Model Checking Real-Time Properties of Symmetric Systems *

E. Allen Emerson and Richard J. Trefler

Computer Sciences Department and
Computer Engineering Research Center
University of Texas, Austin, TX, 78712, USA
http://www.cs.utexas.edu/users/emerson/

Abstract. We develop efficient algorithms for model checking quantitative properties of symmetric reactive systems in the general framework of a Real-Time Mu-calculus. Previous work has been limited to qualitative correctness properties. Our work not only permits handling of quantitative correctness, but it provides a strictly more expressive framework for qualitative correctness since the Mu-calculus strictly subsumes, e.g, CTL*. Unlike the previous "group-theoretic" approaches of [CE96] and [ES96] and the technical "automata-theoretic" approach of [ES97], our new approach may be viewed as "model-theoretic".

1 Introduction

Model checking [CE81] (c.f. [QS82], [LP85]) is an algorithmic method for determining whether a given finite state system M satisfies a temporal logic specification f. Lichtenstein and Pnueli [LP85] argued that in practice the complexity of model checking will be dominated by $|M|$, the size of M. Unfortunately, $|M|$ can be of size exponential in the program text. For example, a system with n processes running in parallel, each having just 3 local states, can have 3^n global states.

Symmetry reduction is a technique designed to substantially ameliorate this state explosion problem by exploiting the fact that many such systems are symmetric in their design and operation (cf. [JR91], [ID96], [ES96], [CE96], [ES97], [GS97]). Symmetry is a form of redundancy that can be factored out. Many synchronization and coordination protocols are the parallel composition of n processes which are identical up to renaming. The state graph M of such a system may reflect considerable symmetry. For example, states (C_1, T_2) and (T_1, C_2) may be present in a solution to the mutex problem. By clustering together such symmetry equivalent states, we can form the symmetry reduced quotient structure $\hat{M}$. $\hat{M}$, whose states are named by representatives of the clusters, may be exponentially smaller than M. Then the temporal formula f may be model

* The authors' work was supported in part by NSF grants CCR-941-5496 and CCR-980-4736 and SRC contract 97-DP-388. The authors can be reached at {**emerson,trefler**}**@cs.utexas.edu**

checked over $\hat{M}$ to determine if f holds of M. In practice, $\hat{M}$ is typically constructed incrementally from the program text, avoiding the self defeating task of first building M.

Work on symmetry reduction in model checking originally reduced M to an 'unannotated' symmetry reduced quotient structure $\overline{M}$ [ES96], [CE96]. However, that work, due to certain technical provisos regarding the internal symmetry of the specifications, was unable to handle fairness despite otherwise catering for CTL*. To remedy this, [ES97] introduced the annotated quotient structure $\hat{M}$ where the transitions between representative states are labeled with permutations indicating how the meaning of all coordinates shift from representative to representative. [ES97] also introduced a threaded quotient structure M^* indicating how the meaning of individual coordinates shift. By combining automata with these quotient graphs in an automata-theoretic [VW86] treatment, [ES97] developed a technical approach that allowed fairness properties to be checked efficiently.

In this paper we investigate model checking quantitative, discrete real-time properties over the quotients $\hat{M}$ and M^* in the framework of the Real-Time μ-calculus (RTLμ) (c.f. [Ko83], [Em92], [Se96]) which strictly subsumes the logics considered in previous work. We define a new notion of "twisted truth" or permuted satisfaction of a formula over annotated structures, $\hat{M}, \hat{s} \models\!\!\!\equiv f$, and prove that this permuted truth corresponds to the usual one over unannotated structures $M, \hat{s} \models f$, that is $\hat{M}, \hat{s} \models\!\!\!\equiv f$ iff $M, \hat{s} \models f$. This new notion leads to an efficient model checking algorithm for a formulation of an Indexed Real-Time Mu-calculus, IRTLμ. In particular, we give an $O(|\hat{M}||f|n)$ algorithm, which actually operates on M^*, for evaluating IRTLμ formulae of alternation depth 1 over $\hat{M}$. This algorithm can be generalized to work on arbitrary formulae of the μ-calculus. Our treatment of these problems, providing an alternative means of handling fairness properties, is done without appeal to automata. Instead, our techniques show how expressive model checking over the annotated quotient structure can be accomplished in a model-theoretic framework.

Interestingly, quantitative temporal properties of the structure M are preserved in $\hat{M}$ even though $\hat{M}$ may be exponentially smaller than M. For example, if the number of states of $\hat{M} < k <$ the number of states of M, then checking for the existence of a path no longer than k steps to a state where symmetric assertion P is true takes time proportional to k in M but proportional to the size of $\hat{M}$ in the symmetry reduced structure. This is not so for arbitrary boolean assertions f and is complicated in the annotated $\hat{M}$ by the shifting meaning of coordinates. A subtlety that arises is the fact that cycles in the annotated quotient may not correspond to cycles in the original structure. The extent to which this subtlety must be clarified in order to solve the model checking problem is a key issue in this paper.

Finally, we present results which relate to the difficulty of model checking temporal formulae of symmetric systems. We show that model checking certain temporal modalities over annotated structures is NP-hard. Furthermore, the model checking problem for certain quantitatively bounded fairness problems is NP-hard even over unreduced structures, in contrast with the polynomial algorithms for checking unbounded fairness. Against the background of these somewhat negative results we identify some classes of formulae and structures for which symmetry can reduce NP-hard problems to problems which can be solved in polynomial time.

The rest of the paper is organized as follows. In section 2 the logics discussed in the paper are introduced. Section 3 discusses the definition of symmetry reduced structures. The precise correspondence between structures, their symmetry reduced annotated quotient structures and temporal formulae is given in section 4. An IRTLμ model checking algorithm is given in section 5. Section 6 discusses the complexity of model checking symmetric structures and some applications of reasoning about symmetry in structures. Finally, section 7 contains a short conclusion.

2 RTLμ

Let LP denote a finite set of local propositions. $\mathcal{I}$ denotes an index set $[1..n]$ for some $n \in \mathbf{N}$ which denotes the set of natural numbers.

RTLμ is formed thusly: the set of atomic propositions is $LP \times \mathcal{I}$; we will write $(P, i) \in LP \times \mathcal{I}$ as P_i. We assume a set of variables over sets of states $\mathrm{Var} = \mathrm{V} \cup (\mathrm{V}' \times \mathcal{I})$, where V and V$'$ are unindexed and disjoint. RTLμ is the set of formulae defined by $P_i \in LP \times \mathcal{I}$ and $Y, Y_i \in \mathrm{Var}$ are atomic formulae; if f and g are formulae then so are $\neg f$, $f \wedge g$ and $\langle R \rangle f$. Finally, suppose $f(Y)$ is a formula syntactically monotone in Y, $Y \in \mathrm{Var}$, that is, every occurrence of Y falls within an even number of $\neg$ occurrences, then $\mu Y.f(Y)$ and $\mu kY.f(Y)$, $k \in \mathbf{N}$, are formulae. $\nu Y.f(Y)$ and $\nu kY.f(Y)$ are abbreviations for $\neg\mu Y.\neg f(\neg Y)$ and $\neg\mu kY.\neg f(\neg Y)$ respectively and $[R]f$ is an abbreviation for $\neg\langle R \rangle \neg f$.

Let iRTLμ be the sub-logic whose atomic formulae are propositions in $LP \times \{i\}$ and variables in V and $\mathrm{V}' \times \{i\}$. Then the indexed mu-calculus, IRTLμ, is the logic whose atomic formulae are variables in V and formulae of the form $\vee_i f_i$ or $\wedge_i f_i$ where the f_i are isomorphic formulae of iRTLμ. Formulae of IRTLμ are also formed from the connectives $\wedge, \neg, \langle R \rangle$, μY and μkY. Furthermore, if $f(Y)$ and g are formulae of IRTLμ, where Y is an unindexed variable not appearing within any μY in f, then the formula $f(g)$, which is obtained by replacing each of occurrence of Y in $f(Y)$ by g is a formula of IRTLμ.

Formulae of RTLμ are given semantics relative to finite structures $M = (S, R)$ where $S \subseteq LP^{\mathcal{I}}$ and $R \subseteq S \times S$. For $s \in S$ we say that $P_i \in s$ iff the ith element of s is $P \in LP$. Given formula f and structure M, the meaning of f, written f^M, is a mapping from valuations, $\sigma \in [\mathrm{Var} \rightarrow 2^S]$, into 2^S and is defined below. We say that state s in M satisfies f if $s \in f^M$ (alternatively written $M, s \models f$).

- For atomic formulae, $P_i^M(\sigma) = \{s \in S \mid P_i \in s\}$ and $Y^M(\sigma) = \sigma(Y)$.
- For boolean combinations, $(f \wedge g)^M(\sigma) = f^M(\sigma) \cap g^M(\sigma)$ and $(\neg f)^M(\sigma) = S \setminus f^M(\sigma)$.
- $(\langle R \rangle f)^M(\sigma) = \{s \in S \mid \exists t \in S,\ t \in f^M(\sigma) \text{ and } (s, t) \in R\}$.

– $(\mu Y.f(Y))^M(\sigma) = \cap\{S' \subseteq S \mid (f(Y))^M(\sigma[S'/Y]) = S'\}$.
$(\mu 0Y.f(Y))^M(\sigma) = \emptyset$.
$(\mu(k+1)Y.f(Y))^M(\sigma) = (f(Y))^M(\sigma[(\mu kY.f(Y))^M(\sigma)/Y])$.

For valuation σ, $\sigma[A/Y]$ is the valuation everywhere equal to σ except that $\sigma[A/Y](Y) = A$. We say that variable Y is free (not bound) in formula f if it does not occur syntactically within a $\mu Y.$ operator. A sentence is a formula with no free variables and we assume that all bound variables are bound uniquely in sentences. Altdepth(f), is the alternation depth of the formula f, that is, 1 + the depth of nesting of alternating μ's and ν's when f is put in positive normal form.

We will at times make use of Computation Tree Logic (CTL) [CE81] and CTL* [EH86] formulae to explicate results. Both logics use the universal A and existential E path quantifiers together with the standard X, F, G and U temporal operators over paths. CTL is restricted to formulae where each path quantifier is matched with a single path operator while CTL* allows arbitrary nesting and boolean combinations of temporal operators over paths to be combined with a single path quantifier. It is known that CTL formulae can be seen as simple macros for μ-calculus formulae of alternation depth 1, while CTL* can be translated into the μ-calculus but with an exponential blowup in formula length. We note that the CTL* fairness formula $\mathsf{E}(\wedge_i \mathsf{GF} P_i)$, which says that there is a path along which for each i, P_i is satisfied infinitely often, can be expressed in RTLμ as $\nu Z. \wedge_i \mathsf{EXEF}(P_i \wedge Z)$ which is expressed in IRTLμ as $\nu Z. \wedge_i (\mu Y_i.\langle R\rangle((P_i \wedge Z) \vee Y_i))$.

3 Symmetry of Structures

$Sym\ \mathcal{I} = \{\pi \mid \pi$ is a permutation on $\mathcal{I}\}$. $Sym\ \mathcal{I}$ together with the function composition operator, $\circ$, is a group, where the inverse of π is denoted by π^{-1}. Given state s and permutation π, we say that π acts on s, written $\pi(s)$, in the following way $\pi(s) = \{P_{\pi(j)} \in LP \times \mathcal{I} \mid P_j \in s\}$. For example, let $s = (C_1, T_2)$ and π be the permutation which flips 1 and 2. Then $\pi(s) = (T_1, C_2)$. Similarly for structures, $\pi(M)$, π acting on M, is the structure (S', R') where $S' = \{\pi(s) \in LP^{\mathcal{I}} \mid s \in S\}$ and $\pi(s) \to \pi(t) \in R'$ iff $s \to t \in R$. Then π is an automorphism of M if $\pi(M) = M$ and $Aut(M)$ is the set $\pi \in Sym\ \mathcal{I}$ such that π is an automorphism of M. $Aut(M)$ is a subgroup of $Sym\ \mathcal{I}$ which will be denoted by $Aut(M) \preceq Sym\ \mathcal{I}$.

Let f be a formula of RTLμ, $\pi(f)$ is the formula g which is identical to f except that each occurrence of $P_i \in LP \times \mathcal{I}$ ($Y_i \in \mathrm{V}' \times \mathcal{I}$)is replaced by $P_{\pi(i)}$ ($Y_{\pi(i)}$). Then $Aut(f) = \{\pi \in Sym\ \mathcal{I} \mid \pi(f) \equiv f\}$ [ES96]. $Auto(f)$ [1] is the set of $\pi \in Sym\ \mathcal{I}$ such that for each maximal propositional sub-formula, g, of f, $\pi(g) \equiv g$. Both $Aut(f)$ and $Auto(f)$ are subgroups of $Sym\ \mathcal{I}$ and $Auto(f) \preceq Aut(f)$.

[1] A more general definition of $Auto(f)$ can be found in [ES96].

Given structure $M = (S, R)$, let G be a subgroup of $Aut(M)$. Two states, $s, s' \in S$ are equivalent with respect to G, written $s \equiv_G s'$, if there exists $\pi \in G$ such that $\pi(s) = s'$ [ES96], [CE96]. Because G is a (sub)group it is clear that $\equiv_G$ is an equivalence relation. Then for each equivalence class in $\equiv_G$ we choose an arbitrary member state to represent that class and refer to that state as $\overline{s}$ and its equivalence class as $[\overline{s}]$. That is, given $\overline{s} \in S$, $\overline{s}$ is the representative of $[\overline{s}] = \{t \in S \mid t \equiv_G \overline{s}\}$. $\overline{M} = M/\equiv_G = (\overline{S}, \overline{R})$, is the unannotated quotient structure where $\overline{S}$ is the set of $\overline{s}$ such that $\overline{s}$ is the representative of an equivalence class of $\equiv_G$. $\overline{R} \subseteq \overline{S} \times \overline{S}$ is the set of transitions $(\overline{s}, \overline{t})$ such that there exists $s' \equiv_G \overline{s}$, $t' \equiv_G \overline{t}$ and $s' \rightarrow t' \in R$.

$\hat{M} = M/\equiv_G = (\hat{S}, \hat{R})$, is the annotated quotient structure where $\hat{S}$ is the set of $\hat{s}$ such that, as above, $\hat{s}$ is the representative of an equivalence class of $\equiv_G$. $\hat{R} \subseteq \hat{S} \times G \times \hat{S}$ is defined by the restrictions : (i) if $\hat{s} \rightarrow t \in R$ then there is a unique π such that $\hat{s} \xrightarrow{\pi} \hat{t} \in \hat{R}$ and $\pi(\hat{t}) = t$; (ii) $\hat{s} \xrightarrow{\pi} \hat{t} \in \hat{R}$ only if $\hat{s} \rightarrow \pi(\hat{t}) \in R$.

Model checking for RTLμ could be carried out on either $\hat{M}$ of M^*. We choose to make use of M^* which can be seen as a data structure which, for a modest increase in the number of states in $\hat{M}$, can help organize the model checking algorithm. However, the fact that the number of states in M^* is larger, by a factor of $|\mathcal{I}|$, over $\hat{M}$ can be misleading. Although it may be unusual two states $\hat{s}$ and $\hat{t}$ may have an exponential number of labeled arcs between them in $\hat{M}$ but in M^* those same two states will have at most a quadratic number of arcs between them. Technically $M^* = (\hat{S} \times (\{0\} \cup \mathcal{I}), R^*, RED)$ where R^* and RED are transition relations. For $i, j \in \mathcal{I}$, $\langle \hat{s}, i\rangle \rightarrow \langle \hat{t}, j\rangle \in R^*$ iff there exists π such that $\hat{s} \xrightarrow{\pi} \hat{t} \in \hat{R}$ and $\pi^{-1}(i) = j$. For $i \in \mathcal{I}$, $\langle \hat{s}, 0\rangle \rightarrow \langle \hat{s}, i\rangle \in RED$ and $\langle \hat{s}, i\rangle \rightarrow \langle \hat{s}, 0\rangle \in RED$. Finally, $\langle \hat{s}, 0\rangle \rightarrow \langle \hat{t}, 0\rangle \in R^*$ iff there is a π such that $\hat{s} \xrightarrow{\pi} \hat{t} \in \hat{R}$.

4 Temporal Formulae on Annotated Structures

We define the meaning of an RTLμ formula, f, on an annotated structure $\hat{M}$. We say that state $\hat{s}$ in $\hat{M}$ satisfies f if $\hat{s} \in f^{\hat{M}}$ (alternatively written $\hat{M}, \hat{s} \models f$). For the purposes of this definition $\hat{M}$ could, in general, be any annotated structure and need not correspond to the symmetry reduced quotient of any particular unannotated structure. $f^{\hat{M}} : [\mathrm{Var} \rightarrow 2^{\hat{S}}] \rightarrow 2^{\hat{S}}$ and valuation $\hat{\sigma} \in [\mathrm{Var} \rightarrow 2^{\hat{S}}]$.

- For P_i, $P_i^{\hat{M}}(\hat{\sigma})$ is the set of states $\hat{s} \in \hat{S}$ such that $P_i \in \hat{s}$. For $Y \in \mathrm{Var}$, $Y^{\hat{M}}(\hat{\sigma})$ is the set of states $\hat{s} \in \hat{S}$ such that $\hat{s} \in \hat{\sigma}(Y)$.
- $(f \wedge g)^{\hat{M}}(\hat{\sigma}) = f^{\hat{M}}(\hat{\sigma}) \cap g^{\hat{M}}(\hat{\sigma})$.
  $(\neg f)^{\hat{M}}(\hat{\sigma}) = \hat{S} \setminus f^{\hat{M}}(\hat{\sigma})$.
- $(\langle \hat{R}\rangle f)^{\hat{M}}(\hat{\sigma}) = \{\hat{s} \in \hat{S} \mid \text{there exists } \hat{s} \xrightarrow{\pi} \hat{t} \in \hat{R} \text{ and } \hat{t} \in (\pi^{-1}(f))^{\hat{M}}(\hat{\sigma})\}$.
- $(\mu Y.f(Y))^{\hat{M}}(\hat{\sigma}) = \cap\{\hat{S}' \subseteq \hat{S} \mid (f(Y))^{\hat{M}}(\hat{\sigma}[\hat{S}'/Y]) = \hat{S}'\}$.
  $(\mu 0 Y.f(Y))^{\hat{M}}(\hat{\sigma}) = \emptyset$.
  $(\mu(k+1)Y.f(Y))^{\hat{M}}(\hat{\sigma}) = f(Y)^{\hat{M}}(\hat{\sigma}[(\mu k Y.f(Y))^{\hat{M}}(\hat{\sigma})/Y])$.

The following theorem relates the meaning of RTLμ sentence f over a structure M and the meaning of f over the annotated structure $\hat{M}$. For $\sigma \in [\mathrm{Var} \to 2^S]$ and $\hat{\psi} \in [\mathrm{Var} \to 2^{\hat{S}}]$ we say that σ and $\hat{\psi}$ correspond iff $\sigma(Y) = \{t \mid t \equiv_G \hat{s},$ for some $\hat{s} \in \hat{\psi}(Y)\}$.

Theorem 1. *Let $\hat{M} = M/\equiv_G$, for $G \preceq Aut(M)$, and let σ and $\hat{\psi}$ be corresponding valuations. Then for any* RTLμ *sentence f, $\hat{s} \in f^M(\sigma)$ iff $\hat{s} \in f^{\hat{M}}(\hat{\psi})$.*

5 Real-Time Mu-Calculus Model Checking

We show how to reduce the problem of model checking IRTLμ formula f over $\hat{M}$ to the problem of checking the transformed formulae $\mathcal{T}(f)$, over the threaded structure M^*. This reduction implies an algorithm for model checking f over M, by checking $\mathcal{T}(f)$ over M^*.

We proceed as follows, firstly, we define a translation from formulae of IRTLμ over $LP \times \mathcal{I}$, $\mathrm{V} \cup (\mathrm{V}' \times \mathcal{I})$ and $\langle R \rangle$ to formulae of RTLμ over LP, V, V$'$, $\langle R \rangle$ and $\langle RED \rangle$. The intuition behind this transformation is that the states of M^*, which are of the form $\langle \hat{s}, i \rangle$, only record the satisfaction of propositions P_i which are true at $\hat{s}$. Therefore the subscript i can be dropped in these 'local' states. We then use the state $\langle \hat{s}, 0 \rangle$ as a 'global' state to collect information about all the $\langle \hat{s}, i \rangle$'s. It is then possible to trade the universal quantification over i in formulae of the form $\wedge_i f_i$ for a modal operator $[RED]$ at state $\langle \hat{s}, 0 \rangle$ and check that all the $\langle \hat{s}, i \rangle$'s satisfy f. We then model check the transformed formulae over the structure M^* [EL86]. Since M^* is an unannotated structure, μ-calculus model checking algorithms may be applied directly to the problem of checking whether the transformed formula is satisfied by M^*. For the purposes of this model checking we define the meaning of $P \in LP$ over M^* as follows $P^{M^*}(\psi) = \{\langle \hat{s}, i \rangle \mid i \neq 0$ and $P_i \in \hat{s}\}$. The meaning of a compound formula or variable is defined by its standard meaning as given in Section 2.

Technically, we distinguish between global and local IRTLμ formulae. Global formulae are those where all indexed propositions and variables appear within the scope of an $\wedge_i$ or $\vee_i$ quantifier. Local formulae have at least one indexed proposition or variable which does not appear within the scope of any $\wedge_i$ or $\vee_i$. Then for formula, f, of IRTLμ we define the transform of f, $\mathcal{T}(f)$ as follows.

- $\mathcal{T}(P_i) = P$. $\mathcal{T}(Y_i) = Y$. $\mathcal{T}(Z) = Z$.
- For f and g both local or both global: $\mathcal{T}(f \wedge g) = \mathcal{T}(f) \wedge \mathcal{T}(g)$.
  $\mathcal{T}(f \vee g) = \mathcal{T}(f) \vee \mathcal{T}(g)$.
- For f global and g_i local: $\mathcal{T}(f \wedge g_i) = (\langle RED \rangle \mathcal{T}(f)) \wedge \mathcal{T}(g_i)$.
  $\mathcal{T}(f \vee g_i) = (\langle RED \rangle \mathcal{T}(f)) \vee \mathcal{T}(g_i)$.
- $\mathcal{T}(\neg f) = \neg \mathcal{T}(f)$.
- $\mathcal{T}(\wedge_i f_i) = [RED]\mathcal{T}(f_1)$. Because the f_i's are isomorphic, $\mathcal{T}(f_1) = \mathcal{T}(f_2)$, we need only check for $\mathcal{T}(f_1)$.
- $\mathcal{T}(\vee_i f_i) = \langle RED \rangle \mathcal{T}(f_1)$.
- $\mathcal{T}(\mu Z.f(Z)) = \mu Z.\mathcal{T}(f(Z))$.
  $\mathcal{T}(\mu k Z.f(Z)) = \mu k Z.\mathcal{T}(f(Z))$.

- $\mathcal{T}(\mu Y_i.f(Y_i)) = \mu Y.\mathcal{T}(f(Y))$.
  $\mathcal{T}(\mu k Y_i.f(Y_i)) = \mu k Y.\mathcal{T}(f(Y))$.

The idea behind $\mathcal{T}(\wedge_i f_i) = [RED]\mathcal{T}(f_1)$ is that $\hat{s}$ satisfies $\wedge_i f_i$ in $\hat{M}$ iff for all $i \in \mathcal{I}$, $\langle \hat{s}, i\rangle$ satisfies f in M^* and we use $\langle \hat{s}, 0\rangle$ to check whether in fact this is the case. Recall from the definition of M^* that the only transitions in RED are from $\langle \hat{s}, 0\rangle$ to $\langle \hat{s}, i\rangle$ and vice versa.

Let $\hat{\sigma}$ be a valuation over $\hat{M}$ and ψ be a valuation over M^*. Then we say that $\hat{\sigma}$ and ψ correspond when for global variable Z, $\psi(Z) = \{\langle \hat{s}, 0\rangle \mid \hat{s} \in \hat{\sigma}(Z)\}$ and for local variable $Y \in \mathrm{V}'$, $\psi(Y) = \{\langle \hat{s}, i\rangle \mid \hat{s} \in \hat{\sigma}(Y_i)\}$.

Proposition 2. *Let f be a global formula of* IRTLμ *while $\hat{\sigma}$ and ψ are corresponding valuations. Then $\hat{s} \in f^{\hat{M}}(\hat{\sigma})$ iff $\langle \hat{s}, 0\rangle \in (\mathcal{T}(f))^{M^*}(\psi)$.*

Let f_i be a local formula of IRTLμ *while $\hat{\sigma}$ and ψ are corresponding valuations. Then $\hat{s} \in f_i^{\hat{M}}(\hat{\sigma})$ iff $\langle \hat{s}, i\rangle \in (\mathcal{T}(f))^{M^*}(\psi)$.*

Theorem 3. *For global sentence f of* IRTLμ *, $\mathcal{T}(f)$ can be model checked over the structure M^* in time $\mathcal{O}((|M^*||f|)^{\mathrm{altdepth}(f)})$.*

Remark : This time bound can be improved to $\mathcal{O}((|M^*||f|)^{\lfloor(\mathrm{altdepth}(f)+1)/2\rfloor})$ [LB94] and in general we may take advantage of any μ-calculus algorithm with a better time bound.

Corollary 4. *The model checking problem 'does* IRTLμ *formula f hold in M' can be solved in time $\mathcal{O}((|M^*||f|)^{\mathrm{altdepth}(f)})$.*

6 Applications

Certain problems that are in general NP-hard become solvable in polynomial time in the special case of symmetric structures. We now discuss the complexity of model checking symmetric structures and symmetric formulae. Our first results show that model checking certain basic temporal logic formulae over annotated symmetry reduced structures is NP-hard. Secondly, we show that model checking the indexed bounded fairness formula $\mathsf{E} \wedge_i \mathsf{GF}^{\leq k} P_i$ over unannotated structures is NP-hard. This implies that it is unlikely that there is a 'short' IRTLμ formula expressing this property and hence is an indication that bounded fairness may be exponentially harder to check than more standard fairness notions.

Theorem 5. *$\hat{M}, \hat{s} \models \mathsf{EF}p$ is NP-hard. $\hat{M}, \hat{s} \models \mathsf{EG}p$ is NP-hard.*

Proof Idea: Let $q(P_1, \ldots, P_n)$ be a boolean formula over the propositions P_1 through P_n. Define p as $q(Q_1, Q_3, \ldots, Q_{2n-1})$ where the $2n, Q_i$'s are fresh propositional symbols. $\hat{M}$ is the annotated structure which consists of the single state $\hat{s}$ which is labeled with the propositions $Q_1, Q_3, \ldots, Q_{2n-1}$. There are two transitions from $\hat{s}$ to $\hat{s}$, the first is labeled by the rotation permutation $(1\ 2 \ldots 2n)$ and the second by the transposition permutation $(1\ 2)$. Arbitrary composition of these two permutations is enough to create any permutation in $Sym\ 2n$. It can then be shown that $q(P_1, \ldots, P_n)$ is satisfiable iff $\hat{M}, \hat{s} \models \mathsf{EF}p$. □

Theorem 6. *$M, s \models \mathsf{E} \wedge_i \mathsf{GF}^{\leq k} P_i$ is NP-hard.*

The following theorems relate the CTL* formula $\mathsf{E}(\wedge_i \mathsf{F} P_i)$ to any equivalent translation into the μ-calculus. $\mathsf{E}(\wedge_i \mathsf{F} P_i)$ says that there is a path such that for all i, eventually P_i is true.

Theorem 7. *[SC85] (c.f. [CE81]) $M, s \models \mathsf{E}(\wedge_i \mathsf{F} P_i)$ is NP-complete.*

However, when $Aut(M) = Aut(s) = Sym\ \mathcal{I}$, where $Aut(s) = \{\pi \in Sym\ \mathcal{I} \mid \pi(s) = s\}$, then the model checking problem for $\mathsf{E}(\wedge_i \mathsf{F} P_i)$ can be solved efficiently.

Proposition 8. *$M, s \models [\mathsf{E}(\wedge_i \mathsf{F} P_i)]$ iff $M, s \models \vee_{\pi \in Sym\ \mathcal{I}}\ \mathsf{EF}(P_{\pi(1)} \wedge \mathsf{EF}(P_{\pi(2)} \wedge \ldots \wedge \mathsf{EF} P_{\pi(n)}))$*

Theorem 9. *$Aut(M) = Aut(s) = Sym\ \mathcal{I}$ implies that $M, s \models \mathsf{E}(\wedge_i \mathsf{F} P_i)$ iff $M, s \models \mathsf{EF}(P_1 \wedge \mathsf{EF}(P_2 \wedge \ldots \wedge \mathsf{EF} P_n))$.*

Proof idea: Right to left follows from the existence of a path through each of the P_i's. Suppose s satisfies $\mathsf{E}(\wedge_i \mathsf{F} P_i)$ then s also satisfies $\mathsf{EF}(P_{\pi(1)} \wedge \mathsf{EF}(P_{\pi(2)} \wedge \ldots \wedge \mathsf{EF} P_{\pi(n)}))$ for some π. By state symmetry [ES96] s also satisfies $\mathsf{EF}(P_1 \wedge \mathsf{EF}(P_2 \wedge \ldots \wedge \mathsf{EF} P_n))$. □

Because $\mathsf{EF}(P_1 \wedge \mathsf{EF}(P_2 \wedge \ldots \wedge \mathsf{EF} P_n))$ is a CTL formula and can be translated into a μ-calculus formula of alternation depth 1 it can be model checked on a structure M in time linear in the size of the structure and the formula as opposed to the presumed exponential time algorithm for model checking $\mathsf{E} \wedge_i \mathsf{F} P_i$.

We can extend this reasoning as follows.

Theorem 10. *Suppose $M, s \models \mathsf{AGEF} s$ and $Aut(s) = Aut(M) = Sym\ \mathcal{I}$. Then $M, s \models \mathsf{E}(\wedge_i \mathsf{F} P_i)$ iff $M, s \models \wedge_i \mathsf{EF} P_i$ iff $M, s \models \mathsf{EF} P_1$.*

The point being that $\wedge_i \mathsf{EF} P_i$ can be translated into an IRTLμ formula of alternation depth 1 and hence can be model checked on the symmetry reduced structure $\hat{M}$ quickly where as it seems that $\mathsf{EF}(P_1 \wedge \mathsf{EF}(P_2 \wedge \ldots \wedge \mathsf{EF} P_n))$ cannot be. It is, in general, interesting to consider the classes of linear time formula h_i and structures M for which s satisfies $\mathsf{E} \wedge_i h_i$ is equivalent to s satisfies $\wedge_i \mathsf{E} h_i$ because the latter formula can be checked much more quickly on both the large and the symmetry reduced structures.

7 Conclusion

This paper has described a general framework for performing model checking for formulae of the μ-calculus on symmetric systems. We have given efficient model checking algorithms for indexed sub-logics of the Real-Time μ-calculus over annotated structures. These real-time logics are useful for describing the quantitative and qualitative properties of a large class of programs that operate in real-time environments, such as network communication protocols and embedded real-time control systems. Furthermore, our framework subsumes indexed formulations of RTCTL [EM92] and CTL* [ES97].

We have also shown that the threaded graph construction of [ES97], used in a different form in [ES96], is more general and thus more applicable than previously thought. M^* supports general μ-calculus model checking. But that leaves the question, 'where did the automata go?' The answer is that M^* may be viewed as an automaton of a particularly simple nature, one whose job it is to steadily keep track of shifting indices. We remark that for checking fairness our method requires essentially quadratic time in $|\hat{M}|$ for weak fairness versus linear time in $|\hat{M}|$ for [ES97]; but this is an artifact of using the more general μ-calculus of alternation depth 2 (c.f. [EL86] vs [EL87]).

The work presented here deals with quantitative, discrete real-time logics. These logics are exponentially more succinct but not strictly more expressive than their untimed counterparts. An interesting area for further research is reasoning about symmetry on explicitly timed structures which model dense or discrete time as discussed in [AC90], [Al91] and [He91]. We have also identified an interesting open problem in the realm of model checking symmetric structures, that is to fully characterize the relationship between formulae of the form $\wedge_i \mathsf{E} h_i$ and $\mathsf{E}(\wedge_i h_i)$ over symmetric structures.

References

[AC90] Alur, R., Courcoubetis, C., and Dill, D., Model Checking for Real-Time Systems. In *Proceedings of the Fifth Annual Symposium on Logic in Computer Science*, pp. 414-425, IEEE Computer Society Press, 1990.

[Al91] Alur, R., *Techniques for Automatic Verification of Real-Time Systems.* PhD thesis, Stanford University, 1991.

[CE81] Clarke, E. M., and Emerson, E. A., Design and Verification of Synchronization Skeletons using Branching Time Temporal Logic, Logics of Programs Workshop, IBM Yorktown Heights, New York, Springer LNCS no. 131., pp. 52-71, May 1981.

[CE96] Clarke, E. M., Filkorn, T., and Jha, S., Exploiting Symmetry in Temporal Logic Model Checking. In *Fifth International Conference on Computer Aided Verification*, Crete, Greece, June 1993. Journal version appears as: Clarke, E. M., Enders, R. Filkorn, T. and Jha, S., Exploiting Symmetry in Temporal Logic Model Checking. In *Formal Methods in System Design*, Kluwer, vol. 9, no. 1/2, August 1996.

[Em92] E. Allen Emerson Real–Time and the μ–Calculus. In *Proceedings of Real-Time: Theory in Practice*, LNCS, Vol. 600, pp. 176-194, Springer, June 1992.

[EH86] Emerson, E. A., and Halpern, J. Y., 'Sometimes' and 'Not Never' Revisited: On Branching versus Linear Time Temporal Logic, *JACM*, vol. 33, no. 1, pp. 151-178, Jan. 86.

[EL86] Emerson, E. A., and Lei, C.-L., Efficient Model Checking in Fragments of the Mu-Calculus, IEEE Symp. on Logic in Computer Science (LICS), Cambridge, Mass., 1986.

[EL87] Emerson, E. A., and Lei, C.-L.m Modalities for Model Checking: Branching Time Strikes Back, pp. 84-96, ACM POPL85; journal version appears in Sci. Comp. Prog. vol. 8, pp 275-306, 1987.

[EM92] Emerson, E. A., Mok, A. K., Sistla, A. P., and Srinivasan, J., Quantitative Temporal Reasoning. In *Journal of Real Time Systems*, vol. 4, pp. 331-352, 1992.

[ES96] Emerson, E. A. and Sistla, A. P., Symmetry and Model Checking. In *Fifth International Conference on Computer Aided Verification*, Crete, Greece, June 1993. Journal Version appeared in *Formal Methods in System Design*, Kluwer, vol. 9, no. 1/2, August 1996.

[ES97] Emerson, E. A. and Sistla, A. P., Utilizing Symmetry when Model Checking under Fairness Assumptions. In *Seventh International Conference on Computer Aided Verification* Springer-Verlag, 1995. Journal version, *TOPLAS* 19(4): 617-638 (1997).

[GS97] Gyuris, V. and Sistla, A. P., On-the-Fly Model checking under Fairness that Exploits Symmetry. In *Proceedings of the 9th International Conference on Computer Aided Verification, Haifa, Israel*, 1997.

[He91] Henzinger, T., The Temporal Specification and Verification of Real-Time Systems, Ph.D. Thesis, Stanford University, 1991, report number STAN-CS-91-1380.

[ID96] Ip, C-W. N., Dill, D. L., Better Verification through Symmetry. In *Proc. 11th International Symposium on Computer Hardware Description Languages(CHDL)*, April, 1993. Journal version appeared in *Formal Methods in System Design*, Kluwer, vol. 9, no. 1/2, August 1996.

[JR91] Jensen, K. and Rozenberg, G. (eds.), High-Level Petri Nets: Theory and Application, Springer- Verlag, 1991.

[Ko83] Kozen, D., Results on the Propositional Mu-Calculus, Theor. Comp. Sci., pp. 333-354, Dec. 83.

[LP85] Litchtenstein, O., and Pnueli, A., Checking That Finite State Concurrent Programs Satisfy Their Linear Specifications, POPL85, pp. 97-107, Jan. 85.

[LB94] Long, D., Browne, A., Clarke, E. Jha, S. and Marrero, W., An Improved Algorithm for the Evaluation of Fixpoint Expressions. In *Proc. of the 6th Inter. Conf. on Computer Aided Verification, Stanford, Springer LNCS no. 818*, June 1994.

[QS82] Queille, J. P., and Sifakis, J., Specification and verification of concurrent programs in CESAR, Proc. 5th Int. Symp. Prog., Springer LNCS no. 137, pp. 195-220, 1982.

[Se96] Seidl, H., A Modal μ-Calculus for Durational Transition Systems. In *Eleventh Annual IEEE Symposium on Logic In Computer Science*, IEEE Computer Society Press, 1996.

[SC85] Sistla, A. P., and Clarke, E. M., The Complexity of Propositional Linear Temporal Logic, J. ACM, Vol. 32, No. 3, pp.733-749, 1985.

[VW86] Vardi, M., and Wolper, P. , An Automata-theoretic Approach to Automatic Program Verification, Proc. IEEE LICS, pp. 332-344, 1986.