Comparing CTL and PCTL on Labeled Markov Chains

M.R.A. Huth
Department of Computing and Information Sciences
Kansas State University, Manhattan, KS66506, USA; phone: +1-785-532-6350; fax: +1-785-532-7353; email: huth@cis.ksu.edu

M.Z. Kwiatkowska
School of Computer Science, University of Birmingham
Edgbaston, B15 2TT, United Kingdom; phone: +44-121-414-7264; fax: +44-121-414-4281; email: mzk@cs.bham.ac.uk

Abstract

We interpret modal mu-calculus formulas in a non-standard way as expectations over labeled Markov chains. In this semantics fixed-points are computed in an iterative incremental/decremental fashion. This provides the foundation for a symbolic model checking algorithm for CTL over labeled Markov chains using Multi-Terminal Binary Decision Diagrams, where expectations of `until`-formulas are calculated via suitable fixed-point approximations. We compare this non-standard semantics of CTL to the standard one for the probabilistic logic PCTL by specifying a translation $\phi \mapsto \phi^\flat$ of the positive existential fragment CTL$^+$ of CTL into PCTL, where PCTL probability thresholds are supplied by the non-standard semantics. Finally, we show soundness of our non-standard semantics of CTL$^+$ with respect to the one for PCTL in the following sense: if our semantics computes a positive evidence for ϕ at state s then s satisfies the translated PCTL formula $\phi^\flat$.

Keywords

Labeled Markov Chains, Probabilistic Verification, Fuzzy Logic, CTL, PCTL.

1 INTRODUCTION

Probabilistic logics have proved successful in the specification and verification of systems that exhibit uncertainty. The behavior of such systems is typically governed by a known probability distribution; for example, randomized algorithms toss fair coins before making choices, and communication protocols must be designed so that the loss of messages in the medium is tolerated up to some appropriately small probability. With the advent of randomization, which ensures that many problems in distributed systems have efficient solutions albeit at a cost of probabilistically complex scenarios developing during execution, it has become necessary to develop formalisms in which the probability of some event happening in the course of execution can be assessed and, whenever possible, calculated automatically and efficiently by means of a model checking tool. The importance of probabilistic verification lies in the fact that it can provide guarantees that the specifications hold with satisfactory probability in cases when conventional model checking fails, for example, when exhaustive search is not feasible due to the size of the system, or when checking 'soft deadlines' in real-time systems, i.e. properties such as 'the system will respond within time t with probability 0.99'.

Established work on probabilistic temporal logics e.g. [25, 13, 12, 3, 23, 10] (and also modal logics, for example a probabilistic extension of Hennessy-Milner logic called PML [19]) uses probability thresholds for truth in the syntax of an underlying language of logic such as CTL [7]. This was achieved by H. Hansson and B. Jonsson by extending the models to variants of Markov chains, which involves adding probabilities to the transitions, and the addition of probabilistic operators $[\cdot]_{>p}$ to form the logic PCTL [13, 12]; then for a path formula ϕ the probabilistic formula $[\phi]_{>p}$ is satisfied in the given state if the probability of the set of paths starting in this state on which ϕ holds is *at least* p for some real number p in the unit interval. Model checking algorithms based on computing the probability and then comparing it against the threshold have been formulated for such systems (and also their extensions to allow concurrency or non-determinism), see e.g. [25, 10, 13, 3, 2]. Their complexity, however, is high, and so efficient techniques, such as symbolic model checking proposed in [1] or approximative reasoning, are sought for.

A more recent alternative to the above is to re-interpret the conventional logics in terms of *expectations*, see [20, 14]; see [16, 18] for a much earlier source of similar ideas. This involves assigning to formulas a map from states to the unit interval instead of the usual truth values, with the value at a given state representing (an estimate of) *the probability* of the formula being true in this state as calculated in the underlying Markov chain model.

This paper draws its main motivation from trying to understand the two seemingly opposing semantic approaches mentioned above when interpreted over labeled Markov chains: the threshold view as espoused by PCTL [13, 12] versus the logic CTL with the non-standard expectations semantics extracted from the quantitative mu-calculus semantics of [14]. If we think of such logics as providing a foundation for probabilistic methods in verification then a number of observations can be made about the threshold approach to probabilistic semantics:

1. The user might have to guess a threshold and the reply will be 'yes' or 'no'; thus, repeated checks are necessary to find out the critical region where 'yes' answers turn into 'no' replies and vice versa; in safety-critical applications this objection might be weak since one could argue that thresholds ought to be extremely close to 1 anyway. But, if we think of typical measures in

performance analysis, like the utilization of some resource, we would have to rely on probabilities, or percentages, anywhere in between 0 and 1. Finding critical regions of thresholds would then be non-trivial.

2. There is no compelling argument for considering only global thresholds: the level of safety required or the performance with respect to some measure is typically a function of the state being considered and leads to probabilistic logics where one considers *vectors of thresholds*.

3. Some methods for computing the probabilities of path formulas $\phi_1 \mathbf{U} \phi_2$ are non-incremental: this is in contrast to e.g. our non-standard semantics of CTL, presented in this paper, which computes fixed-points *incrementally* by iterating the meaning of the underlying recursion, and so the finite approximations of least/greatest fixed-points increase/decrease the information computed so far, thus allowing to compute approximate solutions to precise analytical probabilities.

4. The threshold approach is most sensitive to numerical errors close to the threshold region: thresholds are dis-continuous cut-off points.

5. If models are still in the design phase one might have very little intuition about the actual performance of such systems with respect to various (probabilistic) performance measures; thus, having to begin with any 'random' threshold seems unacceptable if probabilistic logics should become an active part in the design of stochastic systems.

Clearly, an alternative to threshold semantics would have to render replies which are of a *quantitative* rather than a qualitative kind. In this paper we demonstrate that if we drop thresholds altogether from PCTL following [20, 14] then we are essentially looking at the ordinary temporal logic CTL but now CTL formulas are no longer interpreted as predicates (subsets of states) but as functions $[\![\phi]\!]: S \to [0,1]$ from states into the unit interval $[0,1]$, termed 'expectations' in [20]. The difficulty and weakness of such semantics lies in their interpretation of the logical conjunction and disjunction to the effect that $[\![\phi]\!]\, s$ is no longer an exact probability but a 'sound' estimate thereof.

The main effort of the work presented here consists of establishing a firm link between these two semantic paradigms. Thus, one really ought to see these approaches as being *complementary* rather than conflicting. Based on [9, 8, 1] we also develop a symbolic model checking algorithm for our non-standard interpretation of the modal mu-calculus fragment of alternation depth ≤ 1.

2 THE MODAL MU-CALCULUS ON FINITE MARKOV CHAINS

A finite Markov chain [15] is a pair $(S, \mathbf{P})$, where S is a finite set of n states and $\mathbf{P} = (\mathbf{P}(s, s'))$ is a $n \times n$-matrix of entries in $[0,1]$ such that $\mathbf{P}(s, s')$ is the probability of the state transition $s \to s'$. Of course, we require that $\sum_{s' \in S} \mathbf{P}(s, s') = 1$ for all $s \in S$. Our approach allows us to analyze systems where this sum is strictly less than one (= sub-stochastic matrix), in which case we interpret the difference of $\sum_{s' \in S} \mathbf{P}(s, s')$ to 1 as the probability of the system deadlocking in state s. We could also have considered *families* of such sub-stochastic matrices $\mathbf{P}_a$ indexed by actions $a \in \mathsf{Act}$. However, we prefer the simpler presentation based on a single action $*$ for the purpose of illustration. Given sets C of constants and Var of variables such that Var is countable we then obtain a version of Kozen's modal mu-calculus $\mu\mathtt{M}$ [17] by considering the following grammar:

$$\phi ::= \mathtt{tt} \mid \mathtt{ff} \mid c_i \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \langle * \rangle \phi \mid [*]\phi \mid x \mid \mu x.\phi \mid \nu x.\phi$$

with $c_i \in \mathsf{C}$ and $x \in \mathsf{Var}$ and where recursion $\mu x.\phi$ and $\nu x.\phi$ is confined to formally monotone ϕ. We also consider the *positive* fragment of the modal mu-calculus $\mu\mathtt{M}^+$ by dropping the clause for $\neg\phi$ and for $\nu x.\phi$. Given a finite Markov chain $(S, \mathbf{P})$ we may interpret any $\phi \in \mu\mathtt{M}$ in the following way: the meaning $[\![\phi]\!]$ maps an environment ρ to a function $[\![\phi]\!]\rho \colon S \to [0,1]$; thus, ρ is of type $\mathsf{Var} \to S \to [0,1]$. Since S is assumed to have n elements we may think of any $f \colon S \to [0,1]$ as a vector $\mathbf{q} \in [0,1]^n$. Thus, we may now express our semantics, given already in [14], as a transformation of vectors:

$$\begin{array}{ll}
[\![\mathtt{tt}]\!]\rho\,\mathbf{q} = \mathbf{1} & [\![\mathtt{ff}]\!]\rho\,\mathbf{q} = \mathbf{0} \\
[\![c_i]\!]\rho\,\mathbf{q} = \mathbf{c_i} & [\![\neg\phi]\!]\rho\,\mathbf{q} = \mathbf{1} - ([\![\phi]\!]\rho\,\mathbf{q}) \\
[\![\phi_1 \wedge \phi_2]\!]\rho\,\mathbf{q} = \max(\mathbf{0}, [\![\phi_\mathbf{1}]\!]\rho\,\mathbf{q} + [\![\phi_\mathbf{2}]\!]\rho\,\mathbf{q} - \mathbf{1}) & [\![\phi_1 \vee \phi_2]\!]\rho = [\![\neg(\neg\phi_1 \wedge \neg\phi_2)]\!]\rho \\
[\![\langle * \rangle \phi]\!]\rho\,\mathbf{q} = \mathbf{P} \cdot ([\![\phi]\!]\rho\,\mathbf{q}) & [\![[*]\phi]\!]\rho = [\![\neg\langle * \rangle \neg\phi]\!]\rho \\
[\![\mu x.\phi]\!]\rho = \mathsf{fix}\,\mathbf{F}_{[\phi,x]} & [\![\nu x.\phi]\!]\rho = \mathsf{FIX}\,\mathbf{F}_{[\phi,x]}
\end{array}$$

where $\mathbf{P} \cdot \mathbf{q}$ is the usual multiplication of the matrix $\mathbf{P}$ with the vector $\mathbf{q}$; $\mathbf{0}$ is the zero vector and $\mathbf{1}$ is the vector $(1, 1, \ldots, 1)$. Furthermore, $\mathbf{F}_{[\phi,x]} \colon (S \to [0,1]) \to (S \to [0,1])$ is the monotone function defined by $\mathbf{F}_{[\phi,x]}\, f = [\![\phi]\!]\rho[x \mapsto f]$ and $\mathsf{fix}\,\mathbf{F}_{[\phi,x]}$ is the least, $\mathsf{FIX}\,\mathbf{F}_{[\phi,x]}$ the greatest fixed-point of $\mathbf{F}_{[\phi,x]}$. Finally, for the motivation of this semantics for $\wedge$ see [14] (this idea was taken from [24]); in summary, our semantics of conjunction guarantees that we never compute meanings $[\![\phi_1 \wedge \phi_2]\!]s$ which are greater than the value $\mu_s(\phi_1 \wedge \phi_2)$ where μ_s is *any* probability measure on the Borel algebra generated by the lattice of predicates of Kozen's standard interpretation of $\mu\mathtt{M}$. Note that the meaning of c_i can be any (fixed) function $f_{c_i} \colon S \to [0,1]$. In particular, if the image of f_{c_i} is in $\{0,1\}$ for all $c_i \in \mathsf{C}$ then we obtain ordinary *labeled* Markov chains $(S, \mathbf{P}, L)$ where $L(s) = \{c_i \in \mathsf{C} \mid f_{c_i}(s) = 1\}$ with $L \colon S \to \mathcal{P}(\mathsf{C})$. We shall abbreviate $\max(0, a + b - 1)$ by $0 \vee (a + b - 1)$, and likewise its dual $\min(1, a + b)$ by $1 \wedge (a + b)$.

3 MODEL CHECKING $\mu\mathtt{M}_{\leq 1}$ ON FINITE MARKOV CHAINS

We now derive an algorithm for model checking modal mu-calculus formulas ϕ without nested fixed-points [4] based on the semantics defined in the previous section. For that purpose we consider a fragment of the modal mu-calculus, where we restrict the clauses $\mu x.\phi$ and $\nu x.\phi$ to those ϕ which have at most x as a free variable. We call that fragment $\mu\mathtt{M}_{\leq 1}$ (alternation depth ≤ 1 in [11]).

Proposition 1 *Let $\sigma x.\phi$ be in $\mu\mathtt{M}_{\leq 1}$, where $\sigma \in \{\mu, \nu\}$. The function $\mathbf{F}_{[\phi,x]}$, of type $(S \to [0,1]) \to (S \to [0,1])$, defined above is continuous as a map from $\mathbf{R}^n \to \mathbf{R}^n$ in the Euclidean topology. This is also true for other interpretations of conjunction and disjunction which are continuous in the Euclidean topology, e.g. if conjunction is interpreted as minimum or multiplication.*

Proof: We prove the statement above by showing that all functions $\mathbf{F}_{[\psi,x]}$ are continuous where ψ is a subformula of ϕ; this argument proceeds in a bottom-up fashion on the parse tree of ϕ:

First, if $\psi \equiv x$ then $\mathbf{F}_{[\psi,x]}$ is the identity function which is continuous. Second, if x is not free in ψ then $\mathbf{F}_{[\psi,x]}$ is constant, and therefore continuous, for $[\![\psi]\!]\rho = [\![\psi]\!]\rho'$ if ρ and ρ' agree on ψ's free variables.

Third, if $\psi \equiv \neg\psi_1$ then $\mathbf{F}_{[\neg\psi_1,x]}(\mathbf{q}) = \mathbf{1} - \mathbf{F}_{[\psi_1,x]}(\mathbf{q})$ from which we infer that $\mathbf{F}_{[\neg\psi_1,x]}$ is continuous since vector subtraction is continuous.

Fourth, if $\psi = \langle * \rangle\psi_1$ then by construction and by the semantics of $\langle * \rangle$ we get $\mathbf{F}_{[\langle * \rangle\psi_1,x]}(\mathbf{q}) = \mathbf{P} \cdot (\mathbf{F}_{[\psi_1,x]}\,\mathbf{q})$; since $\mathbf{F}_{[\psi_1,x]}$ and matrix multiplication are continuous we infer that $\mathbf{F}_{[\langle * \rangle\psi_1,x]}$ is continuous as well.

Fifth, if $\psi = [*]\psi_1$ then by construction and by the semantics of $[*]$ we get $\mathbf{F}_{[[*]\psi_1,x]}(\mathbf{q}) = (\mathbf{1} - \mathbf{P} \cdot \mathbf{1}) + \mathbf{F}_{[\langle * \rangle\psi_1,x]}(\mathbf{q})$. Since $\mathbf{F}_{[\langle * \rangle\psi_1,x]}$ and the addition with a constant vector are continuous, we see that $\mathbf{F}_{[[*]\psi_1,x]}$ is continuous, too; note that $\mathbf{1} - \mathbf{P} \cdot \mathbf{1} = \mathbf{0}$ if P is a stochastic matrix, i.e. if the process modeled by $(S, \mathbf{P})$ is free of divergence, so in that case the semantics of $\langle * \rangle$ and $[*]$ coincide.

Sixth, if $\psi \equiv \psi_1 \wedge \psi_2$ we have $\mathbf{F}_{[\psi_1 \wedge \phi_2,x]}(\mathbf{q}) = \mathbf{0} \vee (\mathbf{F}_{[\psi_1,x]}(\mathbf{q}) + \mathbf{F}_{[\psi_2,x]}(\mathbf{q}) - \mathbf{1})$ from which we can infer the continuity of $\mathbf{F}_{[\psi_1 \wedge \phi_2,x]}$ since $\mathbf{F}_{[\psi_1,x]}$, $\mathbf{F}_{[\psi_2,x]}$, $-$ and $\vee$ are continuous on $\mathbf{R}^n$.

Seventh, if $\psi \equiv \psi_1 \vee \psi_2$ then this is similar to the previous case, for $\mathbf{F}_{[\psi_1 \vee \phi_2,x]}(\mathbf{q}) = \mathbf{1} \wedge (\mathbf{F}_{[\psi_1,x]}(\mathbf{q}) + \mathbf{F}_{[\psi_2,x]}(\mathbf{q}))$.

Finally, note that this concludes all possible cases since ϕ has at most x as a free variable; thus, alternation depth ≤ 1 excludes the cases $\psi \equiv \mu y.\psi_1$ and $\psi \equiv \nu y.\psi_1$, where y is free in ψ_1. End of Proof.

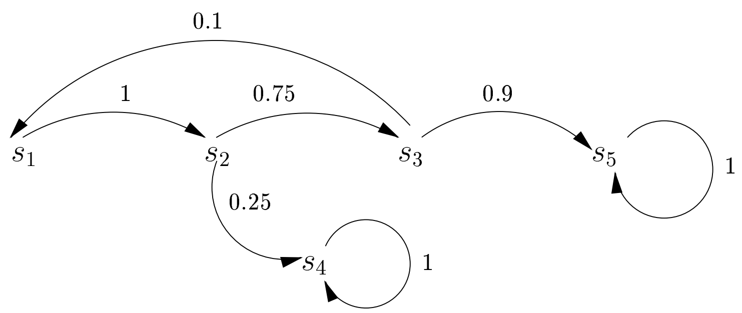
We can exploit the fact that all functions $\mathbf{F}_{[\phi,x]}$ are continuous maps to guarantee that fixed-point iterations on the infinite domain $S \to [0,1]$ reach a fixed-point at ω. This is a non-trivial result since it can not necessarily be inferred from the standard facts about the continuity of $\mathsf{fix}\colon (S \to [0,1]) \to (S \to [0,1])$ since the semantics of subformulas $\nu x.\eta$, via FIX, might give rise to meanings which are not Scott-continuous.

Proposition 2 *Let $\mu x.\phi$ be in $\mu\mathtt{M}_{\leq 1}$. Then $\bigvee_{0\leq m<\omega} {\mathbf{F}_{[\phi,x]}}^m\,\mathbf{0}$ is the least fixed-point of $\mathbf{F}_{[\phi,x]}$. Dually, if $\nu x.\phi$ is in $\mu\mathtt{M}_{\leq 1}$ then $\bigwedge_{0\leq m<\omega} {\mathbf{F}_{[\phi,x]}}^m\,\mathbf{1}$ is the greatest fixed-point of $\mathbf{F}_{[\phi,x]}$. In fact, the above holds for all monotone interpretations of conjunction and disjunction which are continuous in the Euclidean topology.*

Proof: The argument is, in spirit, the same as the usual one for least fixed-points of Scott-continuous functions. However, we need to adapt this argument since the function $\mathbf{F}_{[\phi,x]}$ is not Scott-continuous in general. But directed suprema and filtered infima in $[0,1]^n$ are pointwise, so we see that the supremum and the infimum in $S\to[0,1]$ are also limits in the Euclidean topology. Thus, we infer that $\mathbf{F}_{[\phi,x]}(\bigvee_{0\leq n<\omega} {\mathbf{F}_{[\phi,x]}}^n\,\mathbf{0}) = \mathbf{F}_{[\phi,x]}(\lim_{0\leq n<\omega} {\mathbf{F}_{[\phi,x]}}^n\,\mathbf{0}) = \lim_{0\leq n<\omega} {\mathbf{F}_{[\phi,x]}}^{n+1}\,\mathbf{0} = \bigvee_{1\leq n<\omega} {\mathbf{F}_{[\phi,x]}}^n\,\mathbf{0} = \bigvee_{0\leq n<\omega} {\mathbf{F}_{[\phi,x]}}^n\,\mathbf{0}$ as $\mathbf{F}_{[\phi,x]}$ is monotone and continuous in the Euclidean topology. The argument for the greatest fixed-point is dual to this one. End of Proof.

We use a simple example of a probabilistic protocol to demonstrate our approach. For simplicity we merely consider the states of the protocol pertaining to the transmission of one message only. Initially (state s_1), the protocol will *send* the message to the medium. The *medium* (state s_2) will attempt to transmit the message to the *receiver* (state s_3), which is possible with probability 0.75; with probability 0.25 the message will get *lost* (state s_4). Upon receiving the message, the receiver will attempt to *acknowledge* it. If the acknowledgment is received intact, which is possible with probability 0.9 then the message is *delivered* (state s_5) and the protocol terminates; failure to receive acknowledgment causes the protocol to resend the message again (return to state s_1). Again, for simplicity we shall assume an unbounded number of retries. The example can be easily extended to cater for a bounded number of retries. To model a distributed protocol it may be necessary to go beyond Markov chains.

The diagram below represents the protocol.



The Markov chain matrix for the above protocol is given below.

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0.75 & 0.25 & 0 \\ 0.1 & 0 & 0 & 0 & 0.9 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The property we would like to calculate is: *what is the probability of delivering the message?* This can be expressed as $\mu x.(c_{delv} \vee \langle * \rangle x)$, where $c_{delv} = (0,0,0,0,1)$; in other words, $c_{delv} = \mathtt{tt}$ at the state s_5 and $\mathtt{ff}$ at all other states. The successive iterations starting from $(0,0,0,0,0)$ produce:

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0.9 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0.675 \\ 0.9 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0.675 \\ 0.675 \\ 0.9 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0.675 \\ 0.675 \\ 0.9675 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0.675 \\ 0.725625 \\ 0.9675 \\ 0 \\ 1 \end{pmatrix},$$

$$\begin{pmatrix} 0.725625 \\ 0.725625 \\ 0.9675 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0.725625 \\ 0.725625 \\ 0.9725625 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0.725625 \\ 0.7294218 \\ 0.9725625 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0.7294218 \\ 0.7294218 \\ 0.9725625 \\ 0 \\ 1 \end{pmatrix}, \ldots$$

Thus, the probability of the message being eventually delivered is at least 0.729 (up to 3 significant digits, which one further iteration would reveal); this follows from Corollary 1 in Section 5.

4 SYMBOLIC PROBABILISTIC MODEL CHECKING FOR $\mu\mathtt{M}_{\leq 1}$

We now present a symbolic model checking algorithm for all closed $\phi \in \mu\mathtt{M}_{\leq 1}$ which computes expectations of formulas for a given labeled Markov chain without constructing the state space explicitly. This algorithm represents $[\![\phi]\!]$ symbolically as a Multi-Terminal Binary Decision Diagram; for more details on symbolic model checking with Multi-Terminal Binary Decision Diagrams (MTBDDs) we refer the reader to [9, 8, 1].

Binary-Decision-Diagram-based model checking [21] has proved effective in practice: it tackles the state explosion problem by providing a symbolic representation for the state space. With MTBDDs one can extend existing automatic verification tools to cater for quantitative information contained in the models such as probability [1]. The MTBBD representation is compact and no less efficient than sparse matrix techniques [8].

Given a closed formula $\phi \in \mu\mathtt{M}_{\leq 1}$ the basic idea is to represent the function $[\![\phi]\!]: S \to [0,1]$ symbolically by a Multi-Terminal Binary Decision Diagram $\mathcal{M}[\phi]$. We define this MTBDD by structural induction on ϕ. However, we will only compute *approximate* results for fixed-points $\phi \equiv \sigma x.\psi$ since we cannot iterate $\mathbf{F}_{[\psi,x]}$ forever. We use a function $\mathsf{ITERATE}(A, Q, E)$ which requires its first argument A to represent a map from $\mathbf{R}^n$ to $\mathbf{R}^n$, Q to represent an element of $\mathbf{R}^n$, and E a real number $\varepsilon > 0$. The code for this function is similar to that of fixed-point approximation for the Boolean symbolic model checking algorithm [5] except that we enforce termination w.r.t. ε as follows: $\mathsf{ITERATE}(A, Q, E)$ returns the MTBDD which represents the vector $A^{m+1}Q$ where m is the least natural number such that $\sum_{i=1}^{n} \ |(A^{m+1}Q)_i - (A^m Q)_i| < \varepsilon$. Naturally, termination criteria other than the above are conceivable.

The model checking algorithm will, of course, need to work with the MTBDD representation of the $n \times n$ probability transition matrix $\mathbf{P}$. For $k = \lceil \log n \rceil$ we may identify the $2^k \times 2^k$ matrix $\mathbf{P}$ with the function $f : \{0,1\}^{2k} \to [0,1]$ where $f(x_1, y_1, \ldots, x_k, y_k)$ is the value of $\mathbf{P}$ in the $(i+1)$-th row and $(j+1)$-th column and $i = \sum_{u=1}^{k} \ x_u \cdot 2^{k-u}, \ \ j = \sum_{v=1}^{k} \ y_v \cdot 2^{k-v}$. Thus, $\mathbf{P}$ can be identified with a MTBDD $\mathtt{P}$ over $(x_1, y_1, \ldots, x_k, y_k)$. Similarly, each vector $\mathbf{q}$ with 2^k arguments can be identified with the MTBDD $\mathtt{q}$ over $(x_1, \ldots, x_k)$ for the function $F : \{0,1\}^n \to [0,1]$ where $F(x_1, \ldots, x_n)$ is the $(i+1)$-th argument of $\mathbf{q}$ and where i is as above. The remaining operators needed for the algorithm, both from [8], are: $\mathsf{MV\_MULTI}(\mathtt{P}, \mathtt{q})$ (matrix-by-vector multiplication), and $\mathsf{APPLY}(\cdot)$ (elementwise application of the binary operator *op* on real numbers, such as addition, to two MTBDDs).

Let ϕ be a closed formula in $\mu\mathtt{M}_{\leq 1}$ and $(S, \mathbf{P}, L)$ a labeled Markov chain. We assume that constants c_i can only take values in $\{0,1\}$ and that there are 'enough' of them so that we can identify each state s with the Boolean vector $(x_1, x_2, ..., x_n)$ where $x_i = 1$ iff $c_i \in L(s)$ (an extension to arbitrary functions is straightforward). We define an MTBDD over the variables $\overline{x} = (x_1, x_2, ..., x_n)$ by structural induction on ϕ:

1. $\mathcal{M}[\mathtt{tt}] = \mathtt{TRUE}$, the (unique) MTBDD representing the constant function 1 which consists of a terminal vertex labeled with value 1,

2. $\mathcal{M}[\mathtt{ff}] = \mathtt{FALSE}$, the (unique) MTBDD representing the constant function 0 which consists of a terminal vertex labeled with value 0,

3. $\mathcal{M}[c_i] = \lambda\bar{x}.[x_i]$, the (unique) MTBDD representing the function $f(\overline{x}) = x_i$, i.e. the MTBDD with one nonterminal vertex labeled with x_i and two terminal vertices, the left labeled with value 0 and right with value 1,

4. $\mathcal{M}[\neg\phi] = \mathsf{APPLY}(\mathtt{TRUE}, \mathcal{M}[\phi], -)$, the MTBDD obtained from applying the operator $(1 - \cdot)$ to the MTBDD for ϕ,

5. $\mathcal{M}[\phi_1 \wedge \phi_2]$ is the MTBDD obtained by applying our conjunction semantics to the MTBDDs for ϕ_1 and ϕ_2, where $\mathtt{FALSE}$ is the MTBDD representing the constant function 0: $\mathsf{APPLY}(\mathsf{APPLY}(\mathsf{APPLY}(\mathcal{M}[\phi_1], \mathcal{M}[\phi_2], \cdot + \cdot), \mathtt{TRUE}, \cdot - \cdot), \mathtt{FALSE}, \max)$,

6. $\mathcal{M}[\phi_1 \vee \phi_2] \ = \ \mathsf{APPLY}(\mathsf{APPLY}(\mathcal{M}[\phi_1], \mathcal{M}[\phi_2], \cdot + \cdot), \mathtt{TRUE}, \min)$ is ob-

tained similarly to the above except that the operation is our semantic disjunction: $\min(1, \cdot + \cdot)$,

7. $\mathcal{M}[\langle * \rangle \phi] = \mathsf{MV\_MULTI}(\mathtt{P}, \mathcal{M}[\phi])$, the MTBDD obtained as a result of a call to the matrix-by-vector multiplication operation with the MTBDD P representing the Markov transition probability matrix and the MTBDD for ϕ passed as arguments,

8. $\mathcal{M}[[*]\phi] = \mathcal{M}[\neg\langle * \rangle \neg \phi]$, the MTBDD for the form equivalent to $[*]\phi$,

9. $\mathcal{M}[\mu x.\phi] = \mathsf{ITERATE}(\mathcal{F}_{[\phi,x]}, \mathtt{FALSE}, \mathtt{EPSILON})$, the MTBDD obtained as a result of a call to the fixed-point iteration algorithm with the MTBDDs for $\mathcal{F}_{[\phi,x]}$ and FALSE passed as arguments,

10. $\mathcal{M}[\nu x.\phi] = \mathsf{ITERATE}(\mathcal{F}_{[\phi,x]}, \mathtt{TRUE}, \mathtt{EPSILON})$, similar to the above except that the iteration computes a decreasing sequence of approximations starting with 1.

We should stress that the last clause is actually computing *upper* bounds. This points out the fundamental weakness of our results as presented here. One needs to develop a suitable semantic framework which allows the simultaneous consideration of lower and upper bounds, and then clarify how negation and greatest fixed point act thereon.

The symbol P above is the MTBDD representing $\mathbf{P}$ and EPSILON represents some $\varepsilon > 0$. Of course, this requires that we give symbolic representations $\mathcal{F}_{[\phi,x]}$ of the continuous functions $\mathbf{F}_{[\phi,x]}$ as MTBDDs; such a definition simply copies the constructive argument of Proposition 1:

$$
\begin{aligned}
\mathcal{F}_{[x,x]} &= \mathtt{I}, \text{ the MTBDD representing the identity matrix,} \\
\mathcal{F}_{[\psi,x]} &= \lambda\bar{x}.\mathcal{M}[\psi] \text{ if } x \text{ is not free in } \psi, \\
\mathcal{F}_{[\neg\psi,x]} &= \lambda\bar{x}.\mathsf{APPLY}(\mathtt{TRUE}, \mathcal{F}_{[\psi,x]}\,\bar{x}, -), \\
\mathcal{F}_{[\langle * \rangle\psi,x]} &= \lambda\bar{x}.\mathsf{MV\_MULTI}(\mathtt{P}, \mathcal{F}_{[\psi,x]}\,\bar{x}), \\
\mathcal{F}_{[[*]\psi,x]} &= \lambda\bar{x}.\mathsf{APPLY}(\mathsf{APPLY}(\mathtt{TRUE}, \mathsf{MV\_MULTI}(\mathtt{P}, \mathtt{TRUE}), -), \\
&\qquad \mathsf{MV\_MULTI}(\mathtt{P}, \mathcal{F}_{[\psi,x]}\,\bar{x}), +), \\
\mathcal{F}_{[\psi_1\wedge\psi_2,x]} &= \lambda\bar{x}.\mathsf{APPLY}(\mathtt{FALSE}, \\
&\qquad \mathsf{APPLY}(\mathsf{APPLY}(\mathcal{F}_{[\psi_1,x]}\,\bar{x}, \mathcal{F}_{[\psi_2,x]}\,\bar{x}, +), \mathtt{TRUE}, -), \max), \\
\mathcal{F}_{[\psi_1\vee\psi_2,x]} &= \lambda\bar{x}.\mathsf{APPLY}(\mathsf{APPLY}(\mathcal{F}_{[\psi_1,x]}\,\bar{x}, \mathcal{F}_{[\psi_2,x]}\,\bar{x}, +), \min).
\end{aligned}
$$

5 COMPARING CTL TO PCTL

The semantics of the modal mu-calculus given above for finite labeled Markov chains is *fuzzy* [27] in its interpretation of the logical connectives $\wedge$ and $\vee$; yet, it also employs *probabilistic* meaning as in the weighted average for the modalities $\langle * \rangle$ and $[*]$. This unusual combination warrants a comparison with purely probabilistic semantics which we now develop. First of all, we saw that our semantics identifies $\langle * \rangle$ and $[*]$ if the matrix $\mathbf{P}$ of the underlying model is stochastic which we assume throughout this section. This leads to a simplified grammar for CTL viewed as a subset of $\mu\mathtt{M}_{\leq 1}$ [4, 21]: $\mathbf{AX}\phi$ and $\mathbf{EX}\phi$ correspond to $\langle * \rangle\phi$, $\mathbf{A}(\phi_1 \mathbf{U}\, \phi_2)$ and $\mathbf{E}(\phi_1 \mathbf{U}\, \phi_2)$ translate to $\mu x.\phi_2 \vee (\phi_1 \wedge \langle * \rangle x)$, and $\mathbf{AG}\phi$ and $\mathbf{EG}\phi$ turn into $\nu x.\phi \wedge \langle * \rangle x$.

Based on our semantics for finite Markov chains we may therefore reduce the image of CTL in $\mu\mathtt{M}_{\leq 1}$ to the following, *semantically driven*, grammar:

$$\phi ::= \mathtt{tt} \mid \mathtt{ff} \mid c_i \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \mathbf{EX}\phi \mid \mathbf{E}(\phi_1 \mathbf{U} \phi_2) \mid \mathbf{EG}\phi$$

where $c_i \in C$. A straightforward mathematical induction shows that any closed ϕ generated by this grammar is of alternation depth ≤ 1. Of course, not every closed $\phi \in \mu\mathtt{M}_{\leq 1}$ is generated by the grammar above; e.g. the formula $\nu x.\psi \wedge \langle * \rangle\langle * \rangle x$ which qualitatively expresses the property that ψ occurs every other time cannot be specified in CTL [26]. Note that the grammar above still has some redundancy in the presence of negation, but we intend to have separate clauses for $\vee$ and $\wedge$ as well as for $\mathtt{tt}$ and $\mathtt{ff}$ since we intend to show results for the fragment of CTL without the clauses $\neg\phi$ and $\mathbf{EG}\phi$; we denote the resulting fragment of CTL by CTL^+.

5.1 Translating CTL$^+$ into PCTL

We now introduce the syntax and semantics of PCTL [13, 12] interpreted over labeled Markov chains. Let $(S, \mathbf{P}, L)$ be a labeled Markov chain. A *path* in such a chain is a nonempty (finite or infinite) sequence $\pi = s_0 s_1 s_2, \ldots$ where $s_i \in S$ and $\mathbf{P}(s_{i-1}, s_i) > 0$ for $i = 1, 2, \ldots$. We denote by $\mathsf{Path}_\omega(s)$ the set of all infinite paths starting in the state s and endow $\mathsf{Path}_\omega(s)$ with a σ-algebra structure by taking the smallest σ-algebra on $\mathsf{Path}_\omega(s)$ which contains the basic cylinders $\{\pi \in \mathsf{Path}_\omega(s) \mid u \text{ is a prefix of } \pi\}$ for finite paths u starting in s. Each such basic cylinder corresponds to the event of executing the finite path u. The probability measure $Prob$ on the σ-algebra is the unique measure with $Prob\{\, \pi \in \mathsf{Path}_\omega(s) \mid u \text{ is a prefix of } \pi \,\} = \mathbf{P}(u)$ where $\mathbf{P}(s_0 s_1 \ldots s_k) = \mathbf{P}(s_0, s_1) \cdot \mathbf{P}(s_1, s_2) \cdot \ldots \cdot \mathbf{P}(s_{k-1}, s_k)$ [25].

PCTL [13], the aforementioned probabilistic extension of CTL, contains atomic propositions and the operators next-step $\mathbf{X}$ and until $\mathbf{U}$ decorated with probabilities (we omit the bounded `until` to simplify the presentation). Formally, the syntax of PCTL is as follows:

$$\phi \quad ::= \quad \mathtt{tt} \mid c_i \mid \phi_1 \wedge \phi_2 \mid \neg\phi \mid [\, \mathbf{X}\, \phi\,]_{\sqsupseteq p} \mid [\, \phi_1 \mathbf{U} \phi_2\,]_{\sqsupseteq p}$$

where c_i is an atomic proposition, $p \in [0,1]$, $\sqsupseteq$ is either $\geq$ or $>$. The satisfaction relation $\models\, \subseteq\, S \times \mathrm{PCTL}$ for state formulas is given as usual:

$s \models \mathtt{tt}$ for all $s \in S$ $\qquad$ $s \models c_i$ iff $c_i \in L(s)$

$s \models \phi_1 \wedge \phi_2$ iff $s \models \phi_i$ for $i = 1, 2$ $\qquad$ $s \models \neg\phi$ iff $s \not\models \phi$

In contrast, probabilistic formulas $[\mathbf{X}\,\phi]_{\sqsupseteq p}$ or $[\phi_1 \mathbf{U}\ \phi_2]_{\sqsupseteq p}$, where ϕ, ϕ_1, ϕ_2 are PCTL formulas, are interpreted with the help of the probability measure *Prob* over the σ- algebra of the infinite paths of the labeled Markov chain:

$$
\begin{array}{ll}
s \models [\mathbf{X}\,\phi]_{\sqsupseteq p} & \text{iff}\quad Prob\{\pi \in \mathsf{Path}_\omega(s) \mid \pi \models \mathbf{X}\ \ \phi\} \sqsupseteq p \\
s \models [\phi_1 \mathbf{U}\,\phi_2]_{\sqsupseteq p} & \text{iff}\quad Prob\{\pi \in \mathsf{Path}_\omega(s) \mid \pi \models \phi_1 \mathbf{U}\,\phi_2\} \sqsupseteq p
\end{array}
$$

where $\pi \models \mathbf{X}\,\phi$ iff the first state of π satisfies ϕ, and $\pi \models \phi_1 \mathbf{U} \phi_2$ iff there exists $k \geq 0$ such that ϕ_1 is satisfied at all states of π up to and including $k-1$ and ϕ_2 is satisfied at k. The operators of disjunction and implication can be derived. One can also derive probabilistic analogues of the linear operators 'eventually' and 'always'; these are weaker than their classical linear counterparts since events that happen with probability 1 are not necessarily always guaranteed to happen.

Given a formula $\phi \in \mathrm{CTL}^+$ we translate it into a formula $\phi^\flat \in \mathrm{PCTL}$ such that the annotated probability thresholds in $\phi^\flat$ are the corresponding quantitative evidences computed by our semantics. We then show that $[\![\phi]\!]s > 0$ implies $s \models \phi^\flat$. In order to unfold this argument in the sharpest sense possible we slightly generalize PCTL: if ψ is a path formula and $\psi_{\sqsupseteq p}$ is an ordinary PCTL formula with $p \in [0,1]$ we now also allow $\psi_{\sqsupseteq \mathbf{p}}$ to be a formula of PCTL where $\mathbf{p}$ is a *vector of thresholds* and therefore in $[0,1]^n$. Of course, this only makes sense for models $(S, \mathbf{P}, L)$ with a fixed number n of states. It should also be noted that this translation is dependent on a fixed model since we use the non-standard semantics of this model to compute thresholds needed for the translated formulas, but these constraints are not crucial since our objective is to implement model checking algorithms which work on concrete labeled Markov chains.

Since we generalized thresholds to *vectors of thresholds* we have to give a semantics to those extensions. This works as expected, e.g. $s \models [\phi_1 \mathbf{U}\,\phi_2]_{\sqsupseteq \mathbf{p}}$ iff $Prob\{\pi \in \mathsf{Path}_\omega(s) \mid \pi \models \phi_1 \mathbf{U}\,\phi_2\} \sqsupseteq p_s$ where $\mathbf{p} = (p_s)_{s \in S}$. This generalization is not necessary since we could use the minimum of all the quantitative evidences $[\![\phi]\!]s$, $s \in S$, as a threshold p for the usual PCTL formulas, but a vector semantics for PCTL makes our semantic comparison tighter.

Fixing a model $(S, \mathbf{P}, L)$, we translate closed expressions of CTL^+ into PCTL based on our semantics. The translation is a map ${}^\flat \colon \mathrm{CTL}^+ \to \mathrm{PCTL}$ defined by structural induction on ϕ:

$$
\begin{array}{ll}
\mathtt{tt}^\flat = \mathtt{tt} & \mathtt{ff}^\flat = \mathtt{ff} \\
c_i^\flat = c_i & (\phi_1 \wedge \phi_2)^\flat = \phi_1^\flat \wedge \phi_2^\flat \\
(\phi_1 \vee \phi_2)^\flat = \neg(\neg(\phi_1^\flat) \wedge \neg(\phi_2^\flat)) & (\mathbf{EX}\,\phi)^\flat = [\mathbf{X}\,(\phi^\flat)]_{\geq \mathbf{p}} \\
(\mathbf{E}(\phi_1 \mathbf{U} \phi_2))^\flat = [(\phi_1^\flat)\mathbf{U}\,(\phi_2^\flat)]_{\geq \mathbf{q}} &
\end{array}
$$

where $\mathbf{p} = [\![\langle * \rangle \phi]\!]$ and $\mathbf{q} = [\![\mu x.\phi_2 \vee (\phi_1 \wedge \langle * \rangle x)]\!]$.

5.2 SOUNDNESS OF OUR TRANSLATION

Definition 1 *For closed $\phi \in \mu\mathtt{M}$ let $\mathsf{Pos}(\phi)$ be the set of all $s \in S$ such that $[\![\phi]\!]s > 0$. Given $\eta \in PCTL$, let $\mathsf{Sat}(\eta)$ denote the set of all $s \in S$ such that $s \models \eta$.*

The role of $\mathsf{Pos}(\cdot)$ is that it mediates between the unit interval $[0,1]$ and the domain of truth values in that we identify 0 with `FALSE` and any positive value with `TRUE` if the root of the formula is $\mathtt{tt}$, $\mathtt{ff}$, c_i, $\wedge$, or $\vee$.

Theorem 1 *We assume that CTL$^+$ and PCTL have the same interpretation of the constants $c_i \in C$ for a fixed model $(S, \mathbf{P}, L)$. Then for all $\phi \in CTL^+$ we have $\mathsf{Pos}(\phi) \subseteq \mathsf{Sat}(\phi^\flat)$.*

Proof: We use mathematical induction on the lexicographical ordering of fixed-point depth and rank of ϕ. That way we may reason by ordinary structural induction, but for the case of recursion where we use syntactic approximations. We implicitly identify CTL$^+$ formulas with their equivalent translations into $\mu\mathtt{M}_{\leq 1}$.

The cases $\phi \equiv \mathtt{tt}$ and $\phi \equiv \mathtt{ff}$ are clear and the case $\phi \equiv c_i$ follows from our assumption. If $\phi \equiv \phi_1 \wedge \phi_2$ then

$$\begin{aligned}
\mathsf{Pos}(\phi_1 \wedge \phi_2) &= \{s \in S \mid 0 \vee ([\![\phi_1]\!]s + [\![\phi_2]\!]s - 1) > 0\} \qquad (1)\\
&\subseteq \{s \in S \mid [\![\phi_1]\!]s > 0 \text{ and } [\![\phi_2]\!]s > 0\} \quad \text{since } [\![\phi_i]\!]s \in [0,1]\\
&= \mathsf{Pos}(\phi_1) \cap \mathsf{Pos}(\phi_2)\\
&\subseteq \mathsf{Sat}(\phi_1^\flat) \cap \mathsf{Sat}(\phi_2^\flat) \quad \text{by induction}\\
&= \mathsf{Sat}(\phi_1^\flat \wedge \phi_2^\flat) \quad \text{by the semantics of PCTL}\\
&= \mathsf{Sat}((\phi_1 \wedge \phi_2)^\flat) \quad \text{by def. of }^\flat.
\end{aligned}$$

Dually, if $\phi \equiv \phi_1 \vee \phi_2$ then

$$\begin{aligned}
\mathsf{Pos}(\phi_1 \vee \phi_2) &= \{s \in S \mid 1 \wedge ([\![\phi_1]\!]s + [\![\phi_2]\!]s) > 0\} \qquad (2)\\
&\subseteq \{s \in S \mid [\![\phi_1]\!]s > 0 \text{ or } [\![\phi_2]\!]s > 0\} \quad \text{since } [\![\phi_i]\!]s \in [0,1]\\
&= \mathsf{Pos}(\phi_1) \cup \mathsf{Pos}(\phi_2)\\
&\subseteq \mathsf{Sat}(\phi_1^\flat) \cup \mathsf{Sat}(\phi_2^\flat) \quad \text{by induction}\\
&= \mathsf{Sat}(\phi_1^\flat \vee \phi_2^\flat) \quad \text{by the semantics of PCTL}\\
&= \mathsf{Sat}((\phi_1 \vee \phi_2)^\flat) \quad \text{by def. of }^\flat.
\end{aligned}$$

As for the modalities, if $\phi \equiv \langle * \rangle \psi$ then

$$\mathsf{Pos}(\langle * \rangle \psi) = \{s \in S \mid \sum_{s' \in S} \mathbf{P}(s,s') \cdot ([\![\psi]\!]s') > 0\}. \qquad (3)$$

So let $s \in \mathsf{Pos}(\langle * \rangle \psi)$ be given. We need to show that $s \in \mathsf{Sat}([\mathbf{X}\,(\psi^\flat)]_{\geq \mathbf{p}})$ where $\mathbf{p} = [\![\langle * \rangle \psi]\!]$. Thus, we need to demonstrate the inequality $[\![\langle * \rangle \psi]\!] s \leq \sum_{s' \in \mathsf{Sat}(\psi^\flat)} \mathbf{P}(s, s')$. We compute

$$\begin{aligned}
[\![\langle * \rangle \psi]\!] s &= \sum_{s' \in S} \mathbf{P}(s, s') \cdot [\![\psi]\!] s' \\
&= \sum_{s' \in \mathsf{Pos}(\psi)} \mathbf{P}(s, s') \cdot [\![\psi]\!] s' \quad \text{as } [\![\psi]\!] s' \in [0,1], \\
&\leq \sum_{s' \in \mathsf{Sat}(\psi^\flat)} \mathbf{P}(s, s') \cdot [\![\psi]\!] s' \qquad \text{by ind.} \\
&\leq \sum_{s' \in \mathsf{Sat}(\psi^\flat)} \mathbf{P}(s, s') \qquad \text{as } [\![\psi]\!] s' \in [0,1].
\end{aligned} \tag{4}$$

Finally, if $\phi \equiv \mu x.\phi_2 \vee (\phi_1 \wedge \langle * \rangle x)$ we define syntactic approximations [4] η_m for all $0 \leq m < \omega$ such that

$$[\![\phi]\!] = \bigvee_{0 \leq m < \omega} [\![\eta_m]\!]. \tag{5}$$

We set

$$\begin{aligned}
\eta_0 &= \mathtt{ff} \\
\eta_{m+1} &= \phi_2 \vee (\phi_1 \wedge \langle * \rangle \eta_m) \quad \text{for all } m \geq 0.
\end{aligned} \tag{6}$$

Note that equation (5) holds since all semantic operations in our interpretation of CTL$^+$ are continuous and since we only compute least fixed-points thereof. Suppose that $[\![\mu x.\phi_2 \vee (\phi_1 \wedge \langle * \rangle x)]\!]\, s > 0$. We have to show that $s \models [\phi_1^\flat \mathbf{U}\, \phi_2^\flat]_{\geq \mathbf{p}}$, where $\mathbf{p} = [\![\mu x.\phi_2 \vee (\phi_1 \wedge \langle * \rangle x)]\!]$. If we set

$$q_s = \mathit{Prob}\{\pi \in \mathsf{Path}_\omega(s) \mid \pi \models \phi_1^\flat \mathbf{U}\, \phi_2^\flat\} \tag{7}$$

then we simply have to show that $p_s \leq q_s$. By equation (5), we know that $p_s = \bigvee_{0 \leq m \leq \omega} [\![\eta_m]\!] s$ and therefore it suffices to show that $[\![\eta_m]\!] s \leq q_s$ for all $0 \leq m < \omega$. For $m = 0$ this is true since $[\![\eta_0]\!] s = 0 \leq q_s$. Assuming that $[\![\eta_m]\!] s \leq q_s$ holds for m (and *all* $s \in S$!) we now demonstrate its validity for $m + 1$. We compute

$$\begin{aligned}
[\![\eta_{m+1}]\!] s &= [\![\phi_2 \vee (\phi_1 \wedge \langle * \rangle \eta_m)]\!] s \\
&= 1 \wedge ([\![\phi_2]\!] s + [0 \vee ([\![\phi_1]\!] s + [\![\langle * \rangle \eta_m]\!] s - 1)]),
\end{aligned} \tag{8}$$

for which we can safely assume that it be positive; otherwise, there is nothing to show. Thus, $[\![\phi_2]\!] s$ or $[\![\phi_1]\!] s + [\![\langle * \rangle \eta_m]\!] s - 1$ is positive. In the first case we may use our global induction hypothesis to infer that $s \models \phi_2^\flat$ which gives us $q_s = 1$ and so $[\![\eta_{m+1}]\!]s \leq q_s$ is clear. In the second case we may therefore assume

$$\begin{array}{rcl} s & \not\models & \phi_2^\flat \\ [\![\phi_2]\!]s & = & 0 \\ [\![\phi_1]\!]s + [\![\langle * \rangle \eta_m]\!]s - 1 & > & 0. \end{array} \tag{9}$$

Since $[\![\phi_1]\!]s + [\![\langle * \rangle \eta_m]\!]s - 1$ is positive we conclude that $[\![\phi_1]\!]s$ must be positive. Using our global induction hypothesis we infer that

$$s \models \phi_1^\flat. \tag{10}$$

Using (9) we can continue to unfold equation (8) as

$$\begin{array}{rcl} [\![\eta_{m+1}]\!]s & = & 1 \wedge ([\![\phi_2]\!]s + [0 \vee ([\![\phi_1]\!]s + [\![\langle * \rangle \eta_m]\!]s - 1)]) \\ & = & 1 \wedge (0 + [0 \vee ([\![\phi_1]\!]s + [\![\langle * \rangle \eta_m]\!]s - 1)]) \\ & = & 1 \wedge ([\![\phi_1]\!]s + [\![\langle * \rangle \eta_m]\!]s - 1) \\ & = & [\![\phi_1]\!]s + [\![\langle * \rangle \eta_m]\!]s - 1 \\ & = & ([\![\phi_1]\!]s - 1) + \sum_{s' \in S} \mathbf{P}(s, s') \cdot [\![\eta_m]\!]s' \\ & \leq & (1 - 1) + \sum_{s' \in S} \mathbf{P}(s, s') \cdot [\![\eta_m]\!]s' \\ & \leq & \sum_{s' \in S} \mathbf{P}(s, s') \cdot q_{s'} \qquad \text{by ind. hyp.} \\ & = & q_s \qquad \text{as } s \models \phi_1^\flat \wedge \neg \phi_2^\flat \text{ by (9) and (10).} \end{array} \tag{11}$$

End of Proof.

We should stress that this particular result remains valid for a range of interpretations of conjunction and disjunction as long as such interpretations are monotone, continuous, and satisfy $\mathsf{Pos}(\phi_1 \wedge \phi_2) \subseteq \mathsf{Pos}(\phi_1) \cap \mathsf{Pos}(\phi_2)$ as well as $\mathsf{Pos}(\phi_1 \vee \phi_2) \subseteq \mathsf{Pos}(\phi_1) \cup \mathsf{Pos}(\phi_2)$. For example, one might interpret $\wedge$ as minimum and $\vee$ as maximum which, incidentally, renders bigger thresholds for conjunction in general as $0 \vee (x + y - 1) \leq x \wedge y$ for all $x, y \in [0, 1]$. As for disjunction, we obtain $x \vee y \leq (x + y) \wedge 1$ for all $x, y \in [0, 1]$. Thus, the choice of maximum results in smaller thresholds, so there appears to be a trade-off between conjunction and disjunction in the improvement of lower bounds of probabilities. Again, one needs a clean way of simultaneously expressing lower and upper bound information to avoid this problem.

Corollary 1 *The soundness of our interpretation of CTL$^+$ in PCTL remains* *valid if we prematurely terminate fixed-point iterations in our model-checking algorithms.*

The last insight, which is true since 'being a lower bound' is a transitive property, is most important from a practical point of view. One may wonder why our fuzzy interpretation of $\wedge$ and $\vee$ worked. This is most likely due to that fact that path formulas in CTL are not allowed to contain logical connectives nor those of certain temporal operators. Thus, our approach will be challenged if one desires to extend it to CTL* or all of $\mu\mathtt{M}_{\leq 1}$. Nonetheless, the significance of Theorem 1 is that it provides a semantic bridge between two opposing views of quantitative model checking for labeled Markov chains as explained below.

First, our semantics endorses computations of quantitative evidence employing only one conceptual level, namely, *compositional* meaning in $S \to [0,1]$; this leads to a direct computational engine which need not be concerned with guessing informative thresholds for big systems and which computes evidence in an incremental way. The price being paid is the fuzzy interpretation of disjunction and conjunction by not addressing potential issues of conditional probabilities, so we might not compute precise probabilities although the considered property might induce a measurable set. However, Theorem 1 above can be seen as proving soundness of our semantics with respect to the one for PCTL. In particular, one can take the numbers computed for a CTL formula ϕ as a first safe guess of what will turn out to be 'true' probabilistically. Thus, one might want to check $\phi^\flat$ with appropriately higher thresholds than those computed for ϕ.

Second, the semantics for PCTL necessitates two conceptual levels: the usual qualitative semantics for formulas of CTL (including the usual interpretation of path formulas $\mathbf{X}\,\phi$ and $\phi_1 \mathbf{U}\, \phi_2$), and the computation of a probability for path formulas in a given state s. These distinct levels of analysis are linked via the annotation of thresholds to path formulas. The advantage lies in precise probabilistic results but it is not clear whether the framework of probability spaces can be applied in this manner for arbitrary $\phi \in \mu\mathtt{M}_{\leq 1}$ or for formulas with even higher alternation depths; this approach is also not continuous with respect to small changes in $\mathbf{P}$, the matrix of transition probabilities, for small changes of $\mathbf{P}(s,s')$ can change the Boolean truth value of a formula $\psi_{\sqsupset \mathbf{p}}$ at state s. As such it is quite sensitive to numerical errors. Incidentally, one could improve the symbolic model checking algorithm for PCTL in [1] by allowing *vectors of thresholds* and by overwriting thresholds on-the-fly that have been specified by the user in certain cases. For example, if the user would like to check $s \models [\phi_1 \mathbf{U} \phi_2]_{\geq \mathbf{p}}$ and if the probability of $\phi_1 \mathbf{U} \phi_2$ at s is greater than $\mathbf{p}_s$, then the check is not only successful, but it is also verifying the *stronger* check where we replace the threshold $\mathbf{p}_s$ by that very probability.

It should be noted that our fuzzy semantics of conjunction and disjunction *penalizes* the repeated check of the same property at the same syntactic level within a given formula, for such interpretations are not idempotent: we have $\min(0.5 + 0.5 - 1, 0) = 0$.

6 INFINITE LABELED MARKOV CHAINS

In this section we consider labeled Markov chains $(S, \mathbf{P}, L)$ such that S is countable where we demand that the infinite sums $\sum_{s' \in S} \mathbf{P}(s, s')$ have limit 1 for all $s \in S$ (this generalizes to uncountable state sets by means of integration). Countably infinite state sets suffice to show that fixed-point iterations for Kozen's standard interpretation of the modal mu-calculus may have to go beyond ω in order to reach their semantic invariants; see e.g. [4, pg. 20] for such an example within $\mu\mathtt{M}_{\leq 1}$. Yet, there is a significant difference between qualitative and quantitative fixed-points which guarantees that fixed-points in our semantics are always attained at the ordinal level ω for any fixed-point formula in $\mu\mathtt{M}_{\leq 1}$ even if the underlying state set is countably infinite: computing the meaning of $\mu x.\phi$ we can ask the questions '$1 = \bigvee_{0 \leq m \leq \omega} ({\mathbf{F}_{[\phi,x]}}^m\, \mathbf{0})\, s$?' and '$s \in \bigcup_{0 \leq m \leq \omega} {\mathbf{G}_{[\phi,x]}}^m\, \emptyset$?', where $\mathbf{G}_{[\phi,x]}$ is the function whose least fixed-point determines the meaning of $\mu x.\phi$ in Kozen's standard interpretation. The crucial difference is that s is an element of $\bigcup_{0 \leq m \leq \omega} {\mathbf{G}_{[\phi,x]}}^m\, \emptyset$ only if it is already a member in one of the finite unwindings ${\mathbf{G}_{[\phi,x]}}^m\, \emptyset$, whereas 1's being equal to $\bigvee_{0 \leq m \leq \omega} ({\mathbf{F}_{[\phi,x]}}^m\, \mathbf{0})\, s$ does not necessitate any of the numbers $({\mathbf{F}_{[\phi,x]}}^m\, \mathbf{0})\, s$ to be 1; the powerset lattice $\mathcal{P}(S)$ is algebraic whereas the lattice $S \to [0, 1]$ is 'only' continuous.

Recall that we proved that fixed-points of our semantics are reached at ω for all formulas of $\mu\mathtt{M}_{\leq 1}$ provided that S is finite. The arguments put forward in that proof essentially transfer to the case where S is countable. We simply endow $\mathbf{R}^\infty$ with the topology of pointwise convergence and note that this topology satisfies the following: the componentwise addition or subtraction of vectors in $\mathbf{R}^\infty$ is continuous, the componentwise operations $\mathbf{q} \mapsto 0 \vee \mathbf{q}\colon \mathbf{R}^\infty \to \mathbf{R}^\infty$ and $\mathbf{q} \mapsto 1 \wedge \mathbf{q}\colon \mathbf{R}^\infty \to \mathbf{R}^\infty$ are continuous, constant vectors give rise to continuous functions $\mathbf{R}^\infty \to \mathbf{R}^\infty$, and directed suprema and filtered infima are topological limits. The only fact we still require is that $(x_s)_{s \in S} \mapsto (\sum_{s' \in S} \mathbf{P}(s, s') \cdot x_{s'})_{s \in S}\colon \mathbf{R}^\infty \to \mathbf{R}^\infty$ be continuous, but this follows from basis facts on absolute convergence of real sums and the fact that $\mathbf{P}$ is a bounded linear operator on a Banach space.

Corollary 2 *Proposition 1 and Theorem 1 generalize to countable state sets.*

7 SUMMARY AND OUTLOOK

This paper is part of an effort to develop the *quantitative* model checking paradigm proposed in [14], where one requires quantities as answers to checks, such as performance or reliability measures, instead of simply 'yes' or 'no' answers. We have derived a symbolic model checking algorithm for the modal mu-calculus (without nested fixed points) re-interpreted in terms of expectations, also considered in [20]. Such algorithms form the basis of software tools supporting automatic and semi-automatic checks of programs containing quantitative information in the form of probability and time. The methods proposed here should avoid the state explosion problem in various important application domains by relying on a cogent representation of Markov transition probability matrices as MTBDDs introduced in [9, 1, 6]. The fixed-points can be reached in ω steps by successive approximation, which in practice means that the desired expectations can be computed to within a given $\varepsilon > 0$ in finitely many steps.

The expectations computed by the algorithm are not exact probabilities but lower bounds only; while exact probabilities might be preferable, the corresponding algorithms for satisfiability checking are complex (i.e. exponential in the size of the formula and polynomial in the size of the system for the case of Markov chains [10]), and so in such cases our algorithm provides a more efficient albeit coarser alternative. We can state that, in the case of positive CTL, the expectations semantics is probabilistically sound, in the sense that if the expectation p computed for a positive CTL formula is positive then the corresponding PCTL is also satisfied w.r.t. the threshold p. It would be worthwhile to extend this result to the full CTL and more expressive logics, such as CTL*, and models, for example concurrent Markov chains [25] or probabilistic systems which also exhibit non-determinism [3, 2], but it is not clear how this can be done without suitably extending the framework. Moreover, we would like to know whether Theorem 1 would go through for a framework *with* non-determinism, as in [22]. Most importantly, we need to find out whether there are any sensible ways of handling νx and $\neg$ within a semantics that approximates actual probabilities. Finally, we should point out that our semantic framework cuts across the usual classification of branching time versus linear time temporal logics. Our semantics of CTL has the feel of linear time temporal logic as the two quantifiers collapse into one notion, but it also contains branching aspects in the meaning of the modalities.

8 ACKNOWLEDGMENTS

We would like to thank all the anonymous referees for their constructive criticism.

REFERENCES

[1] Christel Baier, Ed Clarke, Vasiliki Hartonas-Garmhausen, Marta Kwiatkowska, and Mark Ryan. Symbolic model checking for probabilistic processes. In

Proceedings, 24th ICALP, volume 1256 of *Lecture Notes in Computer Science*, pages 430–440. Springer-Verlag, 1997.

[2] Christel Baier and Marta Kwiatkowska. Model checking for a probabilistic branching time logic with fairness. Accepted for *Distributed Computing* subject to revisions. Also available as Technical Report CSR-96-12.

[3] A. Bianco and L. de Alfaro. Model checking of probabilistic and nondeterministic systems. In *Proceedings, FST&TCS*, volume 1026 of *Lecture Notes in Computer Science*, pages 499–513. Springer-Verlag, 1995.

[4] J. C. Bradfield. *Verifying Temporal Properties Of Systems*. Birkhaeuser, Boston, Mass., 1991.

[5] J. R. Burch, E. M. Clarke, D. L. Dill K. L. McMillan, and J. Hwang. Symbolic model checking: 10^{20} states and beyond. *Information and Computation*, 98(2):142–170, 1992.

[6] S.V. Campos, E.M. Clarke, W.Marrero, and M. Minea. Verus: a tool for quantitative analysis of finite-state real-time systems. In *Proc. Workshop on Languages, Compilers and Tools for Real-Time Systems*, 1995.

[7] E. M. Clarke and E. M. Emerson. Synthesis of synchronization skeletons for branching time temporal logic. In D. Kozen, editor, *Proc. Logic of Programs*, number 131 in LNCS. Springer Verlag, 1981.

[8] E.M. Clarke, M. Fujita, P.C. McGeer, J.Yang, and X. Zhao. Multi-Terminal Binary Decision Diagrams: An Efficient Data Structure for Matrix Representation. In *IWLS'93: International Workshop on Logic Synthesis, Tahoe City*, May 1993.

[9] E.M. Clarke, K. L. McMillan, X. Zhao, M. Fujita, and J.Yang. Spectral Transforms for Large Boolean Functions with Applications to Technology Mapping. In *30th ACM/IEEE Design Automation Conference*, pages 54–60. ACM/IEEE, ACM, 1993.

[10] C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.

[11] E. A. Emerson and C.-L. Lei. Efficient Model Checking in Fragments of the Propositional Mu-Calculus. Annual IEEE Symposium on Logic in Computer Science, pages 267–279. IEEE Computer Society Press, 1986.

[12] H. Hansson. *Time and Probability in Formal Design of Distributed Systems*. Elsevier, 1994.

[13] H. Hansson and B. Jonsson. A logic for reasoning about time and probability. *Formal Aspects of Computing*, 6:512–535, 1994.

[14] M. Huth and M. Kwiatkowska. Quantitative Analysis and Model Checking. Annual IEEE Symposium on Logic in Computer Science, pages 111–122. IEEE Computer Society Press, June 1997.

[15] J.G. Kemeny and J.L. Snell. *Finite Markov Chains*. Van Nostrand, 1960.

[16] D. Kozen. A probabilistic PDL. In *15th Annual ACM Symposium on Theory of Computing*, pages 291–297, 1983.

[17] D. Kozen. Results on the propositional mu-calculus. *Theoretical Computer Science*, 27:333–354, 1983.

[18] D. Kozen. A Probabilistic PDL. *JCSS*, 30(2):162–178, 1985.

[19] K. G. Larsen and A. Skou. Bisimulation through Probabilistic Testing. *Information and Computation*, 94(1):1–28, September 1991.

[20] Annabelle McIver and Carroll Morgan. A probabilistic temporal calculus

based on expectations. In *Proceedings, Formal Methods Pacific'97*. Springer-Verlag, 1997. Also available as Technical Report PRG-TR-13-97.

[21] K. L. McMillan. *Symbolic Model Checking*. Kluwer Academic Publishers, 1993.

[22] C. Morgan and A. McIver. An expectation-transformer model for probabilistic temporal logic. Technical report, Computing Laboratory, Oxford University, April 1997. Draft. Personal Communication.

[23] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. In *Proceedings, CONCUR*, volume 836 of *Lecture Notes in Computer Science*, pages 481–496. Springer-Verlag, 1994.

[24] K. Seidel, C. Morgan, and A. McIver. An Introduction to Probabilistic Predicate Transformers. Technical Report PRG-TR-6-96, Oxford University Computing Laboratory, 1996.

[25] M. Vardi. Automatic Verification of Probabilistic Concurrent Finite-State Programs. In *Proc. FOCS'85*, pages 327–338. IEEE, 1985.

[26] P. Wolper. Temporal Logic Can Be More Expressive. Proc. of the 22nd Annual ACM Symposium on Foundations of Computer Science, pages 340–348, 1981.

[27] L. A. Zadeh. Fuzzy Sets. *Information and Control*, 8:338–353, 1965.

BIOGRAPHY

Michael Huth studied mathematics and computer science at the TU Darmstadt in Germany. In 1991 he received his Ph.D. from Tulane University of Louisiana. From 1991 to 1996 he took on post-doctoral fellowships at Kansas State University, Manhattan; Imperial College, London; TU Darmstadt, Darmstadt; and at the University of Birmingham in the United Kingdom. He is currently an assistant professor and a member of the programming languages group at Kansas State University. His interests range from the mathematical foundations of semantics, computational logic, language design and analysis, formal verification of concurrent systems, to the social impact of modern information technologies.

Marta Kwiatkowska graduated in computer science from the Jagiellonian University, Kraków, Poland. She moved to England in 1984, where she took up a lectureship at the University of Leicester and obtained her Ph.D. She held visiting positions at CWI, Amsterdam, and Imperial College, London, and has a wide network of international collaborators. Since 1994 she has been a member of the Theory of Computation group at the University of Birmingham. Her interests are in semantics for concurrency, infinite behaviour and fairness, and more recently probabilistic modelling and verification.