On Reasoning about Infinite-State Systems in the Modal μ-Calculus

Henrik Reif Andersen*

Department of Computer Science
Aarhus University
Denmark

June 1993

Abstract

This paper presents a proof method for proving that infinite-state systems satisfy properties expressed in the modal μ-calculus. The method is sound and complete relative to externally proving inclusions of sets of states. The method can be seen as a recast of a tableau method due to Bradfield and Stirling following lines used by Winskel for finite-state systems. Contrary to the tableau method, it avoids the use of constants when unfolding fixed-points and it replaces the rather involved success criterion in the tableau method with simpler, local success criterions. A proof tree is now merely a means of keeping track of where possible choices are made - and can be changed - and not an essential ingredient in establishing the correctness of a proof: A proof will simply be correct when all leaves can be directly seen to be valid (possibly, however, by performing external reasoning about inclusions among sets of states). Therefore it seems to be well-suited for implementation as a tool for reasoning about concurrent systems based on their operational semantics as labelled transition systems, by, for instance, integration into existing general-purpose theorem provers.

The locality of the success criterions makes the proofs of soundness and completeness rather straightforward using two key lemmas

*Current address: Department of computer science, Building 344, Technical University of Denmark, DK-2800 Lyngby, Denmark

about the minimum and the maximum fixed-points. These proofs are supplied in an appendix.

1 Introduction

In this paper we describe a method for performing model checking on infinite-state systems in the modal μ-calculus. In contrast to the situation with finite state systems allowing more or less efficient *automatic methods*, we are in general forced to consider only *semi-automatic* or *machine-assisted* methods when considering infinite-state systems. This is an obvious fact whenever the class of models and the logic is powerful enough to encode undecidable properties, such as the Halting problem for Turing machines.

The modal μ-calculus is one such powerful logic. It is very straightforward to encode the behaviour of a Turing machine TM as an infinite-state system with states coding the internal state of the Turing machine as well as the contents of the tape; and find an assertion H, s.t.

$$\models TM(i, w) : H$$

is valid, if and only if, the Turing machine TM when started in the initial state i with tape contents w, halts. So for the general case, any hope of finding an algorithm actually deciding the model-checking problem is of course domed to failure.

We describe a general *method*, which can assist in proving that subsets of states of infinite labelled transition systems satisfy formulae in the modal μ-calculus. Success of using the method in one particular situation will depend on proper *choices* in certain steps of applying the method, and on the ability to show properties of infinite sets of states by induction. The actual inductive proof takes place as part of the method but depends on a *well-founded relation* being supplied. The undecidability can now be viewed as a combination of the impossibitity of making these choices algorithmically and of the impossibility of algorithmically supplying the 'right' well-founded relation. The method will be sound in the sense that, whenever a model is shown to satisfy an assertion of the logic, this is certainly a valid conclusion, and it will be complete in the sense that, whenever a model satisfies an assertion it is possible to make correct choices, and provide well-founded relations such that in a finite number of steps of the method this fact will be proven.

The method raises some interesting questions. One is a question of 'relative completeness', i.e. in analogy with Hoare Logic, whether proper notations for infinite sets can be found, making the method complete under the assumption that the mathematical reasoning within this notation of infinite sets can be performed. Another issue is whether non-trivial subclasses of the models and perhaps subsets of the logic yields decidable systems.

It is also of great importance to find reasonable notations for subsets of infinite state systems, which, although not necessarily 'relative complete' at least yields convenient frameworks for application of the method. We show how this might be done for *bounded processes*, a subset of Milner's CCS [9] where processes do not have unlimited, evolving, but only bounded structure. Another example for Petri nets can be found in the work of Bradfield [4, 3], which employs the tableau method of Bradfield and Stirling [5]. The relation between their method and the method described here, will be considered in a later section. However, the main difference is that by annotating fixed-points with sets of states we are able to move validity from being global success criterions for the complete tableaux to more local criterions of inclusions of state-sets.

An interesting point manifest in the method, is the commonly accepted dogma that reasoning about maximum fixed-points is 'easy', like 'partial correctness' in Hoare-Floyd Logic allowing non-termination, and bisimulation equivalences in process algebras, whereas reasoning about minimum fixed-points is often more involved, as when showing termination of programs in Hoare-Floyd Logic. The analogy with Hoare-Floyd Logic can actually be made quite precise, see e.g. Bradfield [3, sec. 3.7]. In the method described here, these parallels manifest themselves, as reasoning about minimum fixed-points requires a well-founded relation to be supplied, whereas no such thing is required for the maximum fixed-point.

2 Fixed-Points

For a monotonic function ψ on a powerset $\mathcal{P}(S)$ we denote by $\mu\psi$ (refip. $\nu\psi$) the minimum (resp. maximum) fixed-point of ψ as given by Tarski's theorem [10]. Winskel [11] has shown that a slightly modified unfolding of a maximum fixed-point can be used as the key step in the development of a model checker for finite-state systems. This property of maximum fixed-points will

be referred to as the *reduction lemma:*

Lemma 1 (Reduction lemma, Kozen [7], Winskel [11]) *Let ψ be a monotonic function on the power-set $\mathcal{P}(S)$. For $V \subseteq S$ we have*

$$V \subseteq \nu\psi \Leftrightarrow V \subseteq \psi(\nu U.V \cup \psi(U)).$$

Winskel uses this lemma in the situation where V is a singleton $\{p\}$. He defines a relation which in a precise sense makes the right-hand side smaller, thus 'simpler' to verify, and because he works with finite-state systems, this relation turns out to be well-founded, ensuring termination of the algorithm. As we consider infinite state systems, termination is no longer guaranteed. Moreover, following Bradfield and Stirling [5] we will try to verify that (possibly infinite) sets of states satisfy an assertion, not only singletons. This seems more appropriate for infinite-state systems; although initially we might only want to know whether one particular state satisfies an assertion, this state can quickly lead to considering whether an infinite number of states satisfy an assertion (an example of this is provided later). So we will be involved in deciding judgements like $V \subseteq U$, where V is a (possibly infinite) set of states and U is a property expressed in our assertional language. We will use lemma 1 to give a rule for the maximum fixed-points, but what about the minimum fixed-points? The Duality Principle for Complete Lattices yields an immediate corollary.

Corollary 1 *Let ψ be a monotonic function on $\mathcal{P}(S)$. For $V \subseteq S$ we have*

$$V \supseteq \mu\psi \Leftrightarrow V \supseteq \psi(\mu U.V \cap \psi(U)).$$

This, however, is not very useful. Being interested in determining whether sets of states satisfy a property corresponds to determining whether $V \subseteq \mu\psi$ and *not* $V \supseteq \mu\psi$. So we must find another formulation. Notice, however, that for *singletons* we can derive a useful bi-implication like the one in the reduction lemma:

$$\begin{aligned} p \in \mu U.\psi(U) &\Leftrightarrow S\backslash\{p\} \not\supseteq \mu U.\psi(U) && \text{by simple set theory} \\ &\Leftrightarrow S\backslash\{p\} \not\supseteq \psi(\mu U.(S\backslash\{p\}) \cap \psi(U)) && \text{by corollary 1} \\ &\Leftrightarrow p \in \psi(\mu U.(\psi(U))\backslash\{p\})). \end{aligned}$$

(The first and last bi-implication fail for arbitrary sets). Hence, the minimum fixed-point on the right-hand side is now slightly 'smaller' as the state p has been excluded. For finite-state systems this is actually enough to ensure termination as the exclusion of states from a fixed-point cannot go on forever; eventually we will find out that a state p belongs to the minimum fixed-point, or we will be involved with deciding whether a state p belongs to a minimum fixed-point from which it has previously been explicitly excluded.

However, for infinite-state systems, excluding singletons are not enough to guarantee termination; we could go on unfolding the fixed-point forever without ever reaching a conclusion. Instead we will use a principle of *well-founded induction* based on the lemma below. Recall, that a relation $\sqsubset$ on the set U is a *well-founded relation* (abbreviated *w.f.r.*) if there does not exist an infinitely decreasing chain $u_0 \sqsupset u_1 \sqsupset \cdots \sqsupset u_n \sqsupset \cdots$. Moreover, we extend a relation $\sqsubset$ on U to a relation on $\mathcal{P}(U)$ by defining

$$V \sqsubset W \Leftrightarrow_{def} \ \forall v \in V, w \in W. v \sqsubset w$$

and we let $(\sqsubset W)$ be the set of elements of U less than all elements of W, i.e.

$$(\sqsubset W) =_{def} \{v \in V \mid \forall w \in W. v \sqsubset w\}.$$

To state the lemma we need the notion of a covering: A *covering* of U is a collection of sets $\{U_i\}_{i \in I}$ s.t. $\bigcup_{i \in I} U_i = U$.

Lemma 2 (Well-founded induction on minimum fixed-points)
Let ψ be a monotonic function on $\mathcal{P}(S)$. For a set $U \subseteq S$, the following holds:

If there exists a w.f.r. $\sqsubset$ on U and a covering $\{U_i\}_{i \in I}$
of U such that
$\quad \forall i \in I.\ U_i \subseteq \psi(\mu V.(\sqsubset U_i) \cup \psi(V))$
then $U \subseteq \mu\psi$

Proof: Recall, the principle of well-founded induction for a predicate Q on a set U with w.f.r. $\sqsubset$:

$$\text{If } \forall u \in U.\ (\forall u' \sqsubset u. Q(u')) \Rightarrow Q(u) \text{ then } \forall u \in U. Q(u).$$

Hence, take any $u \in U$. As $\{U_i\}_{i \in I}$ covers U, there exists a U_i containing u. We now deduce as follows:

$$\begin{aligned}
\forall u' \sqsubset u.\ u' \in \mu\psi \quad & \Rightarrow \forall u' \sqsubset U_i.\ u' \in \mu\psi \\
& \qquad \text{since } u \in U_i \\
& \Rightarrow \sqsubset U_i \subseteq \mu\psi \\
& \Rightarrow \mu V.(\sqsubset U_i) \cup \psi(V) = \mu\psi \\
& \qquad \text{from a simple observation about fixed-points} \\
& \Rightarrow u \in U_i \subseteq \psi(\mu V.(\sqsubset U_i) \cup \psi(V)) = \psi(\mu\psi) = \mu\psi \\
& \qquad \text{where the inclusion follows by assumption.}
\end{aligned}$$

From the principle of well-founded induction we conclude that

$$\forall u \in U.\ u \in \mu\psi,$$

proving the lemma. □

The other direction of the implication holds in a trivial way. Take $I = \{1\}, U_1 = U$, and $\sqsubset$ any w.f.r., for instance, the empty relation. Then as $(\sqsubset U) = \emptyset$, the requirement to this trivial covering degenerates to the validity of unfolding of fixed-points. However, also more interesting choices of covering and well-founded relation exist, indeed in showing completeness of the method we will argue that a certain canonical covering and relation can be found such that the minimum fixed-point will never be unfolded more than once.

3 Logic

We will use a version of Kozen's modal μ-calculus [7], extended with constants, sets of actions in the modalities, and annotations on the fixed-points expressing states 'assumed to satisfy' the fixed-point. The syntax is described by the following grammar:

$$A ::= Q \mid A_0 \vee A_1 \mid A_0 \wedge A_1 \mid \langle\kappa\rangle A \mid [\kappa]A \mid X \mid \mu X\{U\}A \mid \nu X\{U\}A$$

In the modalities κ is a (possibly infinite) set of *labels*. We use the abbreviation '$\cdot$' for all labels. As models we take labelled transition systems $T = (S, L, \rightarrow)$ where S is a set of states, L a set of labels, and $\rightarrow \subseteq S \times L \times S$

a transition relation. Due to the presence of constants and variables in the logic the semantics will be given relative to a valuation V taking constants to sets of states, and an environment ρ taking variables to sets of states. Hence $[\![Q]\!]_{T,V}\rho = V(Q)$ and $[\![X]\!]_{T,V}\rho = \rho(X)$. Conjunction and disjunction are interpreted as intersection and union. The denotation of the modalities are

$$[\![\langle\kappa\rangle A]\!]_{T,V}\rho = \{s \in S \mid \exists s' \in S\ \exists a \in \kappa.\ s \xrightarrow{a} s'\ \&\ s' \in [\![A]\!]\rho\}$$
$$[\![[\kappa] A]\!]_{T,V}\rho = \{s \in S \mid \forall s' \in S\ \forall a \in \kappa.\ s \xrightarrow{a} s' \Rightarrow s' \in [\![A]\!]\rho\}$$

and for the fixed-points, let $\psi : \mathcal{P}(S) \to \mathcal{P}(S)$ be the function $\psi(U) = V \cup [\![A]\!]_\rho[U/X]$ and define

$$[\![\mu X\{V\}A]\!]_{T,V}\rho = \mu\psi$$
$$[\![\nu X\{V\}A]\!]_{T,V}\rho = \nu\psi$$

This means that the usual $\mu X.A$ is an abbreviation for $\mu X\{\emptyset\}A$. For closed assertions define $[\![A]\!]_{T,V} = [\![A]\!]_{T,V}\rho$ for any environment ρ. When there is no risk of confusion we will even leave out T and V. We define the *satisfaction predicate* $\models$ on *correctness assertions* $U : A$ as follows: For a closed assertion A and a set $U \subseteq S$ let

$$\models_{T,V} U : A \ \Leftrightarrow_{def}\ U \subseteq [\![A]\!]_{T,V}.$$

4 The Model Checking Method

In this section we will introduce a syntactic counterpart $\vdash$ of the satisfaction relation $\models$ and give a set of rules that allow us to verify that correctness assertions $U : A$ belongs to $\vdash$. Let $CorrAssn^{cl}$ be the set of closed correctness assertions. We give a binary relation $\longrightarrow \subseteq CorrAssn^{cl} \to \mathcal{P}(CorrAssn^{cl})$ between correctness assertions and sets of correctness assertions. The intuition is that if $(U : A) \longrightarrow \Gamma$ then to prove that $(U : A)$ is valid, we can instead prove each of the correctness assertions in the set Γ. However, as the minimum fixed-points can result in infinite sets of correctness assertions - all of the same 'form' - we will describe a 'schematic relation' $\Longrightarrow \subseteq \mathcal{P}(CorrAssn^{cl}) \to \mathcal{P}(CorrAssn^{cl})$ which will allow *sets* of correctness assertions to be rewritten. The defining clauses for $\longrightarrow$ are given in figure 1.

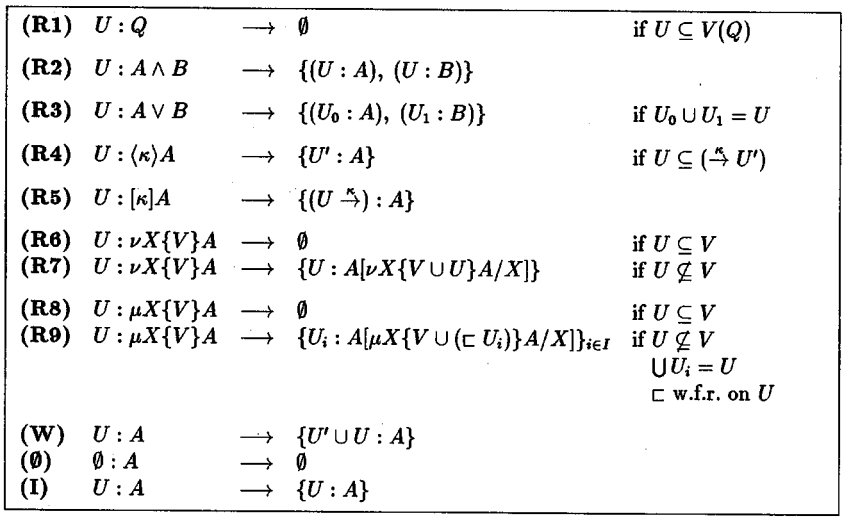

Figure 1: The rules

We notice that the rules **(R1)**, **(R2)**, **(R5)**, **(R6)**, **(R7)**, **(R8)**, **(∅)** and **(I)** all are deterministic, in the sense that, given an instantiation of the left-hand side there is only one instantiation of the right-hand side, whereas **(R8)**, **(R4)** ,**(R9)**, and **(W)** all involve choices, and as there in several will be more than one proper choice, give raise to several possible instantiations of the right-hand sides, thus introducing 'non-determinacy'. For the method to be successful in showing validity of a correctness assertion these choices must all be made successfully. Let us consider the rules in more detail:

(R1), (R2), (R3). All are quite obvious. Only **(R3)** involves a choice.

(R4), (R5). In rule **(R5)**$(U \xrightarrow{\kappa})$ denotes the set of states that can be reached through an action in κ from a state in U, i.e.

$$(U \xrightarrow{\kappa}) = \{s \in S \mid \exists u \in U\ \exists a \in \kappa.u \xrightarrow{a} s\}.$$

The importance of this operator is that

$$U \subseteq [\![[\kappa]A]\!] \Leftrightarrow (U \xrightarrow{\kappa}) \subseteq [\![A]\!].$$

It is, however, not possible to define a similar operation for the diamond-modality, which inevitably must involve some choices. To see this, consider the simple three-state transition system $(\{p,q,r\},\{a\},\rightarrow)$ with $p \stackrel{a}{\rightarrow} q$ and $p \stackrel{a}{\rightarrow} r$. Now, if $\{p\} : \langle a \rangle A$ is to be valid, then *either* $\{q\} : A$ *or* $\{r\} : A$ *or both* must be valid, but it is not possible to tell whether we should insist on this being $\{q\}, \{r\}$ or perhaps $\{q,r\}$. We have chosen to present this choice in a way which also allows for weakening, hence in **(R4)** U' is any set which satisfies $U \subseteq (\stackrel{\kappa}{\rightarrow})U'$, where

$$\stackrel{\kappa}{\rightarrow} U' = \{s \in S \mid \exists u \in U' \; \exists a \in \kappa.\; s \stackrel{a}{\rightarrow} u\}.$$

Notice, that **(R5)** could have been given in a completely analogous fashion, but we keep the current presentation because it is deterministic and the analogue of **(R4)** for the box-modality can be achieved as a derived rule through the weakening rule **(W)**.

(R6), (R7), (R8), (R9). The ν-rule **(R7)** expresses the reduction lemma and **(R6)** an easy consequence of the semantics of the ν-operator. The μ-rule **(R6)** is inspired by lemma 2.

(W). The weakening rule allows for very many choices! It is essential to the completeness of the system.

(∅) Included for convenience. It is derivable from the other rules.

(I). The identity rule making $\longrightarrow$ 'reflexive' is added in order to allow $\Longrightarrow$ to leave some correctness assertions unchanged.

$$\frac{\forall \gamma \in \Gamma.\; \gamma \longrightarrow \Delta_\gamma}{\Gamma \Longrightarrow \bigcup_{\gamma \in \Gamma} \Delta_\gamma} \quad (\Longrightarrow)$$

Figure 2: The simultaneous rewrite relation.

The μ-rule **(R9)** might give raise to infinite sets of correctness assertions being generated. However, they all have the same form and we can expect that they to a large extent can be rewritten simultaneously, considering the index i merely as a parameterization of the correctness assertions. To formalize this idea we introduce a rewriting relation between (possibly infinite)

sets of correctness assertions $\Longrightarrow$. It has one defining rule given in figure 2. As $\longrightarrow$ by (**I**) is reflexive the rule allows one to select some of the correctness assertions in Γ to be rewritten according to $\longrightarrow$ and leave others unchanged.

Let $\Longrightarrow^* = (\bigcup_{n\in\omega} \Longrightarrow^n)$ where $\Longrightarrow^0 = Id, \Longrightarrow^{n+1} = (\Longrightarrow \circ \Longrightarrow^n)$. A correctness assertion $U : A$ is now provable on a transition system T with valuation V, written $\vdash_{T,V} U : A$ if this fact can be derived using $\Longrightarrow^*$, i.e.

$$\vdash_{T,V} U : A \Leftrightarrow_{def} \{U : A\} \Longrightarrow^* \emptyset.$$

With this definition of a provably correct assertion, the rules are sound and complete: (Proofs can be found in appendix A.)

Theorem 1 (Soundness) *Suppose T is a labelled transition system with valuation V and A is a closed assertion. If $\vdash_{T,V} U : A$ then $\models_{T,V} U : A$.*

Theorem 2 (Completeness) *Suppose T is a labelled transition system with valuation V and A is a closed assertion. If $\models_{T,V} U : A$ then $\vdash_{T,V} U : A$.*

5 Examples

In this section we will show how to apply the method to two small examples. We will use (a subset of) Milner's CCS with value-passing [9] for expressing transition systems, and suggest a notation for infinite sets of states which seems to be particularly useful for a class of bounded processes; processes which do not have arbitrarily, unbounded evolving structure.

First, we recall the syntax. Assume that $\mathcal{A}$ is a set of neutral actions or channel names, and assume that $\mathbb{V}$ is a set of values. Process expressions are generated from the syntax

$$E ::= \mathbf{0} \mid \pi.E \mid GE \mid E + E \mid E \mid E \mid C(e_1, \ldots, e_n),$$

where C denotes a process constant with arity n defined through an equation $C(v_1, \ldots, v_n) = E$, where the free value variables of E are among $v_1, \ldots, v_n$ (we will abbreviate this as $\vec{v}$). Constant definitions can be mutually recursive. The actions π are either input (?), output (!), or silent (τ) actions,

$$\pi ::= a?v \mid a!e \mid \tau$$

where $a \in \mathcal{A}$. Value expressions e are build from a set of operators, value variables $v \in var$, and constants $c \in const$. Guards, $G = (\psi)$, are boolean expressions over predicates on the value expressions. The operational semantics of CCS with value-passing is standard giving a 'universal' labelled transition $\mathcal{T}$ (see Milner [9]). In $\mathcal{T}$ states are identified with closed process expressions, so sets of states are sets of processes, which we suggest can be described by

$$\vec{t}; \vec{\psi}\ [\vec{v}],$$

where $\vec{t}$ is a list of process expressions, $\vec{\psi}$ a list of predicates, and $\vec{v}$ a list of free value-variables, which are implicitly assumed to be universally quantified. That is, tentatively the semantics of $\vec{t}; \vec{\psi}\ [\vec{v}]$ is the set

$$[\![\vec{t}; \vec{\psi}\ (\vec{v})]\!] = \{\vec{t}[\vec{c}/\vec{v}] \mid \vec{\psi}[\vec{c}/\vec{v}], c_i \in \mathbb{V}\}$$

where $\vec{v}$ includes all the free variables of $\vec{t}$ and $\vec{\psi}$. An example when the values are natural numbers is

$$P, Q(n); n > 0, n \leq 3\ [n]$$

i.e. the set $\{P, Q(1), Q(2), Q(3)\}$. We will simply write $\vec{t}\,[\vec{v}]$ instead of $\vec{t}; \vec{\psi}\ [\vec{v}]$ when $\vec{\psi}$ is empty. Now, entailments will be on the form $\vdash (\vec{t}; \vec{\psi}\ [\vec{v}]) : A$ or $\vdash (\vec{t}\,[\vec{v}]) : A$.

Example 1 This is a classic example. It has been used to show that on a very simple transition system, $\mu X\{\}[.]X$ cannot be found as the ω-limit of its approximants: $F, [.]F, [.][.]F, \ldots$; the ordinal $\omega + 1$ is necessary. Define P and $Q(n)$ as follows:

$$\begin{aligned} P &= a?n.Q(n) \\ Q(n) &= (n > 0)r.Q(n-1) \end{aligned}$$

Hence P inputs a number n on the channel a, and then proceeds by making n τ's. We will show that P always terminates, i.e. that all execution sequences are finite. This is expressed in the modal μ-calculus as $\mu X\{\}[.]X$. We rewrite as follows:

$$
\begin{array}{lll}
P : \mu X\{\}[.]X & \xrightarrow{(\mathbf{R9})} & P :[.]\mu X\{\}[.]X \\
 & & \quad \text{with trivial singleton covering, arbitrary w.f.r.} \\
 & \xrightarrow{(\mathbf{R5})} & Q(n)[n] : \mu X\{\}[.]X \\
 & \xrightarrow{(\mathbf{R9})} & \{Q(n) : [.]\mu X\{\sqsubset Q(n)\}[.]X\}_{n\in\omega} \\
 & & \quad \text{with covering } Q(n)_{n\in\omega} \text{ and} \\
 & & \quad \text{w.f.r. } Q(m) \sqsubset Q(n) \Leftrightarrow_{def} m < n \\
 & = & (Q(0) : [.]\mu X\{\sqsubset Q(0)\}[.]X), \\
 & & \quad \{Q(n) : [.]\mu X\{\sqsubset Q(n)\}[.]X\}_{n>0} \\
 & \xRightarrow{(\mathbf{R5})} & \emptyset : \mu X\{\sqsubset Q(0)\}[.]X), \{Q(n) : [.]\ldots\}_{n>0} \\
 & & \quad \text{since } Q(0) \not\rightarrow, \\
 & & \quad \text{hence } (Q(O) : [.]\mu\ldots) \longrightarrow (\emptyset : \mu\ldots) \\
 & \xRightarrow{(\emptyset)} & \{Q(n) : [.]\mu X\{\sqsubset Q(n)\}[.]X\}_{n>0} \\
 & \xRightarrow{(\mathbf{R5})} & \{Q(n-1) : \mu X\{\sqsubset Q(n)\}[.]X\}_{n>0} \\
 & \xRightarrow{(\mathbf{R8})} & \emptyset \text{ as } n-1 < n.
\end{array}
$$

Notice, that the splitting of the ω-set of correctness assertions after the third step was strongly suggested to us by the guard $n > 0$ in the definition of $Q(n)$.

It is also worthwhile to observe that although we used a covering with singleton sets here, it is not always necessary to fall back on singletons. If we instead had the definition

$$
\begin{array}{rcl}
P & = & a?n.b?m.Q(n,m) \\
Q(n,m) & = & (n > 0)c!m.b?m.Q(n-1,m),
\end{array}
$$

we could use the covering

$$\{\{Q(n,m) \mid m \in \omega\}\}_{n\in\omega}$$

and the w.f.r. $Q(n',m') \sqsubset Q(n,m) \;\Leftrightarrow_{def}\; n' < n$. $\Box$

Example 2 This is an example from Bradfield [4, p.6]. Consider the following definition of a process M:

$$M(A,B,C) = (A \geq 1)a.M(A,B+1,C)$$

$$+ (A \geq 1)b.M(A-1, B, C+1)$$
$$+ (B \geq 1 \wedge C \geq 1)c.M(A, B-1, C)$$

The process $M(l, m, n)$ is really describing the firing sequence of a certain *Petri net* with l *tokens* on the *place* A, m tokens on place B, and n tokens on the place C, and the actions a, b, and c are *transitions* of the Petri net.

Using the previously defined notation, sets of states will now be described by

$$M(A, B, C); \vec{\psi}$$

For convenience, we will omit $M(A, B, C)$ and just write $\vec{\psi}$. The initial marking we consider is $A = 1, B = 0, C = O$ and we will show that c only happens finitely often, expressed as the assertion $\mu X\{\}\nu Y\{\}[c]X \wedge [a, b]Y$. Intuitively this is obvious: Either a fires indefinitely, increasing the number of tokens on B, or at some point b fires, and then only c can fire. As there is only a finite number of tokens on B when this happens and c removes one token whenever fired, it must eventually stop.

Formally, we show:

$$(A = 1, B = 0, C = 0 : \mu X\{\}\nu Y\{\}[c]X \wedge [a, b]Y) \Longrightarrow^* \emptyset$$

Let $X_0 = \mu X\{\}\nu Y\{\}[c]X \wedge [a, b]Y$ and rewrite as follows:

$(A = 1, B = 0, C = 0 : X_0)$

$\xrightarrow{(\mathbf{W})}$ $A + C = 1 : X_0$

$\xrightarrow{(\mathbf{R9})}$ $\{A + C = 1, B = n : \nu Y\{\}[c]X_1 \wedge [a, b]Y\}_{n \in \omega}$
where $X_1 = \mu X\{A + C = 1, B < n\}\nu Y\{\}[c]X \wedge [a, b]Y$

$\xrightarrow{(\mathbf{R7})}$ $\{A + C = 1, B = n : [c]X_1 \wedge [a, b]Y_0\}_{n \in \omega}$
where $Y_0 = \nu Y\{A + C = 1\}[c]X_1 \wedge [a, b]Y$

$\xrightarrow{(\mathbf{R2})}$ $\{A + C = 1, B = n : [c]X_1, A + C = 1, B = n : [a, b]Y_0\}_{n \in \omega}$

$\xrightarrow{(\mathbf{R5})}$ $\{A + C = 1, B = n : [c]X_1\}_{n \in \omega}, \{A + C = 1, B = n,$
$n = 0 \Rightarrow C = 1 : Y_0\}_{n \in \omega}$

$\xrightarrow{(\mathbf{R6})}$ $\{A + C = 1, B = n : [c]X_1\}_{n \in \omega}$

$\xrightarrow{(\mathbf{R5})}$ $\{A = 0, B = n - 1, C = 1 : X_1\}_{n \in \omega}$

$\xrightarrow{(\mathbf{R8})}$ $\emptyset$

It is essential to extend the sets of markings in the first weakening step in order to make the later application of rule (**R9**) successful. □

In the previous two examples, the processes involved were of a particular simple kind, they did not have 'evolving structure'. To be precise about this, let $\widehat{\ }$ be the operation which, by simply ignoring values, maps CCS process expressions with values to CCS process expressions without. I.e. on the action prefixes it behaves as: $\widehat{a?v} = a, \widehat{a!v} = \bar{a}, \hat{r} = r$.

Definition 1 A CCS process P with values is *bounded* if the set

$$\{\widehat{Q} \mid \exists n\ \exists a_1, \ldots, a_n.\ P \xrightarrow{a_1} \xrightarrow{a_2} \ldots \xrightarrow{a_n} Q\}$$

is finite. □

The notation we have used seems to be particularly well-suited for bounded processes, as all the reachable states can be described by a finite number of process expressions, together with a collection of constraints on the free value-variables. We claim that it is not difficult to see that each particular *state* can actually be described by a process expression and a finite number of constraints, but whether any *set of states* expressible in the modal μ-calculus can actually be described by finitely many constraints, yielding a relative completeness result, is another issue not addressed here.

$$\frac{U : \langle\kappa\rangle A}{U' : A} \quad (U \subseteq (\xrightarrow{\kappa} U'))$$

$$\frac{U : [\kappa] A}{(U \xrightarrow{\kappa}) : A}$$

$$\frac{U : \mu X\{V\}A}{} \quad (U \subseteq V)$$

$$\frac{U : \mu X\{V\}A}{\{U_i : A[\mu X\{V \cup (\sqsubset U_i)\}A/X]\}_{i\in I}} \quad \left(\begin{array}{l} U \not\subseteq V \\ \bigcup U_i = U \\ \sqsubset \text{ w.f.r. on } U \end{array}\right)$$

$$\frac{U : \nu X\{V\}A}{} \quad (U \subseteq V)$$

$$\frac{U : \nu X\{V\}A}{U : A[\nu X\{V \cup U\}A/X]} \quad (U \not\subseteq V)$$

Figure 3: Some of the rewrite rules presented as 'goal-oriented' proof rules.

6 Relation to the Tableau Method of Bradfield and Stirling

We have chosen to present the method as a set of rewrite rules. However, it is not difficult to give a presentation of the rewrite rules as 'goal-oriented' proof rules (see figure 3). In general, a rewrite rule of the form

$$(U : A) \longrightarrow \Gamma \text{ if } C$$

gives raise to a proof rule

$$\frac{U : A}{\Gamma}(C)$$

which has side condition C.

Besides the annotations on fixed-points which localizes validity, i.e. makes it independent of the proof tree, the main difference to the tableau method of Bradfield and Stirling [5, 3] is in the treatment of the minimum fixed-points. Whereas Bradfield and Stirling constructs a finite proof tree with certain non-trivial success criterions - a tableau - which for the minimum fixed-point involves determining, outside the system, well-foundedness of a relation *induced by the tableau*, we supply a well-founded relation on the states which is *independent* of the proof being constructed; and carry out the inductive reasoning *inside the system* as we proceed with the rules.

For the present method, building a proof tree, showing how rules are applied, is not an essential ingredient, but it could be used as an organizational trick that makes explicit where choices were taken and perhaps could be altered.

Another apparent difference is that the tableau method of Bradfidd and Stirling constructs a finite proof tree, whereas the application of $\Longrightarrow$ seems to have an inherent infinite nature. However, the appealing feature of generating finite proof trees has the cost of pushing the infinite reasoning into the reasoning involved in showing well-foundedness of the relation induced by the tableau. Moreover, the infinite nature of $\Longrightarrow$ is only apparent. As the examples show the infinite reasoning performed with $\Longrightarrow$ is rather innocent; the correctness assertions all have the same form, so the proof proceeds in the same manner for each correctness assertion, and is thus more a means of proving 'parameterized' correctness assertions.

7 Conclusion

When restricting ourselves to finite-state systems and using only singletons in the correctness assertions, we can replace the few choices that remains by finite disjunctions, thereby rediscovering the model checker of Winskel - in a version without negations, but with an explicit rule for the minimum fixed-point. Note, however that for the finite case, more efficient algorithms exist (see for instance Cleaveland and Stelfen [6], Larsen [8], and Andersen [2].) One short-coming of the method presented so far, is the inability to show that $\models U : A$ does *not hold*. The rules are not very appropriate for this; one has to show that all the possible choices lead to false expressions.

An obvious attempt to remedy this would be to simply try to show that U satisfies another assertion making $\models U : A$ impossible. If U is a singleton $\{u\}$, this is quite easy as $\not\models \{u\} : A \Leftrightarrow \models \{u\} : \neg A$ where we have introduced negation with semantics $[\![\neg A]\!] = S \setminus [\![A]\!]$, i.e. the complement of A. This is not the case for general U, but we instead observe that

$$\not\models U : A \Leftrightarrow \exists U'(\emptyset \neq U' \subseteq U).\ \models U' : \neg A.$$

Instead of introducing a new rule for negation - which is just as difficult as to cope with as showing non-validity - we consider $\neg A$ to be simply an operation on assertions that dualizes every operator in A (taking constants to new constants denoting their complement, $\langle\kappa\rangle$ to $[\kappa]$, μ to ν etc.), thereby making the method applicable as it is.

References

[1] P. Aczel. An introduction to inductive definitions. In Jon Barwise, editor, *Handbook of Mathematical Logic.* North-Holland, 1983.

[2] Henrik Reif Andersen. Model checking and boolean graphs (extended abstract). In B. Krieg-Brückner, editor, *Proceedings of 4'th European Symposium on Programming, ESOP'92, Rennes, France*, volume 582 of *LNCS*. Springer-Verlag, 1992.

[3] Julian C. Bradfield. *Verifying Temporal Properties of Systems with Applications to Petri Nets.* PhD thesis, Laboratory for Foundations of Computer Science, University of Edinburgh, July 1991.

[4] Julian C. Bradfield. A proof assistant for symbolic model-checking. Technical Report ECS-LFCS-92-199, Laboratory for Foundations of Computer Science, University of Edinburgh, March 1992.

[5] Julian C. Bradfidd and Colin P. Stirling. Verifying temporal properties of processes. In J.C.M. Baeten and J.W. Klop, editors, *Proceedings of CONCUR '90*, volume 458 of *LNCS*, pages 115-125. Springer-Verlag, 1990.

[6] Rance Cleaveland and Bernhard Steffen. A linear-time model-checking algorithm for the alternation-free modal mu-calculus. In Kim G. Larsen and Arne Skou, editors, *Proceedings of the 3rd Workshop on Computer Aided Verification, July 1991, Aalborg*, volume 575 of *LNCS*. Springer-Verlag, 1992.

[7] Dexter Kozen. Results on the propositional mu-calculus. *Theoretical Computer Science,* 27, 1983.

[8] Kim G. Larsen. Efficient local correctness checking. In *Proceedings of the 4th Workshop on Computer Aided Verification, June 29 - July 1, 1992, Montreal, Quebec, Canada*, 1992. Forthcoming.

[9] Robin Milner. *Communication and Concurrency*. Prentice Hall, 1989.

[10] A. Tarski. A lattice-theoretial fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5:285-309, 1955.

[11] Glynn Winskel. A note on model checking the modal ν-calculus. In G. Ausiello, M. Dezani-Ciancaglini, and S. Ronchi Della Rocca, editors, *Proceedings of ICALP*, volume 372 of *LNCS*, pages 761-772, 1989.

A Proofs of Soundness and Completeness

In this section we show soundness (theorem 1) and completeness (theorem 2) of the method.

A.1 Soundness

In order to show soundness we assume that $\vdash U : A$, i.e. $\{U : A\} \Longrightarrow^* \emptyset$ and argue that $\models U : A$.

Proof (Soundness): Let the predicate Q be defined by

$$Q(n) \Leftrightarrow_{def} (\Gamma \Longrightarrow^n \emptyset) \Rightarrow \text{for all } (U : A) \in \Gamma.\ \models U : A.$$

We prove by induction on $n \in \omega$ that $Q(n)$ holds for all n, from which the theorem follows. The base case is trivial. As the only clause defining $\Longrightarrow$ is

$$\frac{\forall \gamma \in \Gamma.\ \gamma \longrightarrow \Delta_\gamma}{\Gamma \Longrightarrow \cup_{\gamma \in \Gamma} \Delta_\gamma} \quad (\Longrightarrow)$$

the inductive step amounts to showing that if $(U : A) \longrightarrow \Delta$ and $\Delta \longrightarrow^{n-1} \emptyset$ then $\models U : A$. By the induction hypothesis $\Delta \Longrightarrow^{n-1} \emptyset$ implies that for all $(U' : A') \in \Delta$ we have $\models U' : A'$, hence we must argue that if

$$(U : A) \longrightarrow \Delta \ \ \& \ \ \forall (U' : A') \in \Delta.\ \models U' : A'$$

then

$$\models U : A.$$

We consider each rule in turn.

(R1), (R2), (R3), (R4), (R5). Straightforward.

(R6), (R7). From the semantics of the annotated maximum fixed-point we deduce as follows

$$[\![\nu X\{V\}A]\!]\rho = \nu W.V \cup [\![A]\!]\rho[W/X] = V \cup [\![A]\!]\rho[\nu W. \ldots /X] \supseteq V.$$

Hence, certainly if $U \subseteq V$, we have $U \subseteq [\![\nu X\{V\}A]\!]\rho$ and therefore $\models U : \nu X\{V\}A$ proving soundness of **(R6)**. Otherwise, if $U \not\subseteq V$, the soundness of **(R7)** follows from lemma 1.

(R8), (R9). Rule **(R8)** is like for the minimum fixed-point above. The rule **(R9)** is sound by lemma 2.

(W),(∅),(I). Trivial.

□

A.2 Completeness

In order to show completeness we will need some facts about the ordinals, On. Let $<$ be the well-founded relation on On. Define for a monotone function $f : \mathcal{P}(S) \to \mathcal{P}(S)$ the set $\mu^\alpha f$ inductively as follows:

$$\begin{array}{rcl} \mu^0 f & = & \emptyset \\ \mu^{\alpha+1} f & = & f(\mu^\alpha f) \\ \mu^\lambda f & = & \bigcup_{\alpha<\lambda} \mu^\alpha f \quad \text{for } \lambda \text{ a limit ordinal.} \end{array}$$

The following proposition shows, that the minimum fixed-point of a monotonic function f on a powerset can be found as the least upper bound of all the approximants $\mu^\alpha f$.

Proposition 1 *Let $\mathcal{P}(S)$ be a powerset, and assume $f : \mathcal{P}(S) \to \mathcal{P}(S)$ is a monotonic function. Then $\{\mu^\alpha f\}_{\alpha \in On}$ is an increasing sequence with*

$$\mu f = \bigcup_{\alpha \in On} \mu^\alpha f,$$

and there is a least ordinal β (the closure ordinal)*, such that $\mu^\beta f = \mu^{\beta+1} f$ and*

$$\mu f = \mu^\beta f.$$

We denote this ordinal cl(f).

Proof: The proposition holds in all complete lattices, consult e.g. Aczel [1] for a proof. □

In the completeness proof we construct a canonical proof which only need to unfold each fixed-point once. A simple property of the annotations on fixed-points that make this possible is captured by the following lemma.

Lemma 3 *Assume that V is a valuation with $V(Q_X) = W$. If A is an assertion with only one free variable X and*

$$\vdash_V U : A[Q_X/X]$$

then

$$\vdash_V U : A[\mu X\{W\}B/X]$$

and

$$\vdash_V U : A[\nu X\{W\}B/X]$$

Proof: Simple structural induction on A. Any application of the **(R1)**-rule is replaced by an application of **(R6)** or **(R8)**. □

Define the relation $\prec$ on closed assertions by $A' \prec A$ if and only if, $A' = A''[\vec{Q}/\vec{X}]$ for some proper subassertion A'' of A where $\vec{Q}$ is a vector of constants and $\vec{X}$ are the free variables of A''. Surely, $\prec$ is well-founded. We can now prove the completeness:

Proof (Completeness): We show that the following predicate $P(A)$ holds for all closed assertions A by induction on $\prec$:

$$P(A) \Leftrightarrow_{def} \vdash_V [\![A]\!] : A$$

Given an A, let $U = [\![A]\!]$ and consider the possible forms of A.

$A \equiv \nu X\{V\}B$. If $U \subseteq V$ the claim follows from rule **(R6)**. Otherwise, assume given a constant Q with valuation $V(Q) = U$. Then by the induction hypothesis

$$\vdash_V [\![B[Q/X]]\!] : B[Q/X]$$

which by lemma 3 implies

$$\vdash_V [\![B[Q/X]]\!] : B[\nu X\{V \cup U\}B/X].$$

Hence, as $[\![B[Q/X]]\!] = [\![B[\nu X\{V \cup U\}B/X]\!] = U$ then

$$\vdash_V U : B[\nu X\{V \cup U\}B/X].$$

and the result follows from rule **(R7)**.

$A \equiv \mu X\{V\}B$. If $U \subseteq V$ the claim follows from rule **(R8)**. Otherwise, we use rule **(R9)** in the following way. Let $\psi(Z) = [\![B]\!]_\rho[Z/X] \cup V$ for an arbitrary environment ρ. Let β be the closure ordinal of ψ, and let for all $u \in U, \alpha_u$ be the ordinal such that $\alpha_u + 1$ is the least ordinal with

$$u \in \mu^{\alpha_u+1}\psi$$

(Notice, that this *must* be a successor ordinal, as for a limit ordinal λ,

$$u \in \bigcup_{\gamma<\lambda} \mu^\gamma\psi$$

implies that there exists $\gamma < \lambda$ such that $u \in \mu^\gamma\psi$. Moreover, $\alpha_u + 1$ can be no bigger than the closure ordinal β of ψ.)

Define the relation $\sqsubset$ on elements of $U = \mu\psi$ by $u' \sqsubset u$ iff $\alpha_{u'} < \alpha_u$. By the wellfoundedness of the ordinals, $\sqsubset$ is a well-founded relation. Notice, that $\mu^{\alpha_u}\psi = (\sqsubset u)$.

Assume given a set of constants Q_u with valuation $V(Q_u) = V \cup (\sqsubset u)$. Then since $B[Q_u/X] \prec \mu X\{V\}B$ the induction hypothesis yields

$$\vdash_V [\![B[Q_u/X]]\!] : B[Q_u/X]$$

which by lemma 3 implies

$$\vdash_V [\![B[Q_u/X]]\!] : B[\mu X\{V \cup (\sqsubset u)\}B/X]. \quad (1)$$

Observe, that $u \in \mu^{\alpha_u+1}\psi = \psi(\mu^{\alpha_u}\psi) = \psi(\sqsubset u) \subseteq [\![B[Q_u/X]]\!]$ assuming $V(Q_u) = V \cup (\sqsubset u)$.

We can now proceed rewriting with $\Longrightarrow$ as follows:

$$
\begin{aligned}
[\![\mu X\{V\}B]\!] : \mu X\{V\}B &\longrightarrow \{u : B[\mu X\{V \cup (\sqsubset u)\}B/X]\}_{u\in[\mu X\{V\}B]} \\
&\quad \text{by } \textbf{(R9)} \\
&\Longrightarrow \{[\![B[Q_u/X]]\!] : B[\mu X\{V \cup (\sqsubset u)\}B/X]\}_{u\in[\![\mu x\{V\}B]\!]} \\
&\quad \text{by } \textbf{(W)} \text{ since } u \in [\![B[Q_u/X]]\!] \\
&\Longrightarrow \emptyset \\
&\quad \text{by (1).}
\end{aligned}
$$

Remaining cases. They are all very straightforward.

We can now prove the completeness: For any $U \subseteq [\![A]\!]$ we use the weakening rule to rewrite as follows

$$(U : A) \longrightarrow ([\![A]\!] : A)$$

and then by the inductive proof above,

$$([\![A]\!] : A) \Longrightarrow^* \emptyset.$$

□