

# A Simple Probabilistic Extension of Modal Mu-calculus

**Wanwei Liu**

School of Computer Science,  
National University of Defense Technology,  
Changsha, P. R. China

**Ji Wang**

State Key Lab. of HPC,  
National University of Defense Technology,  
Changsha, P. R. China

**Lei Song**

University of Technology,  
Sydney, Australia

**Lijun Zhang**

State Key Lab. of Computer Science,  
Institute of Software, CAS,  
Beijing, P. R. China

## Abstract

Probabilistic systems are an important theme in AI domain. As the specification language, PCTL is the most frequently used logic for reasoning about probabilistic properties. In this paper, we present a natural and succinct probabilistic extension of  $\mu$ -calculus, another prominent logic in the concurrency theory. We study the relationship with PCTL. Surprisingly, the expressiveness is highly orthogonal with PCTL. The proposed logic captures some useful properties which cannot be expressed in PCTL. We investigate the model checking and satisfiability problem, and show that the model checking problem is in  $\mathbf{UP} \cap \mathbf{co-UP}$ , and the satisfiability checking can be decided via reducing into solving parity games. This is in contrast to PCTL as well, whose satisfiability checking is still an open problem.

## 1 Introduction

Temporal logics are heavily used in theoretical computer science and AI-related fields. Among those, modal  $\mu$ -calculus receives a lot of attraction ever since Kozen's seminal work [Kozen, 1983]. See for example, [Banieqbal and Barringer, 1987; Katoen, 1998; Walukiewicz, 2000; Berezin, 2002]. Moreover, various temporal logics including LTL [Pnueli, 1977], CTL [Emerson and Clarke, 1980], CTL\* [Emerson and Halpern, 1986] are extensively studied. It is known that their expressiveness is strictly less [Dam, 1995] than  $\mu$ -calculus (aka.  $\mu$ TL), and their model checking algorithm has been proposed: for CTL the problem can be solved in polynomial time, whereas for LTL the problem is  $\mathbf{PSPACE}$ -complete [Sistla and Clarke, 1985].

Probabilistic systems, such as Markov chains and Markov decision processes, are an important theme in AI domain. To reason about properties for probabilistic systems, the logic CTL was first extended with probabilistic quantifiers in [Hansson and Jonsson, 1994], resulting in the logic PCTL. Intuitively,  $(a\mathbf{U}^{\geq 0.9}b)$  means that the probability of reaching

$b$ -states along  $a$ -states is at least 0.9. At the same time, probabilistic LTL and its extension PCTL\* have all been studied. As in the classical setting, model checking problem for PCTL can be solved in polynomial time, whereas only exponential algorithms are known for LTL [Couvreur *et al.*, 2003]. There have also been several attempts to extend  $\mu$ TL with probabilities in the literature. As we shall discuss in the related work, the extensions are either highly non-trivial in terms of the complexity of the corresponding model checking and satisfiability problems, or hindered from the restriction of fix-point nesting.

We propose a natural and succinct extension of  $\mu$ TL in this paper, and name it  $\mathbf{P}\mu$ TL. The logic is acquired by equipping the next operator with probability quantifiers, and keeping other parts as standard  $\mu$ TL. We have for instance the formula  $\nu Z.(a \wedge X^{\geq 0.8}Z)$ . We investigate the model checking, expressiveness, and satisfiability problems of  $\mathbf{P}\mu$ TL.

In detail, we first investigate the model checking problem of  $\mathbf{P}\mu$ TL upon Markov chains. It turns out to be a straightforward adaptation of the classical algorithms for  $\mu$ TL, and the complexity remains in  $\mathbf{UP} \cap \mathbf{co-UP}$ . We then give a comprehensive study on the expressiveness of  $\mathbf{P}\mu$ TL by comparing with PCTL, and prove that  $\mathbf{P}\mu$ TL is orthogonal with PCTL in expressiveness. However, for the qualitative fragments (i.e., probabilities may appear in a formula are only 0 and 1), we show that qualitative  $\mathbf{P}\mu$ TL is strictly more expressive (w.r.t. finite Markov chains). On the other side, the satisfiability checking is quite challenging: we exploit the notion of probabilistic alternating parity automata (PAPA, for short), and reduce the SATISFIABILITY problem into the EMPTINESS problem of PAPA. Further, this is reduced to solving parity games, and it is shown that both of these two problems are in  $\mathbf{2EXPTIME}$ . This is in contrast to PCTL as well, whose SATISFIABILITY checking is still an open problem (cf. [Brázdil *et al.*, ; Bertrand *et al.*, 2012]).

**An illustrating example** We introduce a running example to motivate our work: Suppose there is a hacker trying to attack a remote server. The hacker has a supercomputer at hand and is trying to guess the password in a brute-force manner. For simplicity, we assume the password is a sequence of  $l$  let-

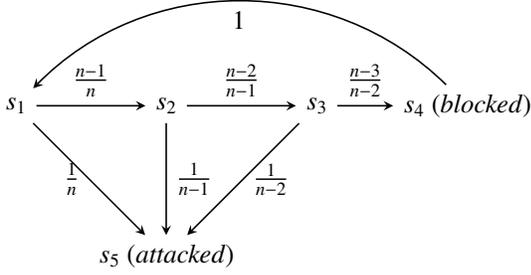


Figure 1: An illustration of the hacking process

ters, each of which is from ‘0’-‘9’, ‘a’-‘z’, and ‘A’-‘Z’. Therefore, the total number of possible passwords is  $n = 62^l$ . The hacker let the supercomputer randomly generate a password, and see whether the decryption succeeds. If yes, the hacker wins; otherwise he tries with another one. However, if the supercomputer generates three wrong passwords in a row, it will be blocked for a certain amount of time until it can start another round of attacking — assuming that the password may be changed during the blocked moment, hence it does not make sense for the supercomputer to store all generated passwords. The whole process is illustrated in Fig. 1. Starting from  $s_1$ , we can see that the probability of eventually reaching *attacked*, i.e., the hacker decrypts successfully, equal 1, no matter how big  $l$  is (hence, the PCTL formula  $F^{\geq 1} \textit{attacked}$  holds), and we may conclude that the system is unsafe — this is of course against our intuition, as such system is considered to be safe if  $l$  is big enough. However, as we will show later, all PCTL formulae are not capable of expressing this property. By making use of  $P\mu\text{TL}$ , such property of security can be characterized easily as follows:  $\nu Z.(\neg \textit{attacked} \wedge X^{\geq p} Z)$  with  $p = n^{-3}/n^{-2}$ , where  $\neg \textit{attacked}$  denotes all other states in Fig. 1 different from  $s_5$ .

**Motivation from AI perspective** The presented logic has the following potential application in AI domain:

- First of all, Markov chains and Markov decision processes are the basic models in several areas of AI. As a logic with semantics defined w.r.t. such models, it could definitely be used in designating probability-relevant properties upon them. Particularly, the properties that could not be expressed by PCTL.
- Motion planning is an important topic in AI area, where standard  $\mu\text{TL}$  has once been adopted [Bhatia *et al.*, 2011], because of its powerful expressiveness and the decidability of its SATISFIABILITY problem. Thus, we expect that  $P\mu\text{TL}$  could be used in stochastic motion planning — since,  $P\mu\text{TL}$  is a decidability-preserving extension of  $\mu\text{TL}$ .
- Fixpoints play an important role in mathematics and computer science. In AI area, it is used to designate non-terminating behaviors of intelligent systems, such as maintenance goals [Singh, 1998]. Fixpoints act as the elementary ingredients in  $P\mu\text{TL}$ , hence such logic can also be used in such a situation.

**Related work** Probabilistic extensions of  $\mu\text{TL}$  have been studied by many authors: e.g.,  $\mu$ -calculi proposed in [Morgan and McIver, 1997; Huth and Kwiatkowska, 1997; de Alfaro and Majumdar, 2001; McIver and Morgan, 2002; 2007; Mio, 2012b] interpret a formula as a function from states to real values in  $[0, 1]$ , whose semantics is different from  $P\mu\text{TL}$ . A further extension of  $\mu$ -calculus was proposed in [Mio, 2012a], which is able to encode the full PCTL. However, the model checking and SATISFIABILITY algorithms are still unknown for these calculi and are “far from trivial” [Mio, 2012b]. The other probabilistic  $\mu$ -calculus was introduced in [Cleaveland *et al.*, 2005] along with a model checking algorithm for it. Moreover, it is able to encode PCTL formulae as well. However, that calculus only allows alternation-free formulae (cf. [Emerson and Lei, 1986]).

Very recently — and independently —, Castro, Kilmurray, and Piterman present another extension by adding fixpoints to full PCTL [Castro *et al.*, 2015]. The calculus they introduced is more expressive than logics PCTL and PCTL\*. Moreover, it is also easy to see that it is a proper super logic of our logic  $P\mu\text{TL}$  as well. They show the model checking problem is in  $\text{NP} \cap \text{co-NP}$ . We note that some examples in our paper are similarly investigated in [Castro *et al.*, 2015]. Since the logic in [Castro *et al.*, 2015] subsumes PCTL, its SATISFIABILITY problem is also left open. However in this paper we show SATISFIABILITY of  $P\mu\text{TL}$  could be reduced to solving parity games, which makes this problem solvable in  $2\text{EXPTIME}$ .

## 2 Preliminaries

In this paper, we fix a countable set  $\mathcal{A}$  of *atomic propositions*, ranging over  $a, b, a_1$  etc, and fix a countable set  $\mathcal{Z}$  of *formula variables*, ranging over  $Z, Z_1$  etc.

A *Markov chain* is a tuple  $M = (S, T, L)$ , where  $S$  is a finite set of *states*;  $T : S \times S \rightarrow [0, 1]$  is the matrix of transition-probabilities, fulfilling  $\sum_{s' \in S} T(s, s') = 1$  for every  $s \in S$ ; and  $L : S \rightarrow 2^{\mathcal{A}}$  is the labeling function. A *pointed Markov chain* is a pair  $(M, s)$  where  $M$  is a Markov chain  $(S, T, L)$  and  $s \in S$  is the *initial state*.

An (infinite) *path*  $\pi$  of  $M$  is an infinite sequence of states  $s_0, s_1, \dots$ , such that  $s_i \in S$  and  $T(s_i, s_{i+1}) > 0$  for each  $i$ . A basic *cylinder*  $\text{cyl}(s_0, s_1, \dots, s_n)$  of  $M$  is the set of infinite paths having  $s_0, s_1, \dots, s_n$  as the prefix.

According to the standard theory of Markov process, the pointed Markov chain  $(M, s)$  uniquely derives a *measure space*  $(\Pi_{M,s}, \Delta_{M,s}, \text{prob}_{M,s})$  where  $\Pi_{M,s}$  consists of all infinite paths of  $M$ ;  $\Delta_{M,s}$  is the minimal Borel field containing all basic cylinder of  $M$  (i.e.,  $\Delta_{(M,s)}$  is closed under complementation and countable intersection); and the measuring function  $\text{prob}_{M,s}$  fulfills:  $\text{prob}_{M,s}(\text{cyl}(s_0, s_1, \dots, s_n))$  equals 0 if  $s \neq s_0$ , and equals  $\prod_{i < n} T(s_i, s_{i+1})$  otherwise. We say a set  $P \subseteq \Pi_{M,s}$  is *measurable* if  $P \in \Delta_{M,s}$ . [Vardi, 1985] shows that the intersection of  $\Pi_{M,s}$  and an omega-regular set must be measurable.

The syntax of PCTL formulae is described by the following abstract grammar:

$$f ::= \top \mid \perp \mid a \mid \neg a \mid X^{\sim p} f \mid f \wedge f \mid f \vee f \mid f U^{\sim p} f \mid f R^{\sim p} f$$

where  $\sim \in \{>, \geq\}$  and  $p \in [0, 1]$ . We also abbreviate  $\top U^{\sim p} f$  and  $\perp R^{\sim p} f$  as  $F^{\sim p} f$  and  $G^{\sim p} f$ , respectively.

Semantics of a PCTL formula is given w.r.t. a Markov chain. For each PCTL formula  $f$  and a Markov chain  $M = (S, \mathbf{T}, L)$ , we will use  $\llbracket f \rrbracket_M$  to denote the subset of  $S$  satisfying  $f$ , inductively defined as follows.

- $\llbracket \top \rrbracket_M = S$ ;  $\llbracket \perp \rrbracket_M = \emptyset$ .
- $\llbracket a \rrbracket_M = \{s \in S \mid a \in L(s)\}$ ;  $\llbracket \neg a \rrbracket_M = \{s \in S \mid a \notin L(s)\}$ .
- $\llbracket X^{\sim p} f \rrbracket_M = \{s \in S \mid \sum_{s' \in \llbracket f \rrbracket_M} \mathbf{T}(s, s') \sim p\}$ .
- $\llbracket f_1 \wedge f_2 \rrbracket_M = \llbracket f_1 \rrbracket_M \cap \llbracket f_2 \rrbracket_M$ ;  $\llbracket f_1 \vee f_2 \rrbracket_M = \llbracket f_1 \rrbracket_M \cup \llbracket f_2 \rrbracket_M$ .
- $\llbracket f_1 U^{\sim p} f_2 \rrbracket_M = \{s \in S \mid \text{prob}_{M,s}\{\pi \in \text{cyl}(s) \mid \pi \models f_1 U f_2\} \sim p\}$  and  $\llbracket f_1 R^{\sim p} f_2 \rrbracket_M = \{s \in S \mid \text{prob}_{M,s}\{\pi \in \text{cyl}(s) \mid \pi \models f_1 R f_2\} \sim p\}$ .

In addition, for an infinite path  $\pi = s_0, s_1, \dots$  of  $M$ , the notation  $\pi \models f_1 U f_2$  stands for that there is some  $i \geq 0$  such that  $s_i \in \llbracket f_2 \rrbracket_M$  and  $s_j \in \llbracket f_1 \rrbracket_M$  for each  $j < i$ . Meanwhile,  $\pi \models f_1 R f_2$  holds if either  $\pi \models f_2 U (f_1 \wedge f_2)$  or  $s_j \in \llbracket f_2 \rrbracket_M$  for each  $j$ . To simplify notations, in what follows we denote by  $M, s \models f$  whenever  $s \in \llbracket f \rrbracket_M$  holds.

### 3 P $\mu$ TL, Syntax and Semantics

In this section we present a simple probabilistic extension of modal  $\mu$ -calculus, called P $\mu$ TL. The syntax of P $\mu$ TL formulae is depicted as follows:

$$f ::= \top \mid \perp \mid a \mid \neg a \mid Z \mid X^{\sim p} f \mid f \wedge f \mid f \vee f \mid \mu Z.f \mid \nu Z.f$$

Semantics of a P $\mu$ TL formula is given w.r.t. a Markov chain  $M = (S, \mathbf{T}, L)$  and an assignment  $e : \mathcal{Z} \rightarrow 2^S$ . Similarly, for each P $\mu$ TL formula  $f$ , we denote by  $\llbracket f \rrbracket_M(e)$  the state set satisfying  $f$  under  $e$ . Inductively:

- $\llbracket \top \rrbracket_M(e) = S$  and  $\llbracket \perp \rrbracket_M(e) = \emptyset$ .
- $\llbracket a \rrbracket_M(e) = \{s \in S \mid a \in L(s)\}$  and  $\llbracket \neg a \rrbracket_M(e) = \{s \in S \mid a \notin L(s)\}$ .
- $\llbracket Z \rrbracket_M(e) = e(Z)$ .
- $\llbracket X^{\sim p} f \rrbracket_M(e) = \{s \in S \mid \sum_{s' \in \llbracket f \rrbracket_M(e)} \mathbf{T}(s, s') \sim p\}$ .
- $\llbracket f_1 \wedge f_2 \rrbracket_M(e) = \llbracket f_1 \rrbracket_M(e) \cap \llbracket f_2 \rrbracket_M(e)$   
and  $\llbracket f_1 \vee f_2 \rrbracket_M(e) = \llbracket f_1 \rrbracket_M(e) \cup \llbracket f_2 \rrbracket_M(e)$ .
- $\llbracket \mu Z.f \rrbracket_M(e) = \bigcap \{S' \subseteq S \mid \llbracket f \rrbracket_M(e[Z \mapsto S']) \subseteq S'\}$  and  
 $\llbracket \nu Z.f \rrbracket_M(e) = \bigcup \{S' \subseteq S \mid \llbracket f \rrbracket_M(e[Z \mapsto S']) \supseteq S'\}$ .

Indeed,  $\llbracket \mu Z.f \rrbracket_M(e)$  (resp.  $\llbracket \nu Z.f \rrbracket_M(e)$ ) could be computed as in the classical setting via the following iteration:

1. let  $S_0 = \emptyset$  (resp.  $S_0 = S$ );
2. subsequently, let  $S_{i+1} = \llbracket f \rrbracket_M(e[Z \mapsto S_i])$ ;
3. stops if  $S_{\ell+1} = S_\ell$ , and returns  $S_\ell$ .

Note that the algorithm obtains a monotonic chain with such an iteration, and hence it must terminate within finite steps. Actually,  $\llbracket \mu Z.f \rrbracket_M(e)$  (resp.  $\llbracket \nu Z.f \rrbracket_M(e)$ ) captures the least (resp. greatest) solution of  $X = \llbracket f \rrbracket_M(e[Z \mapsto X])$  within  $2^S$ .

Semantical definition of P $\mu$ TL formulae also yields the model checking algorithm.

**Theorem 1.** *The model checking problem of P $\mu$ TL is in UP  $\cap$  co-UP.*

Indeed, the proof is analogous to the non-probabilistic version [Jurdziński, 1998; Wilke, 2002] and the only noteworthy difference lies from handling  $X^{\sim p}$ -subformulae, opposing to  $\square$ - and  $\diamond$ -subformulae, which could be proceeded in (deterministic) polynomial time.

In what follows, we directly denote by  $\llbracket f \rrbracket_M$  in the case that  $f$  is a closed formula (i.e., each variable of  $f$  is bound), and we also denote by  $M, s \models f$  if  $s \in \llbracket f \rrbracket_M$ .

Below we give some example properties:

- (1) The formula  $\nu Z.(a \wedge X^{>0.8} Z)$  describes that there exists an  $a$ -region, where each state has less than 0.2 probability to escape from it immediately (i.e., in one step).
- (2)  $\nu Z.(a \wedge X^{>0} X^{>0} Z)$  says that there is a cycle in the Markov chain, such that  $a$  holds at least in every even step.
- (3)  $M, s \models \mu Z.(a \vee X^{\geq 0.6} Z)$  if some  $a$ -state is reachable from  $s$ , but at each step, one just has some probability (not less than 0.6) to go on with the right direction.
- (4) The P $\mu$ TL formula  $\mu Z.(b \vee (a \wedge X^{\geq 1} Z))$  holds if  $aUb$  holds along each path. It is stronger than the property described by the PCTL formula  $aU^{\geq 1} b$ . For the latter allows the existence of  $a$ -cycles.
- (5) As a more complicated example, the formula  $\nu Z_1.(a \vee \mu Z_2.(a \vee X^{>0} Z_2) \wedge X^{\geq 1} Z_1)$  just tells the story that “ $a$  will be surely encountered”, as described by  $F^{\geq 1} a$  with PCTL.

Given a P $\mu$ TL formula  $f$  and a bound variable  $Z$ , we use  $\mathcal{D}_f(Z)$  to denote the subformula which binds  $Z$  in  $f$ . For example, let  $f = \mu Z_1.(a \wedge \nu Z_2.(b \wedge X^{>0.3} Z_2) \vee X^{>0.6} Z_1)$ , then we have  $\mathcal{D}_f(Z_1) = f$  and  $\mathcal{D}_f(Z_2) = \nu Z_2.(b \wedge X^{>0.3} Z_2)$ .

We say that a P $\mu$ TL formula  $f$  is *guarded*, if the occurrence of each bound variable  $Z$  in  $\mathcal{D}_f(Z)$  is in the scope of some  $X$ -operator. The following theorem could be proven in a same manner as that in [Walukiewicz, 2000].

**Theorem 2.** *For each P $\mu$ TL formula  $f$ , there is a guarded formula  $f'$  such that  $\llbracket f' \rrbracket_M(e) = \llbracket f \rrbracket_M(e)$  for every  $M$  and  $e$ .*

Thus, in what follows, we always assume that each P $\mu$ TL formula is guarded.

### 4 Expressiveness

In this section, we will give a comparison between P $\mu$ TL and PCTL, and we are only concerned about closed P $\mu$ TL formulae. For a P $\mu$ TL formula  $f$  and a PCTL formula  $g$ , we say that  $f$  and  $g$  are *equivalent* if  $\llbracket f \rrbracket_M = \llbracket g \rrbracket_M$  for every Markov chain  $M$ , denoted as  $f \equiv g$ .

First of all, we will show that some P $\mu$ TL formula could not be equivalently expressed by any PCTL formula.

**Theorem 3.** *Let  $f = \nu Z.(a \wedge X^{\geq 0.5} Z)$ , then  $g \not\equiv f$  for every PCTL formula  $g$ .*

*Proof.* To show this, we need first construct two families of Markov chains, namely,  $M_0, M_1, \dots$ , and  $M'_0, M'_1, M'_2, \dots$ .

For the first group, let  $M_n = (\{s_0, s_1, \dots, s_n\}, \mathbf{T}_n, L_n)$ , where:  $\mathbf{T}_n(s_0, s_0) = 1$  and  $\mathbf{T}_n(s_{i+1}, s_i) = 1$  for each  $i < n$  (hence  $\mathbf{T}_n(s_i, s_j) = 0$  for any other  $s_i, s_j$ ). In addition,  $L_n(s_0) = \emptyset$  and  $L_n(s_i) = \{a\}$  for each  $0 < i \leq n$ .

For the second ones, let  $M'_n = (\{s'_0, s'_1, \dots, s'_n\}, \mathbf{T}'_n, L'_n)$  where:  $\mathbf{T}'_n(s'_n, s'_n) = \mathbf{T}'_n(s'_n, s'_{n-1}) = 0.5$ ,  $\mathbf{T}'_n(s'_0, s'_0) = 1$ , and

$T'_n(s'_{i+1}, s'_i) = 1$  for every  $i < n - 1$ . In addition,  $L'_n(s'_0) = \emptyset$  and  $L'_n(s'_i) = \{a\}$  for each  $0 < i \leq n$ .

Given a PCTL formula  $g$ , let  $N(g)$  be the maximal nesting depth of temporal-operators of  $g$ . According to [Baier and Katoen, 2008, Thm. 10.45], we have that  $M'_n, s'_n \models g$  if and only if  $M_n, s_n \models g$  whenever  $n \geq N(g)$ .

Observe the fact that  $M'_n, s'_n \models f$  and  $M_n, s_n \not\models f$  for every  $n \geq 1$ . Assume that there exists some PCTL formula  $g$  fulfilling  $f \equiv g$ , then we have

$$\begin{aligned} M'_{N(g)}, s'_{N(g)} \models f &\iff M'_{N(g)}, s'_{N(g)} \models g \\ \iff M_{N(g)}, s_{N(g)} \models g &\iff M_{N(g)}, s_{N(g)} \models f \end{aligned}$$

and hence it results in a contradiction.  $\square$

Conversely, the following theorem reveals that there also exists some PCTL formula that could not be equivalently expressed by any  $P\mu$ TL formula.

**Theorem 4.** *Let  $f = F^{\geq 0.5}a$ , then  $g \not\equiv f$  for every (closed)  $P\mu$ TL formula  $g$ .*

*Proof.* Let  $M = (\{s_1, s_2, s_3\}, T, L)$  be the (family of) Markov chain(s) where:  $L(s_1) = L(s_2) = \emptyset$ ,  $L(s_3) = \{a\}$ ,  $T(s_1, s_1) = x$ ,  $T(s_1, s_2) = y$ ,  $T(s_1, s_3) = z$ , and  $T(s_2, s_2) = T(s_3, s_3) = 1$ , with  $x, y, z \in (0, 1)$  and  $x + y + z = 1$ .

For every PCTL and/or closed  $P\mu$ TL formula  $g$ , we let  $P_x(g)$  be the proposition that “for the fixed  $x$ , there are infinitely many  $y$  making  $M, s_1 \models g$  and there are infinitely many  $y$  making  $M, s_1 \not\models g$ ”. We now show that if  $g$  is a closed  $P\mu$ TL formula, then there exists some  $x_g < 1$  such that  $P_x(g)$  does not hold whenever  $x \in (x_g, 1)$ .

- Such  $x_g$  can be arbitrarily chosen if  $g = \perp$ ,  $g = \top$ ,  $g = a$  or  $g = \neg a$ .
- In the case that  $g = g_1 \wedge g_2$ , assume by contradiction that such  $x_g$  does not exist, then it implies that for every  $x \in (0, 1)$ , there exists some  $x' > x$  such that  $P_{x'}(g)$  holds. Observe that  $M, s_1 \models g$  implies both  $M, s_1 \models g_1$  and  $M, s_1 \models g_2$ ; and  $M, s_1 \not\models g$  implies either  $M, s_1 \not\models g_1$  or  $M, s_1 \not\models g_2$ . Thus, we can infer that either  $x_{g_1}$  or  $x_{g_2}$  does not exist, which violates the induction hypothesis.
- Proof for the case of  $g = g_1 \vee g_2$  is similar to the above.
- If  $g = X^{\sim p}g'$  and  $p \in (0, 1)$ , whenever  $x \in (\max\{p, 1 - p\}, 1)$ , since  $\sim \in \{>, \geq\}$ , then  $M, s_1 \models g$  iff  $M, s_1 \models g'$  because  $y + z < p$  in such situation. In this case, we may just let  $x_g = \max\{x_{g'}, p, 1 - p\}$ .
- If  $g = X^{\geq 1}g'$ , then we need to distinguish two cases: 1) There exist  $x, y \in (0, 1)$  such that  $M, s_1 \models g$  holds, then we can immediately infer that both  $M, s_2 \models g'$  and  $M, s_3 \models g'$ . In addition, observe that truth values of  $g'$  on  $s_2$  and  $s_3$  are irrelevant to  $x$  and  $y$ . It implies that in such case  $M, s_1 \models g$  iff  $M, s_1 \models g'$ , and hence, we may just let  $x_g = x_{g'}$ . 2) There is no such  $x$  and  $y$  having  $M, s_1 \models g$  holds, in such situation,  $x_g$  can be any number in  $(0, 1)$ .
- If  $g = X^{>0}g'$ , then the proof is similar to the above.
- When  $g = X^{\geq 0}g'$  (or  $g = X^{>1}g'$ ), things would be trivial, because  $g$  could be reduced to  $\top$  (resp.  $\perp$ ) in such case.

- If  $g = \mu Z.g'$ , we let  $g_0 = \perp$  and  $g_{i+1} = g'[Z/g_i]$ . Since that  $M$  is a 3-state Markov chain, then  $g$  and  $\bigvee_{i \leq 3} g_i$  share the same truth value at every state of  $M$ . This indicates that all least fix-points could be eliminated w.r.t. such Markov chain.

- When  $g = \nu Z.g'$ , the preprocessing is almost similar, but we just replace  $g$  with  $\bigwedge_{i \leq 3} g_i$  where  $g_0 = \top$ .

Now, for the PCTL formula  $f = F^{\geq 0.5}a$ , such  $x_f$  does not exist, because, for every  $x \in (0, 1)$  we have:  $M, s_1 \models f$  provided that  $y \in [(1 - x)/2, 1)$ ; and  $M, s_1 \not\models f$  if  $y \in (0, (1 - x)/2)$ . This implies that  $P_x(f)$  holds for every  $x \in (0, 1)$ , and hence  $f$  cannot be equally expressed by any  $P\mu$ TL formula.  $\square$

Note that the value 0.5 in the previous two theorems can be generalized to any other probability  $p \in (0, 1)$ .

We also provide a comparison on the *qualitative* fragments of PCTL and  $P\mu$ TL. Probabilities occurring in such fragments can only be 0 or 1.

**Theorem 5.** *Every qualitative PCTL formula can be equally expressed by a qualitative  $P\mu$ TL formula.*

*Proof.* We will give a constructive translation procedure, which takes a qualitative PCTL formula  $g$  and outputs an equivalent qualitative  $P\mu$ TL formula  $\tilde{g}$ . Inductively:

1.  $\tilde{g} = \perp$  if  $g = \perp$ , or its root operator is  $X^{>1}$ ,  $U^{>1}$  or  $R^{>1}$ ;  $\tilde{g} = \top$  if  $g = \top$ , or its root operator is  $X^{\geq 0}$ ,  $U^{\geq 0}$  or  $R^{\geq 0}$ .
2.  $\tilde{g} = \tilde{g}_1 \wedge \tilde{g}_2$  if  $g = g_1 \wedge g_2$ ; and  $\tilde{g} = \tilde{g}_1 \vee \tilde{g}_2$  if  $g = g_1 \vee g_2$ .
3.  $\tilde{g} = X^{>0}\tilde{g}'$  if  $g = X^{>0}g'$ ; and  $\tilde{g} = X^{\geq 1}\tilde{g}'$  if  $g = X^{\geq 1}g'$ .
4.  $\tilde{g} = \mu Z.(\tilde{g}_2 \vee (\tilde{g}_1 \wedge X^{>0}Z))$  if  $g = g_1 U^{>0}g_2$ ; and  $\tilde{g} = \nu Z.(\tilde{g}_2 \wedge (\tilde{g}_1 \vee X^{\geq 1}Z))$  if  $g = g_1 R^{\geq 1}g_2$ .
5.  $\tilde{g} = \nu Z.(\tilde{g}_2 \vee (\tilde{g}_1 \wedge \widetilde{F^{>0}Z}) \wedge X^{\geq 1}Z)$  if  $g = g_1 U^{\geq 1}g_2$ ; and  $\tilde{g} = \mu Z.(\tilde{g}_2 \wedge (\tilde{g}_1 \vee \widetilde{G^{\geq 1}Z}) \vee X^{>0}Z)$  if  $g = g_1 R^{>0}g_2$ .

The proof of equivalence could be done by induction on the structure of the formula.  $\square$

Note that Thm. 5 holds because we are only concerned about finite models in this paper. Interested readers may show that it is not true for infinite Markov chains.

**Theorem 6.** *The qualitative  $P\mu$ TL formula  $f = \nu Z.(a \wedge X^{>0}X^{>0}Z)$  cannot be expressed in qualitative PCTL.*

*Proof.* Construct a series of Markov chains  $M''_2, M''_3, \dots$  such that each  $M''_n$  is the Markov chain  $(\{s''_0, s''_1, \dots, s''_n\}, T''_n, L''_n)$ , where  $T''_n(s''_0, s''_0) = 1$  and  $T''_n(s''_{i+1}, s''_i) = 1$  for each  $i < n$ . In addition,  $L''_n(s''_i) = \{a\}$  for each  $i \neq 1$ , and  $L''_n(s''_1) = \emptyset$ .

For a given PCTL formula  $g$ , let  $\hat{g}$  be the LTL formula obtained from  $g$  by discarding all probability quantifiers, e.g., we have  $\hat{g} = aU(b \vee G\neg a)$  if  $g = aU^{\geq 0.3}(b \vee G^{>0.6}\neg a)$ . Since that from  $s''_n$  the Markov chain  $M''_n$  has exactly one infinite path  $\pi_n = s''_n, \dots, s''_1, (s''_0)^\omega$ , then for each  $n \geq 2$  we have  $M''_n, s''_n \models g$  if and only if  $\pi_n \models \hat{g}$ . It is shown in [Wolper, 1983] that  $M''_n, s''_n \models g$  iff  $M''_{n+1}, s''_{n+1} \models g$  in the case of  $n \geq N'(\hat{g}) = N'(g)$ , where  $N'(g)$  and  $N'(\hat{g})$  are the nesting depth of  $X$ -operator of  $g$  and  $\hat{g}$ , respectively. Thus, we have

$M''_n, s''_n \models g$  iff  $M''_{n+1}, s''_{n+1} \models g$  in such situation. This implies that  $\nu Z.(a \wedge X^{>0} X^{>0} Z)$  has no equivalent qualitative PCTL expression, because we cannot simultaneously have  $M''_n, s''_n \models f$  and  $M''_{n+1}, s''_{n+1} \models f$  for each  $n \geq 2$ .  $\square$

Note that the conclusion of Thm. 6 is also pointed out in [Cleaveland *et al.*, 2005], and we here provide a detailed proof. Indeed, this proof also works for general PCTL formulae, and hence the property  $\nu Z.(a \wedge X^{>0} X^{>0} Z)$  even cannot be expressed by any PCTL formula.

## 5 Automata Characterization

In this section, we will define a new type of automata recognizing (pointed) Markov chains, called *probabilistic alternating parity automata* (PAPA, for short), and such automata could be viewed as the probabilistic extension of those defined in [Wilke, 2002].

A PAPA  $A$  is a tuple  $(Q, q_0, \delta, \Omega)$  where:  $Q$  is a finite set of states,  $q_0 \in Q$  is the initial state,  $\delta$  is the transition function to be defined later, and  $\Omega : Q \rightsquigarrow \mathbb{N}$ , is a partial function of coloring; in what follows, we say a state is *colored* if  $\Omega$  is defined for the state.

The notion of *transition conditions* over  $Q$  is inductively defined as follows:

1.  $\perp$  and  $\top$  are transition conditions over  $Q$ .
2. For every  $a \in \mathcal{A}$ , the literals  $a$  and  $\neg a$  are transition conditions over  $Q$ .
3. If  $q \in Q$ , then  $q$  is a transition condition over  $Q$ .
4. If  $q \in Q$  and  $p \in [0, 1]$ , then  $\circ^{\sim p} q$  is a transition condition over  $Q$ , where  $\sim \in \{\geq, >\}$ .
5. If  $q_1, q_2 \in Q$  then both  $q_1 \vee q_2$  and  $q_1 \wedge q_2$  are transition conditions over  $Q$ .

The transition function  $\delta$  assigns each state  $q \in Q$  a transition condition over  $Q$ .

We denote by  $R_A$  the *derived graph* of  $A$ , its vertex set is just  $Q$ , and there is an edge from  $q_1$  to  $q_2$  iff  $q_2$  appears in  $\delta(q_1)$ . We say that  $A$  is *well-structured*, if for every path  $q_1, q_2, \dots, q_n$  that forms a cycle (i.e.,  $q_1 = q_n$ ) in  $R_A$ , we have that: 1) there exists some  $1 \leq i < n$  such that  $\delta(q_i) = \circ^{\sim p} q_{i+1}$  with some  $p \in [0, 1]$ ; 2) there exists some  $1 \leq j < n$  such that  $q_j$  is colored. In what follows, we are only concerned about well-structured PAPA.

Given a pointed Markov chain  $(M, s_0)$  with  $M = (S, T, L)$  and  $s_0 \in S$ , a *run* of  $A$  over  $(M, s_0)$  is a  $Q \times S$ -labeled tree  $(T, \lambda)$  fulfilling:  $\lambda(v_0) = (q_0, s_0)$  for the root vertex  $v_0$ ; and for each internal vertex  $v$  of  $T$  with  $\lambda(v) = (q, s)$  we require that

- $\delta(q) \neq \perp$ , and if  $\delta(q) = \top$  then  $v$  has no child;
- $a \in L(s)$  if  $\delta(q) = a$ , and  $a \notin L(s)$  if  $\delta(q) = \neg a$ ;
- if  $\delta(q) = q_1 \wedge q_2$  then  $v$  has two children  $v_1$  and  $v_2$  respectively having  $\lambda(v_1) = (q_1, s)$  and  $\lambda(v_2) = (q_2, s)$ ;
- if  $\delta(q) = q_1 \vee q_2$  then  $v$  has one child  $v'$  with  $\lambda(v') \in \{(q_1, s), (q_2, s)\}$ ;
- $v$  has one child  $v'$  having  $\lambda(v') = (q', s)$ , if  $\delta(q) = q'$ ;
- if  $\delta(q) = \circ^{\sim p} q'$  then  $v$  has a set of children  $v_1, \dots, v_n$  such that  $\lambda(v_i) = (q', s_i)$ , where  $\sum_{i=1}^n T(s, s_i) \sim p$ .

For an infinite branch  $\tau = v_0, v_1, \dots$  of  $T$ , let  $n_\tau$  be the number

$$\max\{n \mid \text{there are infinitely many } i \text{ s.t. } \Omega(\text{proj}_1(\lambda(v_i))) = n\}$$

where  $\text{proj}_1(q, s) = q$ . A run  $(T, \lambda)$  is *accepting* if  $n_\tau$  is an even number, for every infinite branch  $\tau$  of  $T$ . A pointed Markov chain  $(M, s_0)$  is *accepted* by  $A$  if  $A$  has an accepting run over it. We denote by  $\mathcal{L}(A)$  the set consisting of pointed Markov chains accepted by  $A$ .

**Theorem 7.** *Given a closed P $\mu$ TL formula  $f$ , there is a PAPA  $A_f$  such that:  $M, s \models f$  iff  $(M, s) \in \mathcal{L}(A_f)$ , for each pointed Markov chain  $(M, s)$ .*

*Proof.* We just let  $A_f = (Q_f, q_f, \delta_f, \Omega_f)$ , where:

- $Q_f = \{q_g \mid g \text{ is a subformula of } f\}$ , and hence  $q_f \in Q_f$ ;
- $\delta_f$  is defined as follows:
  - $\delta_f(q_\perp) = \perp$  and  $\delta_f(q_\top) = \top$ ;
  - $\delta_f(q_a) = a$  and  $\delta_f(q_{\neg a}) = \neg a$ ;
  - $\delta_f(q_{g_1 \wedge g_2}) = q_{g_1} \wedge q_{g_2}$  and  $\delta_f(q_{g_1 \vee g_2}) = q_{g_1} \vee q_{g_2}$ ;
  - $\delta_f(q_{X^{\sim p} g}) = \circ^{\sim p} q_g$ ;
  - $\delta_f(q_{\mu Z.g}) = q_g$  and  $\delta_f(q_{\nu Z.g}) = q_g$ ;
  - $\delta_f(q_Z) = q_{\mathcal{D}_f(Z)}$ .
- $\Omega_f$  is defined at every state  $q_Z$  with  $Z \in \mathcal{Z}$  fulfilling: If  $Z$  is a  $\mu$ -variable (resp.  $\nu$ -variable), then  $\Omega_f(q_Z)$  is the minimal odd (resp. even) number which is greater than every  $\Omega_f(q_{Z'})$  such that  $\mathcal{D}_f(Z')$  is a subformula of  $\mathcal{D}_f(Z)$ .

It could be directly examined that  $A_f$  is well-structured since  $f$  is guarded. The proof of equivalence can be similarly done as that in [Wilke, 2002] — the only different induction step is to deal with transitions being of  $\circ^{\sim p} q$  (in that paper, the corresponding cases are  $\square q$  and  $\diamond q$ ). Actually, we can see that if a PAPA  $(Q, q, \delta, \Omega)$  corresponds to the P $\mu$ TL formula  $g$ , then the PAPA  $(Q \cup \{q'\}, q', \delta[q' \mapsto \circ^{\sim p} q], \Omega)$  must correspond to  $X^{\sim p} g$ .  $\square$

## 6 Satisfiability Decision

It is known from Section 5 that the SATISFIABILITY problem of P $\mu$ TL could be reduced to the EMPTYNESS problem of PAPA. In this section, we will further reduce it to parity game solving.

A *parity game*  $G$  is a tuple  $(V, E, C)$ , where:  $V$  is a finite set of *locations*, and  $V$  could be partitioned into two disjoint sets  $V^0$  and  $V^1$ ;  $E \subseteq V \times V$  is the set of *moves*, required to be total; and  $C : V \rightsquigarrow \mathbb{N}$  is a partial function of *coloring*, and we say a location  $v$  is *colored*, if  $C(v)$  is defined. In addition, for the game  $G$ , we require that each loop involves at least one colored location.

Two players — player 0 and player 1, are respectively in charge of  $V^0$  and  $V^1$  when  $G$  is being played. A *play* of  $G$  starting from  $v_0 \in V$  is an infinite sequence of locations  $v_0, v_1, \dots$  made by player 0 and player 1 — for every  $i \in \mathbb{N}$ , the location  $v_{i+1}$  is chosen by player 0 (resp. player 1) with  $(v_i, v_{i+1}) \in E$  whenever  $v_i \in V^0$  (resp.  $v_i \in V^1$ ).

Player 0 (resp. player 1) *wins* the play  $v_0, v_1, \dots$  if the maximal color occurring infinitely often in it is even (resp. odd) — and we say that a color  $c$  *occurs* in this play if there is some  $v_i$  with  $C(v_i) = c$ .

A winning strategy for player  $i$  is a mapping  $H_i : V^* \cdot V^i \rightarrow V$ , such that for every play  $v_0, v_1, \dots$ , player  $i$  always wins if  $v_{j+1} = H_i(v_0, \dots, v_j)$  whenever  $v_j \in V^i$ . In addition,  $H_i$  is memoryless if  $H_i(v_0, \dots, v_j)$  agrees with  $H_i(v_j)$  for every  $j$ .

**Theorem 8** ([Gurevich and Harrington, 1982; Zielonka, 1998; Jurdziński, 1998]). *For a parity game  $G$ , from every location, there is exactly one player having a winning strategy. The problem of deciding the winner at a location is in  $UP \cap \text{co-UP}$ . In addition, if a player has a winning strategy then she also has a memoryless one from the same location.*

We use  $\mathcal{W}_i(G)$  to denote the set consisting of all locations from which player  $i$  has a winning strategy.

Given a PAPA  $A = (Q, q, \delta, \Omega)$ , a gadget  $D$  of  $A$  is a finite directed acyclic digram  $(P, \gamma)$  where  $P \subseteq Q$ ,  $\gamma \subseteq P \times P$ , and for each  $q \in P$ :

1. if  $\delta(q) = q'$ , then  $q' \in P$  and  $(q, q') \in \gamma$ ;
2. if  $\delta(q) = q_1 \wedge q_2$  then  $q_1, q_2 \in P$ , and  $(q, q_1), (q, q_2) \in \gamma$ ;
3. if  $\delta(q) = q_1 \vee q_2$  then there is some  $i \in \{1, 2\}$  such that  $q_i \in P$  and  $(q, q_i) \in \gamma$ ,
4.  $q$  has no successor for the other cases.

For convenience, we sometimes directly write  $q \in D$  whenever  $D = (P, \gamma)$  and  $q \in P$ . We denote by  $\mathcal{D}(A)$  the set consisting of all gadgets of  $A$ . Since we require that each PAPA  $A$  is well-structured, then  $\mathcal{D}(A)$  must be a finite set.

Given a sequence of gadgets  $D_1, D_2, \dots$  such that  $D_i = (P_i, \gamma_i)$ , an infinite path within it is a sequence of states  $q_{1,1}, \dots, q_{1,\ell_1}, q_{2,1}, \dots, q_{2,\ell_2}, \dots$  such that each  $(q_{i,j}, q_{i,j+1}) \in \gamma_i$  and  $\delta(q_{i,\ell_i}) = \bigcirc^{\sim p_i} q_{i+1,1}$  for some  $p_i \in [0, 1]$ . We say such an infinite path is *even* (resp. *odd*) if the maximal color (w.r.t.  $\Omega$ ) occurring infinitely often is even (resp. odd).

We say that a gadget  $D = (P, \gamma)$  is *incompatible* if there exist  $q_1, q_2 \in P$  and  $\delta(q_1) = a$ ,  $\delta(q_2) = \neg a$  for some  $a \in \mathcal{A}$ ; or there is some  $q \in P$  with  $\delta(q) = \perp$ . Otherwise, we say that  $D$  is *compatible*.

Let  $D$  be a gadget and  $\Gamma = \{D_1, \dots, D_k\}$  be a set of gadgets, we denote by  $\Gamma \Vdash D$  if there exist  $k$  positive numbers  $x_1, \dots, x_k$  such that:  $\sum_{i=1}^k x_i \leq 1$ , and for each  $q \in D$  with  $\delta(q) = \bigcirc^{\sim p} q'$ , we have  $\sum_{q' \in D_i} x_i \sim p$ . We in what follows call  $x_1, \dots, x_k$  the *enabling condition*. Note that the relation  $\Vdash$  could be decided by solving a linear system of inequality.

According to automata theory, we may construct a deterministic (word) parity automaton  $\tilde{A} = (\tilde{Q}, \tilde{q}, \tilde{\delta}, \tilde{\Omega})$  where  $\tilde{\delta} : \tilde{Q} \times \mathcal{D}(A) \rightarrow \tilde{Q}$  and  $\tilde{\Omega}$  is a total coloring function. It takes a gadget sequence as input, and accepts it if every gadget in it is compatible and every infinite path within it is even.

Then, we may create a parity game  $G_A = (V_A, E_A, C_A)$  for the PAPA  $A$ , in detail:

- $V_A = V_A^0 \cup V_A^1$ , where  $V_A^0 = 2^{\mathcal{D}(A) \times \tilde{Q}}$  and  $V_A^1 = \mathcal{D}(A) \times \tilde{Q}$ .
- $E_A = \{((D_1, \tilde{q}_1), \dots, (D_k, \tilde{q}_k)), (D_i, \tilde{q}_i) \mid 1 \leq i \leq k\} \cup \{((D, \tilde{q}), ((D_1, \tilde{q}_1), \dots, (D_k, \tilde{q}_k))) \mid (D_1, \dots, D_k) \Vdash D, \text{ and each } \tilde{q}_i = \tilde{\delta}(\tilde{q}, D_i)\}$ .
- $C_A(D, \tilde{q}) = \tilde{\Omega}(\tilde{q})$ , hence every location in  $V_A^1$  is colored.

**Theorem 9.** *Let the PAPA  $A = (Q, q, \delta, \Omega)$ , then  $\mathcal{L}(A) \neq \emptyset$  if and only if there is some  $D \in \mathcal{D}(A)$  with  $q \in D$  such that  $\{(D, \tilde{\delta}(\tilde{q}, D))\} \in \mathcal{W}_0(G_A)$ .*

Intuitively, player 0 could extract a winning strategy from an accepting run of  $A$  over any pointed Markov chain; and conversely, one can construct a pointed Markov chain accepted by  $A$  according to the (memoryless) winning strategy of player 0. Interested readers may find the detailed proof in the accompanied report [Liu *et al.*, 2015].

As a consequence of Thm. 7, Thm. 8 and Thm. 9 we have the following main conclusion of this section.

**Theorem 10.** *Both the EMPTINESS problem of PAPA and the SATISFIABILITY problem of  $P\mu\text{TL}$  are decidable, and both of them are in  $2\text{EXPTIME}$ .*

Indeed, from Thm. 7 one can get a PAPA whose scale is linear in the size of the input formula, and an  $n$ -state PAPA could be converted to a parity game with scale  $2^{2^{O(n)}}$ . From standard game theory (see [Jurdziński, 1998; Wilke, 2002], and see [Schewe, 2008] for an improved bound), and with a similar analysis of [Wilke, 2002] (see also the analysis of the coloring number in that paper), one can infer that this problem is in  $2\text{EXPTIME}$ .

## 7 Discussion

In this paper, we present the logic  $P\mu\text{TL}$ , a simple and succinct probabilistic extension of  $\mu\text{TL}$ . We have compared the expressiveness of these two kinds of logics: In general,  $P\mu\text{TL}$  captures ‘local’ and ‘stepwise’ probabilities; whereas PCTL could describe ‘global’ probabilities in the system. Hence, these two logics are orthogonal and complementary, and one can obtain a more powerful and expressive logic by combing them together, as done in [Castro *et al.*, 2015]. i.e., we may use formulae like  $(\mu Z.(a \vee X^{\geq 0.8} Z)) \cup^{\geq 0.6} (\nu Z'.(b \wedge F^{> 0.3} Z'))$ . Model checking algorithm of such an extension can be acquired from those of the underlying logics.

In this paper, we have also investigated the decision problem of  $P\mu\text{TL}$ , the key issue and the most challenging part is to deal with probabilistic quantifiers when doing reduction to parity games, which is a highly nontrivial extension of the non-probabilistic case. As a cost, we have only now got an algorithm with double-exponential time complexity for solving it — in contrast, the SATISFIABILITY problem for the standard  $\mu\text{TL}$  is in  $\text{EXPTIME}$ .

## Acknowledgement

First and foremost, the authors would thank all the anonymous reviewers for the valuable and helpful comments on this paper. We would also thank Nir Piterman for his valuable comments on our work.

Wanwei Liu is supported by National Natural Science Foundation of China (Grant Nos. 61103012, 61379054 and 61272335). Lei Song is supported by Australian Research Council under Grant DP130102764. Ji Wang is supported by National Natural Science Foundation of China (Grant No. 61120106006). Lijun Zhang (corresponding author) is supported by National Natural Science Foundation of China (Grant Nos. 61428208, 61472473 and 61361136002), the CAS/SAFEA International Partnership Program for Creative Research Teams.

## References

- [Baier and Katoen, 2008] C. Baier and J.-P. Katoen. *Principles of Model Checking*. MIT Press, 2008.
- [Banieqbal and Barringer, 1987] B. Banieqbal and H. Barringer. Temporal logic with fixed points. In B. Banieqbal, H. Barringer, and A. Pnueli, editors, *Temporal Logic in Specification*, volume 398 of *Lecture Notes in Computer Science*, pages 62–74. Springer-Verlag, 1987.
- [Berezin, 2002] S. Berezin. *Model Checking and Theorem Proving: A Unified Framework*. Phd thesis, Carnegie Mellon University, Pittsburgh, PA, USA, Jan. 2002.
- [Bertrand et al., 2012] N. Bertrand, J. Fearnley, and S. Schewe. Bounded satisfiability for PCTL. In *CSL*, volume 16 of *LIPICs*, pages 92–106. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.
- [Bhatia et al., 2011] Amit Bhatia, Matthew R Maly, Lydia E Kavradi, and Moshe Y Vardi. Motion planning with complex goals. *Robotics & Automation Magazine, IEEE*, 18(3):55–64, 2011.
- [Brázdil et al., ] Tomáš Brázdil, Vojtech Forejt, Jan Kretínský, and Antonín Kucera.
- [Castro et al., 2015] P. Castro, C. Kilmurray, and N. Piterman. Tractable probabilistic  $\mu$ -calculus that expresses probabilistic temporal logics. In *32nd Symposium on Theoretical Aspects of Computer Science*, volume 30 of *Leibniz International Proceedings in Informatics*, pages 211–223, 2015.
- [Cleaveland et al., 2005] R. Cleaveland, S. Iyer, and M. Narasimha. Probabilistic temporal logics via the modal  $\mu$ -calculus. *Theor. Comput. Sci.*, 342(2-3):316–350, 2005.
- [Couvreur et al., 2003] J.-M. Couvreur, N. Saheb, and G. Sutre. An optimal automata approach to LTL model checking of probabilistic systems. In *LPAR 2003*, volume 2850 of *Lecture Notes in Computer Science*, pages 361–375. Springer, 2003.
- [Dam, 1995] M. Dam. Translating CTL\* into the modal  $\mu$ -calculus. Technical Report ECS-LFCS-90-123, Laboratory for Foundations of Computer Science, University of Edinburgh, November 1995.
- [de Alfaro and Majumdar, 2001] L. de Alfaro and R. Majumdar. Quantitative solution of omega-regular games. In *STOC*, pages 675–683. ACM, 2001.
- [Emerson and Clarke, 1980] E. Emerson and E. Clarke. Characterizing correctness properties of parallel programs using fixpoints. In *Proc. of the 7th Int. Colloquium on Automata, Languages and Programming (ICALP'80)*, volume 85 of *Lecture Notes in Computer Science*, pages 169–181. Springer-Verlag, 1980.
- [Emerson and Halpern, 1986] E. A. Emerson and J. Y. Halpern. Decision procedures and expressiveness in the temporal logics of branching time. *Journal of the ACM*, 33(1):151–178, 1986.
- [Emerson and Lei, 1986] E. A. Emerson and C. L. Lei. Efficient model checking in fragments of the propositional  $\mu$ -calculus. In *First IEEE Symposium on Logic in Computer Science*, pages 267–278. Los Alamitos: IEEE Computer Society, 1986.
- [Gurevich and Harrington, 1982] Y. Gurevich and L. Harrington. Trees, automata, and games. In *Proceeding of 14th ACM Symposium on the Theory of Computing*, pages 60–65, San Francisco, California, 1982.
- [Hansson and Jonsson, 1994] H. Hansson and B. Jonsson. A logic for reasoning about time and reliability. *FAC*, 6(5):512–535, 1994.
- [Huth and Kwiatkowska, 1997] M. Huth and M. Z. Kwiatkowska. Quantitative analysis and model checking. In *LICS*, pages 111–122. IEEE Computer Society, 1997.
- [Jurdiński, 1998] M. Jurdiński. Deciding the winner in parity games is in  $UP \cap co-UP$ . *Information Processing Letters*, 68(3):119–124, 1998.
- [Katoen, 1998] J.-P. Katoen. *Concepts, Algorithms, and Tools for Model Checking*. FAU, Lehrstuhl für Informatik VII Friedrich-Alexander Universität Erlangen-Nürnberg, 2 edition, 1998. Lecture Notes of the Course “Mechanised Validation of Parallel Systems”.
- [Kozen, 1983] D. Kozen. Results on the propositional  $\mu$ -calculus. *Theoretical Computer Science*, 27:333–354, 1983.
- [Liu et al., 2015] W. Liu, L. Song, J. Wang, and Lijun Zhang. <http://arxiv.org/abs/1504.07737>, Apr 2015.
- [McIver and Morgan, 2002] A. McIver and C. Morgan. Games, probability and the quantitative  $\mu$ -calculus  $qm\mu$ . In *LPAR*, volume 2514 of *Lecture Notes in Computer Science*, pages 292–310. Springer, 2002.
- [McIver and Morgan, 2007] A. McIver and C. Morgan. Results on the quantitative  $\mu$ -calculus  $qm\mu$ . *TOCL*, 8(1):3, 2007.
- [Mio, 2012a] M. Mio. *Game semantics for probabilistic modal  $\mu$ -calculi*. Phd thesis, The University of Edinburgh, 2012.
- [Mio, 2012b] M. Mio. Probabilistic modal  $\mu$ -calculus with independent product. *Logical Methods in Computer Science*, 8(4), 2012.
- [Morgan and McIver, 1997] C. Morgan and A. McIver. A probabilistic temporal calculus based on expectations. In L. Groves and S. Reeves, editors, *Proc. Formal Methods Pacific*, pages 4–22. Springer, 1997.
- [Pnueli, 1977] A. Pnueli. The temporal logic of programs. In *Proc. of 18th IEEE Symposium on Foundation of Computer Science (FOCS' 77)*, pages 46–57. IEEE Computer Society, 1977.
- [Schewe, 2008] S. Schewe. *Synthesis of Distributed Systems*. Phd thesis, Saarbrücken, 2008.
- [Singh, 1998] Munindar P. Singh. Applying the  $\mu$ -calculus in planning and reasoning about action. *J. Log. Comput.*, 8(3):425–445, 1998.
- [Sistla and Clarke, 1985] A. P. Sistla and E. M. Clarke. The complexity of propositional linear temporal logics. *Journal of Assoc. Comput. Mach.*, 32(3):733–749, 1985.
- [Vardi, 1985] M. Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *FOCS*, pages 327–338. IEEE Computer Society, 1985.
- [Walukiewicz, 2000] I. Walukiewicz. Completeness of Kozen’s axiomatization of the propositional  $\mu$ -calculus. *Information and Computation*, 157:142–182, 2000.
- [Wilke, 2002] T. Wilke. Alternating tree automata, parity games, and modal  $\mu$ -calculus. *Bull. Belg. Math. Soc.*, 8(2):359–391, 2002.
- [Wolper, 1983] P. Wolper. Temporal logic can be more expressive. *Information and Control*, 56(1–2):72–99, 1983.
- [Zielonka, 1998] W. Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science*, 200:135–183, 1998.