see that finitely generated models of W are a wqo with respect to the embeddability relation.

Consider a finitely generated model $\mathcal{M}$ of W: since the signature is relational, the model is finite; since $\leq^{\mathcal{M}}$ is reflexive, transitive and linear, the support $|\mathcal{M}|$ of $\mathcal{M}$ can be partitioned as $\bigcup_{i=1}^{n} E_i$ in such a way that (for $p, q \in |\mathcal{M}|$) $\mathcal{M} \models p \leq q$ holds iff there are $i \leq j$ such that $p \in E_i$ and $q \in E_j$. To each $p \in |\mathcal{M}|$ we can also associate the subset $S(p)$ of $\mathcal{P}$ formed by the predicates P such that $\mathcal{M} \models P(p)$. In conclusion, we can extract from each finite $\mathcal{M}$ the data $< E_1, \dots, E_n, S >$; we now define $\theta(\mathcal{M})$ to be the word of multisets $M_1 \cdots M_n$, where $M_i(H)$ (for $i = 1, \dots, n$) is taken to be the number of the $p \in E_i$ such that $S(p) = H$.

It remains to prove (2): suppose that $\theta(\mathcal{M}) \preceq \theta(\mathcal{N})$ holds. Suppose also that $(E_1, \dots, E_k, S)$ are the data extracted from $\mathcal{M}$, that $(F_1, \dots, F_n, T)$ are the data extracted from $\mathcal{N}$, that $\theta(M)$ is equal to $M_1 \cdots M_k$ and that $\theta(N)$ is is equal to $N_1 \cdots N_n$: we have $M_1 \leq N_{i_1}, \dots, M_k \leq N_{i_k}$, for some $1 \leq i_1 < \cdots < i_k \leq n$. To define the embedding $\mu : \mathcal{M} \longrightarrow \mathcal{N}$ it is sufficient to notice that for every $H \subseteq \wp(\mathcal{P})$ and $j = 1, \dots, k$, there are (by construction) more $q \in F_{i_k}$ such that $T(q) = H$ than $p \in E_k$ such that $S(p) = H$: this means that it is possible to associate in a one-to-one way with every $p \in |\mathcal{M}|$ some $\mu(p) \in |\mathcal{N}|$ in such a way that $S(p) = T(\mu(p))$ and in such a way that $p \in E_j$ iff $\mu(p) \in F_{i_j}$ (for every $j = 1, \dots, k$). This is precisely what is needed for μ to be an embedding, because in this way p and $\mu(p)$ satisfy the same predicates from $\mathcal{P}$ and $\leq$ is preserved and reflected along μ.

[7] Notice that antisymmetry is not assumed.

B Proof of Theorem 4.6

In this Section we report the proof of Theorem 4.6. We start with some lemmas.

Lemma B.1. *Let T be a theory eliminating quantifiers for a sort S and having QE-degree N with respect to a set of representative predicates. For every finite family $\{L_i\}_{i\in I}$ of representative literals, the formula*

$$\exists x\,(\bigwedge_{i\in I} L_i) \leftrightarrow \bigwedge_{J\in\mathcal{F}} \exists x\,(\bigwedge_{i\in J} L_i) \tag{8}$$

is T-valid, provided x has sort S and the family $\mathcal{F}$ of subsets of I has the property that for every $I_0 \subseteq_N I$ there is some $J \in \mathcal{F}$ such that $I_0 \subseteq J$.

Proof. Trivial because $\bigwedge_{J\in\mathcal{F}} \exists x\,(\bigwedge_{i\in J} L_i)$ entails $\bigwedge_{I_0\subseteq_N I} \exists x\,(\bigwedge_{i\in I_0} L_i)$ and the latter entails $\exists x\,(\bigwedge_{i\in I} L_i)$ (modulo T) by (4) (the reverse implication is a logical validity). □

Lemma B.2. *Every existential Σ_W-sentence $\exists \underline{i}\,\alpha(\underline{i})$ is equivalent (modulo W) to the existential closure of a disjunction of formulae which are diagrams of finite models of W.*

Proof. It is sufficient to prove the claim for the sentences $\exists \underline{i}\,\alpha(\underline{i})$ which are primitive and differentiated (i.e. those existential sentences $\exists \underline{i}\,\alpha(\underline{i})$ for which $\alpha(\underline{i})$ is a conjunction of literals containing for every distinct $i_1, i_2 \in \underline{i}$ the literal $i_1 \neq i_2$): in fact, every existential formula can be rewritten as a disjunction of such formulae by using disjunctive normal forms and the validities $\exists i_2(i_1 = i_2 \wedge \beta) \leftrightarrow \beta(i_2/i_1)$. Let $\mathcal{M}_j$ ($j \in J$) be the finite minimal models of W satisfying our primitive differentiated $\exists \underline{i}\,\alpha(\underline{i})$: notice that every model of W has a minimal submodel satisfying our formula because the theory W is universal, the signature is relational and the formula is differentiated (so a minimal model can be obtained by taking the substructure generated by the elements assigned to the $\underline{i}$ by a satisfying assignment). Since the formula is differentiated, the supports of such minimal models $\mathcal{M}_j$ are generated by the $\underline{i}$ and the diagrams $\delta_{\mathcal{M}_j}(\underline{i})$ all entails $\alpha(\underline{i})$ (modulo W).[8] In particular (as Σ_W does not have function symbols), J is finite, we can write the formula $\bigvee_j \delta_{\mathcal{M}_j}(\underline{i})$ and get that $W \models \bigvee_j \delta_{\mathcal{M}_j}(\underline{i}) \to \alpha$. Vice versa, it is not possible to find a model of W satisfying α and the negations of all the $\delta_{\mathcal{M}_j}(\underline{i})$, because there would be a minimal such which would be among the $\mathcal{M}_j$. □

[8] This is obtained by a classical 'diagram' argument (see [9]). In detail (we expand the language with free constants $\underline{i}$ for notation simplicity): if for a model $\mathcal{M}$ of W in the expanded language, we have $\mathcal{M} \models \delta_{\mathcal{M}_j}$, then $\mathcal{M}_j$ has an embedding into $\mathcal{M}$ (by Robinson diagram lemma) and the truth of α transfers to $\mathcal{M}$ via the embedding because α does not have quantifiers. Since $\mathcal{M}$ is arbitrary, we have $W \models \delta_{\mathcal{M}_j}(\underline{i}) \to \alpha(\underline{i})$.

Lemma B.3. *Suppose that $\delta_{\mathcal{M}}(\underline{i})$ is the diagram of a finite model of W and let $\psi(\underline{i}, a[\underline{i}])$ be a formula such that $A_I^E \models \delta^*_{\mathcal{M}} \to \psi$. Let $\underline{t}$ be a tuple of A_I^E-terms of suitable length and let $\delta^*_{\mathcal{M}}(\underline{i}, \underline{t}), \psi(\underline{i}, \underline{t})$ be the formulae obtained by replacing componentwise all the occurrences of the $a[\underline{i}]$ with the $\underline{t}$ in $\delta^*_{\mathcal{M}}, \psi$, respectively. Then we have $A_I^E \models \delta^*_{\mathcal{M}}(\underline{i}, \underline{t}) \to \psi(\underline{i}, \underline{t})$.*

Proof. Clearly, $\delta^*_{\mathcal{M}}$ is a quantifier-free formula of the kind $\phi(\underline{i}, a[\underline{i}])$,[9] which is differentiated, i.e. it contains as conjuncts all the inequations $i_1 \neq i_2$ for all different $i_1, i_2 \in \underline{i}$ (this is because inequations between distinct elements from the support of a model belongs to its diagram and inequations are translated identically by $(-)^*$). We shall prove that if $\phi(\underline{i}, a[\underline{i}])$ is differentiated and it entails (modulo A_I^E) the formula $\psi(\underline{i}, a[\underline{i}])$, then $\phi(\underline{i}, \underline{e})$ entails (modulo $T_I \cup T_E$) the formula $\psi(\underline{i}, \underline{e})$, where $\underline{e}$ is a (length and sorts matching) tuple of distinct variables. In fact, if it is not the case, there is a model $\mathcal{N}$ of $T_I \cup T_E$ such that $\mathcal{N} \models \phi(\underline{i}, \underline{e}) \wedge \neg\psi(\underline{i}, \underline{e})$ (under a suitable assignment to the variables $\underline{i}, \underline{e}$). Now, expand $\mathcal{N}$ to a model of A_I^E, by interpreting the $\mathtt{ARRAY}_\ell$ sorts in the obvious way, i.e. as functions sets. Since the (elements assigned to) the $\underline{i}$ are distinct, we can assign to the a functions whose values at places $\underline{i}$ are just the elements previously assigned to the $\underline{e}$, thus getting a models in which the entailment $\phi(\underline{i}, a[\underline{i}]) \to \psi(\underline{i}, a[\underline{i}])$ fails. Since we just established that $T_I \cup T_E \models \phi(\underline{i}, \underline{e}) \to \psi(\underline{i}, \underline{e})$, it follows also that $A_I^E \models \phi(\underline{i}, \underline{t}) \to \psi(\underline{i}, \underline{t})$ holds for every $\underline{t}$. □

We are now ready to face the complexity of the main proof. We refer to Figure 2 for the pseudo-code of our backward reachability procedure: in the Figure, the variables P and B are used to store the formulae describing the states that can reach in n-steps (in at most n-steps, rspectively) an unsafe state, i.e. a state satisfying U. Below, we use the following notation: if $\mathcal{M}$ is a model and $\underline{i}$ are elements from the support of $\mathcal{M}$, we indicate by $\mathcal{M}[\underline{i}]$ the substructure of $\mathcal{M}$ generated by the $\underline{i}$ (notice that, if the signature is relational, the support of $\mathcal{M}[\underline{i}]$ will coincide with $\underline{i}$).

Proof. We start the proof of of Theorem 4.6 with the following claim:

(*) the content of the variable P in the algorithm from Figure 2 is always an $\exists^I$-formula which is a translation.

This is true when the variable is initialized because U is a translation by hypothesis. Hence we only need to check that any formula of the kind $Pre(\tau_h, \exists \underline{i}\, \alpha^*)$ is always a translation. Below, we suppose that τ_h is given by (1) and that the case defined functions $F = F_1, \ldots, F_s$ occurring in (1) are specified as in (2). By Lemma B.2 and since the disjunction of translations

[9] Recall that the a are considered as free constants in the specification of an array-based system, so they can occur too.

```
function BReach((a, I, τ), U)
P ⟵ U; B ⟵ ⊥;
while (P ∧ ¬B is $A_I^E$-sat.) do
    if (I ∧ P is $A_I^E$-sat.) then return unsafe;
    B ⟵ P ∨ B;
    P ⟵ Pre(τ, P);
end
return (safe, B);
```

Figure 2: The symbolic backward reachability algorithm

is a translation, it is sufficient to check that

$$\exists a' (\exists e\, \exists \underline{k}\, (\phi_L(e, \underline{k}, a[\underline{k}]) \wedge a' = \lambda j. F(e, \underline{k}, a[\underline{k}], j, a[j])) \wedge \exists \underline{i}\, \delta^*_{\mathcal{M}}(\underline{i}, a[\underline{i}])) \tag{9}$$

is a translation for every finite model $\mathcal{M}$ of W. Here we suppose that the support of $\mathcal{M}$ has the same cardinality as the tuple of variables $\underline{i}$, so that the diagram of $\mathcal{M}$ can be written as a quantifier free formula $\delta_{\mathcal{M}}(\underline{i})$.

Now notice that, since $\delta_{\mathcal{M}}$ is the conjunctions of the true literals in $\mathcal{M}$ and since each true literal can mention at most M elements from the support of $\mathcal{M}$ (because there are no functions symbols in Σ_W and because Σ_W-predicates have at most arity M), we have that $\delta_{\mathcal{M}}$ is the conjunction of the diagrams $\delta_{\mathcal{M}[\underline{i}_0]}$ varying $\underline{i}_0$ among the subsets of the support of $\mathcal{M}$ having at most cardinality M. Thus, we can rewrite (9) as

$$\exists \underline{i}\, \exists \underline{k}\, \exists e\, \exists a' (\phi_L(e, \underline{k}, a[\underline{k}]) \wedge a' = \lambda j. F(e, \underline{k}, a[\underline{k}], j, a[j]) \wedge \bigwedge_{\underline{i}_0 \subseteq_M \underline{i}} \delta^*_{\mathcal{M}[\underline{i}_0]}(\underline{i}_0, a[\underline{i}_0])). \tag{10}$$

According to hypothesis (i) of Theorem 4.6, the translation $\delta^*_{\mathcal{M}[\underline{i}_0]}$, is equivalent (modulo A_I^E) to a conjunction of representative literals involving the variables $\underline{i}_0$ of sort INDEX.

By standard logical passages (β-conversion, elimination of case-defined function symbols, and the like), this formula can be rewritten to the disjunction of the formulae

$$\begin{aligned} &\exists \underline{k}\, \exists \underline{i}\, \exists e\, (\phi_L \wedge \textstyle\bigwedge_{i \in \underline{i}} C_f(i)(i, a[i]) \wedge \\ &\quad \wedge \textstyle\bigwedge_{\underline{i}_0 = \{i_{l_1}, \dots, i_{l_{M'}}\} \subseteq_M \underline{i}} \delta^*_{\mathcal{M}[\underline{i}_0]}(\underline{i}_0, t_{1f(i_{l_1})}(i_{l_1}, a[i_{l_1}]), \dots, t_{sf(i_{l_{M'}})}(i_{l_{M'}}, a[i_{l_{M'}}])) \end{aligned}$$

varying f under the set of the appropriate case-marking functions (notice that, here and below, we follow the convention of not displaying the dependency of our terms and formulae on $\underline{k}, a[\underline{k}], e$). We need to show that the formula obtained from the elimination of the single existentially quantified variable e from

$$\exists e\, (\phi_L \wedge \bigwedge_{i \in \underline{i}} C_f(i)(i, a[i]) \wedge \bigwedge_{\underline{i}_0} \delta^*_{\mathcal{M}[\underline{i}_0]}(\underline{i}_0, t_{1f(i_{l_1})}(i_{l_1}, a[i_{l_1}]), \dots, t_{sf(i_{l_{M'}})}(i_{l_{M'}}, a[i_{l_{M'}}])) \tag{11}$$

is A_I^E-equivalent to a formula of the kind β^*, for some quantifier-free Σ_W-formula β (in (11), the index $\underline{i}_0$ varies again on the subsets $\{i_{l_1}, \dots, i_{l_{M'}}\} \subseteq_M \underline{i}$). We first observe that the matrix of the formula (11) is a conjunction of representative literals, by Assumption (**E2**).[10] Thus we can apply Lemma B.1: we choose as $\mathcal{F}$ the family of the sets of literals occurring in (11) mentioning (besides the $e, \underline{k}$) only the variables belonging to a given subset $\underline{j}$ of $\underline{i}$ having at most cardinality $M * N$. This choice satisfies the condition of Lemma (B.2) because N literals taken from (11) can mention at most $N * M$ variables belonging to $\underline{i}$: in fact, at most M variables - besides $\underline{k}, e$ - can occur in $\delta^*_{\mathcal{M}[\underline{i}_0]}(\underline{i}_0, t_{1f(i_{l_1})}(i_{l_1}, a[i_{l_1}]), \dots, t_{sf(i_{l_{M'}})}(i_{l_{M'}}, a[i_{l_{M'}}]))$ because $\underline{i}_0$ has at most cardinality M and because the terms $t_{rf(i_{l_p})}$ depend only on the single displayed variable i_{l_p} (besides $\underline{k}, e$).

Our claim (*) is then proved if we show that the elimination of the existentially quantified variable e yields a formula of the kind β^* for every choice of $\underline{j} \subseteq_{N*M} \underline{i}$. The idea is to apply the hypothesis (iii) from Theorem 4.6: to succeed, we show that selecting from the matrix of (11) just the literals containing (besides $\underline{k}, e$) only the variables $\underline{j}$ yields precisely the formula $\tau_h[\delta^*_{\mathcal{M}[\underline{j}]}, \bar{f}]$, where $\bar{f}$ is the case-marking function obtained by restricting f to $\underline{j}$ in the domain.[11] Now recall that $\tau_h[\delta^*_{\mathcal{M}[\underline{j}]}, \bar{f}]$ is

$$\phi_L \wedge \bigwedge_{i \in \underline{j}} C_{\bar{f}(i)}(i, a[i]) \wedge \delta^*_{\mathcal{M}[\underline{j}]}(\underline{j}, t_{1\bar{f}(i_{l_1})}(i_{l_1}, a[i_{l_1}]), \dots, t_{s\bar{f}(i_{l_K})}(i_{l_K}, a[i_{l_K}])) \tag{12}$$

(here we suppose that $\underline{j} = i_{l_1}, \dots, i_{l_K}$ for $K \leq M * N$). However, the diagram of $\mathcal{M}[\underline{j}]$ is the conjunction of the diagrams of the substructures of $\mathcal{M}[\underline{j}]$ having at most cardinality M, hence we can rewrite (12) as

$$\phi_L \wedge \bigwedge_{i \in \underline{j}} C_f(i)(i, a[i]) \wedge \bigwedge_{\underline{i}_0} \delta^*_{\mathcal{M}[\underline{i}_0]}(\underline{i}_0, t_{1f(i_{l_1})}(i_{l_1}, a[i_{l_1}]), \dots, t_{sf(i_{l_{M'}})}(i_{l_{M'}}, a[i_{l_{M'}}])) \tag{13}$$

(with the index $\underline{i}_0$ ranging over the subsets $\{i_{l_1}, \dots, i_{l_{M'}}\} \subseteq_M \underline{j}$). This is precisely the conjunctions of the literals from the matrix of (11) mentioning at most the $\underline{j}$ (besides the $\underline{k}, e$), *because of the hypothesis (ii) of Theorem 4.6*: in fact, by that hypothesis, a representative literal coming from the translation of the diagram of some substructure $\mathcal{N}$ of $\mathcal{M}$ having cardinality at most M and mentioning a subset $\underline{i}_0 \subseteq_M \underline{j}$ included in the support of $\mathcal{N}$[12] must be

[10] Recall that only the root predicate of an atom is relevant for it to be a representative literal, hence manipulations on terms and subterms do not affect the property of being a representative literal.

[11] Notice that $\mathcal{M}[\underline{j}]$ is a model of W, because W is universal, hence its models are closed under taking substructures.

[12] Such literal could represent a potential problem: it might have been extracted when making the selection of literals from the matrix of the formula (11) because it mentions a subset $\underline{i}_0$ of the $\underline{j}$ without however coming from a diagram of a substructure of $\mathcal{M}[\underline{j}]$ (it might come from the diagram of some substructure $\mathcal{N} := \mathcal{M}[\underline{j}']$ with $\underline{i}_0 \subseteq \underline{j}'$).

A_I^E-entailed by $\delta^*_{\mathcal{M}[\underline{i}_0]}$ (notice that the representative literals

$$\delta^*_{\mathcal{M}[\underline{i}_0]}(\underline{i}_0, t_{1f(i_{l_1})}(i_{l_1}, a[i_{l_1}]), \dots, t_{sf(i_{l_{M'}})}(i_{l_{M'}}, a[i_{l_{M'}}]))$$

are in (13) and we are entitled, by Lemma B.3, to replace $a_1[i_{l_1}], \dots, a_s[i_{l_{M'}}]$ by the terms $t_{1f(i_{l_1})}, \dots, t_{sf(i_{l_{M'}})}$ in such A_I^E-entailment).

The claim (*) is proved; we now complete also the proof of the main Theorem 4.6. Because of the claim, there are Σ_W-formulae

$$\alpha_0(\underline{i}_0), \alpha_1(\underline{i}_1), \dots, \alpha_n(\underline{i}_n), \dots$$

such that the states that can reach a state satisfying U in n-steps are described by the $\exists^I$-formulae $\exists \underline{i}_n\, \alpha^*_n(\underline{i}_n, a[\underline{i}_n])$. Let $\exists \underline{k}_n \beta_n$ be $\exists \underline{i}_0 \cdots \underline{i}_n(\alpha_0 \vee \cdots \vee \alpha_n)$;[13] if we succeed in showing that $W \models \exists \underline{k}_{n+1}\beta_{n+1} \to \exists \underline{k}_n \beta_n$ holds for some n, then the theorem is proved (because we have $A_I^E \models \exists \underline{k}_{n+1}\beta^*_{n+1} \to \exists \underline{k}_n \beta^*_n$, i.e. $A_I^E \models \exists \underline{k}_{n+1}\beta^*_{n+1} \leftrightarrow \exists \underline{k}_n \beta^*_n$ and backward search halts at step $n+1$). Suppose, for reductio, that $W \not\models \exists \underline{k}_{n+1}\beta_{n+1} \to \exists \underline{k}_n \beta_n$ holds for all n; then for every $n > 1$ there exists a model $\mathcal{M}_n$ of W such that $\mathcal{M}_n \models \exists \underline{k}_n \beta_n$ and $\mathcal{M}_n \not\models \exists \underline{k}_{n-1}\beta_{n-1}$ (thus also $\mathcal{M}_n \not\models \exists \underline{k}_m \beta_m$ for all $m < n$). Since the theory W is universal, we can assume $\mathcal{M}_n$ to be finitely generated (i.e. finite being Σ_W relational) just by restricting $\mathcal{M}_n$ to the substructure generated by the $\underline{k}_n$-tuple satisfying $\beta_n(\underline{k}_n)$. Since W is a wqo theory, there are $m < n$ such that $\mathcal{M}_m$ embeds into $\mathcal{M}_n$: however this is impossible, because the formula $\exists \underline{k}_m \beta_m$ is existential, it is true in $\mathcal{M}_m$ and false in $\mathcal{M}_n$. □

A side remark: it is not clear whether the hypothesis (ii) of Theorem 4.6 can be eliminated (probably it cannot). It could be eliminated in presence of an extra condition on the translation, which we do not want to assume in the general case, because it could be non-effective. Such extra condition could be for instance formulated in the following way: if $\underline{i}_0 \subseteq \underline{j}$, then for all α, ψ we have that

$$A_I^E \models \alpha^*(\underline{j}) \to \psi(\underline{i}_0, a[\underline{i}_0]) \quad \text{implies} \quad A_I^E \models \beta^*(\underline{i}_0) \to \psi(\underline{i}_0, a[\underline{i}_0]) \tag{14}$$

for some $\beta(\underline{i}_0)$ such that $W \models \alpha(\underline{j}) \to \beta(\underline{i}_0)$. Instead of assuming (14) in the general case, we assume in Theorem 4.6(ii) just a sufficient special case (where α is a diagram of a structure of cardinality up to M, ψ is a certain representative literal and β is chosen to be the diagram of a substructure).

[13] Bound variables renamings are applied if the $\underline{i}_0, \dots, \underline{i}_n$ are not disjoint.