Quantitative Verification and Control via the Mu-Calculus*

Luca de Alfaro

Department of Computer Engineering, UC Santa Cruz, USA

Abstract. Linear-time properties and symbolic algorithms provide a widely used framework for system specification and verification. In this framework, the verification and control questions are phrased as *boolean* questions: a system either satisfies (or can be made to satisfy) a property, or it does not. These questions can be answered by symbolic algorithms expressed in the μ-calculus. We illustrate how the μ-calculus also provides the basis for two quantitative extensions of this approach: a *probabilistic* extension, where the verification and control problems are answered in terms of the probability with which the specification holds, and a *discounted* extension, in which events in the near future are weighted more heavily than events in the far away future.

1 Introduction

Linear-time properties and symbolic algorithms provide a widely adopted framework for the specification and verification of systems. In this framework, a *property* is a set of linear sequences of system states. Common choices for the specification of system properties are *temporal logic* [MP91] and *ω-regular automata* [BL69,Tho90]. The *verification question* asks whether a system satisfies a property, that is, whether all the sequences of states that can be produced during the activity of the system belong to the property. Similarly, the *control question* asks whether it is possible to choose (a subset of) the inputs to the system to ensure that the system satisfies a property. These questions can be answered by algorithms that operate on sets of states, and that correspond to the iterative evaluation of μ-calculus fixpoint formulas [Koz83b,EL86,BC96]. This approach is often called the *symbolic* approach to verification and control, since the algorithms are often able to take advantage of compact representations for sets of states, thus providing an efficient way to answer the verification and control questions on systems with large (and, under some conditions [HM00,dAHM01b], infinite) state spaces. The approach is completed by property-preserving equivalence relations, such as bisimulation [Mil90] (for verification) and alternating bisimulation [AHKV98] (for control).

We refer to this approach as the *boolean* setting for verification and control. Indeed, the verification question is answered in a boolean fashion (either a system

* This research was supported in part by the NSF CAREER award CCR-0132780, the NSF grant CCR-0234690, and the ONR grant N00014-02-1-0671.

R. Amadio, D. Lugiez (Eds.): CONCUR 2003, LNCS 2761, pp. 103–127, 2003.

satisfies a property, or it does not). Correspondingly, the symbolic verification algorithms are boolean in nature: the subsets of states on which they operate can be (and very often are [Bry86]) represented by their characteristic functions, that are mappings from states to $\{0,1\}$. Bisimulation itself can be seen as a binary distance function, associating distance 0 to two states if they are bisimilar, and distance 1 if they are not. In this paper, we illustrate how all the elements of this approach, namely, linear properties, symbolic algorithms, and equivalence relations, can be extended to a *quantitative* settings, where the control and verification questions are given quantitative answers, where the algorithms operate on mappings from states to real numbers, and where the equivalence relations correspond to real-valued distances [HK97,DGJP99,vBW01b,DEP02]. We consider two such quantitative settings: a *probabilistic* setting, where the verification and control questions are answered in terms of the probability that the system exhibits the desired property, and a *discounted* setting, where events in the near future are weighted more than those in the distant future. Our extensions rely on *quantitative* versions of the μ-calculus for solving the verification and control problems and, in the discounted setting, even for expressing the linear-time (discounted) specifications.

1.1 Games

We develop the theory for the case of two-player stochastic games [Sha53,Eve57, FV97], also called *concurrent probabilistic games* [dAH00], and for control goals. A *stochastic game* is played over a state space. At each state, player 1 selects a move, and simultaneously and independently, player 2 selects a move; the game then proceeds to a successor state according to a transition probability determined by the current state and by the selected moves. An outcome of a game, called *trace,* consists in the infinite sequence of states that are visited in the course of the game. We say that a linear property holds for a trace if the trace belongs to the property. A simple example of game is the game MATCHBIT. The game MATCHBIT can be in one of two states, s_{try} or s_{goal}. In state s_{try}, player 1 chooses a bit $b_1 \in \{0,1\}$, and player 2 chooses a bit $b_2 \in \{0,1\}$. If $b_1 = b_2$, the game proceeds to state s_{goal}; otherwise, the game stays in state s_{try}. The state s_{goal} is *absorbing:* once entered it, the game never leaves it.

Games are a standard model for control problems: the moves of player 1 model the inputs from the controller, while the moves of player 2 model the remaining inputs along with the internal nondeterminism of the system. Stochastic games generalize transition systems, Markov chains, Markov decision processes [Ber95], and turn-based games.[1] The verification setting can be recovered as a special case of the control setting, corresponding to games where only one player has a choice of moves.

[1] For many of these special classes of systems, there are algorithms for solving verification and control problems that have better worst-case complexity than those that can be obtained by specializing the algorithms for stochastic games. A review of the most efficient known algorithms for these structures is beyond the scope of this paper.

1.2 Probabilistic Verification and Control

In systems with probabilistic transitions, such as Markov chains, it is possible that a linear property does not hold for all traces, but nevertheless holds with positive probability. Likewise, even in games with deterministic transitions, player 1 may not be able to ensure that a property holds on all traces, but may nevertheless be able to ensure that it holds with some positive probability [dAHK98]. For example, consider again the game MATCHBIT, together with the property of reaching s_{goal} (consisting of all traces that contain s_{goal}). Starting from s_{try}, player 1 is not able to ensure that all traces reach s_{goal}: whatever sequence of bits player 1 chooses, there is always the possibility that player 2 chooses the complementary sequence, confining the game to s_{try}. Nevertheless, if player 1 chooses each bit 0 and 1 with equal probability, in each round he will proceed to s_{goal} with probability 1/2, so that s_{goal} is reached with probability 1. Another example is provided by the game MATCHONE, a variant of MATCHBIT where the bits can be chosen once only. The game MATCHONE can be in one of three states s_{try}, s_{goal}, and s_{fail}. At s_{try}, players 1 and 2 choose bits b_1 and b_2; if $b_1 = b_2$, the game proceeds to s_{goal}, otherwise it proceeds to s_{fail}. Both s_{goal} and s_{fail} are absorbing states. In the game MATCHONE, the maximal probability with which player 1 can ensure reaching s_{goal} is 1/2.

Hence, it is often of interest to consider a probabilistic version of verification and control problems, that ask the maximal probability with which a property can be guaranteed to hold. We are thus led to the problem of computing the maximal probability with which player 1 can ensure that an ω-regular property holds in a stochastic game. This problem can be solved with quantitative μ-calculus formulas that are directly derived from their boolean counterparts used to solve boolean control problems.

Specifically, [EJ91] showed that for turn-based games with deterministic transitions, the set of states from which player 1 can ensure that an ω-regular specification holds can be computed in a μ-calculus based on the set-theoretic operators $\cup, \cap$ and on the *controllable predecessor* operator *Cpre*. For a set T of states, the set $Cpre(T)$ consists of the states from which player 1 can ensure a transition to T in one step. As an example, consider the *reachability* property $\diamond T$, consisting of all the traces that contain a state in T. The set of states from which player 1 can ensure that all traces are in $\diamond T$ can be computed by letting $R_0 = T$, and for $k = 0, 1, 2, \ldots$, by letting $R_{k+1} = T \cup Cpre(R_k)$. The set R_k consists of the states from which player 1 can force the game to T in at most k steps; in a finite game, the solution is thus given by $\lim_{k\to\infty} R_k$. Computing the sequence $R_0, R_1, R_2, \ldots$ of states corresponds to evaluating by iteration the least fixpoint of $R = T \cup Cpre(R)$, which is denoted in μ-calculus as $\mu x.(T \cup Cpre(R))$: this formula is thus a μ-calculus solution formula for reachability. Solution formulas are known for general parity conditions [EJ91], and this suffices for solving games with respect to arbitrary ω-regular properties.

The solution formulas for the probabilistic setting can be obtained simply by giving a *quantitative* interpretation to the solution formulas of [EJ91]. In this quantitative interpretation, subsets of states are replaced by *state valuations* that associate with each state a real number in the interval $[0, 1]$; the set operators $\cup$, $\cap$ are replaced by the pointwise maximum and minimum operators $\sqcup$, $\sqcap$ [Rei80, FH82,Koz83a,Fel83]. The operator *Cpre* is replaced by an operator *Qpre* that, given a state valuation f, gives the state valuation $Qpre(f)$ associating with each state the maximal expectation of f that player 1 can achieve in one step.

As an example, consider again the goal $\diamond T$. Denote by $\chi(T)$ the characteristic function of T, that assigns value 1 to states in T and value 0 to states outside T. We can compute the maximal probability of reaching T by letting $f_0 = \chi(T)$ and, for $k = 0, 1, 2, \ldots$, by letting $f_{k+1} = \chi(T) \sqcup Qpre(f_k)$. It is not difficult to see that $f_k(s)$ is the maximal probability with which player 1 can reach T from state s. The limit of f_k for $k \to \infty$, which corresponds to the least fixpoint $\mu x.(\chi(T) \sqcup Qpre(x))$, associates with each state the maximal probability with which player 1 can ensure $\diamond T$. As an example, in the game MATCHBIT we have $f_0(s_{try}) = 0$ and, for $k \geq 0$, $f_k(s_{try}) = 1 - 2^{-k}$; the limit $\lim_{k\to\infty} 1 - 2^{-k} = 1$ is indeed the probability with which player 1 can ensure reaching s_{goal} from s_{try}. We note that the case for reachability games is in fact well known from classical game theory (see, e.g., [FV97]). However, this quantitative interpretation of the μ-calculus yields solution formulas for the complete set of ω-regular properties [dAM01]. Moreover, even for reachability games, the μ-calculus approach leads to simpler correctness arguments for the solution formula, since it is possible to exploit the complementation of μ-calculus and the connection between μ-calculus formulas and winning strategies in the construction of the arguments [dAM01].

1.3 Discounted Verification and Control

The probabilistic setting is quantitative with respect to states, but not with respect to traces: while state valuations are quantitative, each trace is still evaluated in a boolean way: either it is in the property, or it is not. This boolean evaluation of traces does not enable us to specify "how well" a specification is met. For instance, a trace satisfies $\diamond T$ as long as the set T of target states is reached, no matter how long it takes to reach it: no prize is placed on reaching T sooner than later, and even if T is reached in a much longer time than the reasonable life expectancy of the system, the property nevertheless holds. Furthermore, if a property *does not* hold, the boolean evaluation of traces does not provide a notion of property approximation. For example, the safety property $\square T$ is violated if a state outside T is ever reached: no prize is placed on staying in T as long as possible, and the property fails even if the system stays in T for an expected time much larger than the system's own expected life time. As these examples illustrate, the boolean evaluation of traces is sensitive to changes in behavior that occur arbitrarily late: in technical terms, ω-regular properties are not continuous in the Cantor topology, which assigns distance 2^{-k} to traces that are identical up to position $k - 1$, and differ at k. Discounted control and verification proposes to remedy this situation by weighting events that occur in the near future more heavily than events that occur in the far-away future [dAHM03].

Discounting reachability and safety properties is easy. For reachability, we assign the value α^k to traces that reach the target set after k steps, for $\alpha \in [0, 1]$; for safety, we assign the value $1-\alpha^k$ to traces that stay in the safe set of states for k steps. For more complex temporal-logic properties, however, many discounting schemes are possible. For example, a Büchi property $\Box\Diamond B$ consists of all the traces that visit a subset B of states infinitely many times [MP91]. We can discount the property $\Box\Diamond B$ in several ways: on the basis of the time required to reach B, or on the basis of the number of visits to B, or on the basis of some more complex criterion (for instance, the time required to visit B twice). On the other hand, the predecessor operators of the μ-calculus provide a natural locus for discounting the next-step future. Discounted μ-calculus replaces *Qpre* with two discounted versions, $\alpha\, Qpre$ and $(1 - \alpha) + \alpha\, Qpre$, where $\alpha \in [0, 1]$ is a discount factor. Using these operators, we can write the solution to discounted reachability games as $\phi_{\alpha\text{-}reach} = \mu x.(\chi(T) \sqcup \alpha\, Qpre(x)$, and the solution to discounted safety games as $\phi_{\alpha\text{-}safety} = \nu x.(\chi(T) \sqcap (1 - \alpha) + \alpha\, Qpre(x))$.

We propose to use discounted μ-calculus as the common basis for the specification, verification, and control of discounted properties. We define *discounted properties* as the *linear semantics* of formulas of the discounted μ-calculus. The resulting setting is continuous in the Cantor topology, and provides notions of satisfaction quality and approximation for linear properties.

1.4 Linear and Branching Semantics for the μ-Calculus

Given a formula ϕ of the μ-calculus, we can associate a *linear semantics* to ϕ by evaluating it on linear traces, and by taking the value on the first state. This linear semantics is often, but not always, related to the evaluation of the formula on the game, which we call the *branching semantics.* As an example, if we evaluate the fixpoint $\phi_{reach} = \mu x.(T \cup Cpre(R))$ on a trace $s_0, s_1, s_2, \ldots$, we have that $s_0 \in \phi_{reach}$ if there is $k \in \mathbb{N}$ such that $s_k \in T$. Hence, the linear semantics of ϕ_{reach}, denoted $[\phi_{reach}]^{\text{blin}}$, coincides with $\Diamond T$, In this case, we have that the formula ϕ_{reach}, evaluated on a game, returns exactly the states from which player 1 can ensure $[\phi_{reach}]^{\text{blin}}$. This connection does not hold for all formulas. For example, consider the formula $\psi = \mu x.(Cpre(x) \cup \nu y.(T \cap Cpre(y)))$. If we evaluate this formula on a trace $s_0, s_1, s_2, \ldots$, we can show that $s_k \in \nu y.(T \cap Cpre(y))$ iff we have $s_j \in T$ for all $j \geq k$. Hence, we have $s_0 \in \psi$ iff there is $k \in \mathbb{N}$ such that $s_j \in T$ for all $j \geq k$: in other words, the linear semantics $[\psi]^{\text{blin}}$ coincides with the co-Büchi property $\Diamond\Box T$ [Tho90]. On the other hand, the formula ϕ, evaluated on a game, does *not* correspond to the states from which player 1 can ensure $\Diamond\Box T$ (see Example 1 in Sect. 3).

In the boolean setting, the linear and branching semantics are related for all *strongly deterministic* formulas [dAHM01a], a set of formulas that includes the solution formulas for games with respect to ω-regular properties [EJ91]. We show that this correspondence carries through to the probabilistic and discounted settings. Indeed, in both the probabilistic and the discounted settings, we show that the values computed by strongly deterministic formulas is equal to the maximal expectation of their linear semantics that player 1 can ensure.

In the discounted setting, the linear semantics of discounted μ-calculus provides the specification language, and the branching semantics provides the verification algorithms. For example, the value of the discounted formula $\phi_{\alpha\text{-}reach}$ on the first state of a trace $s_0, s_1, s_2, \ldots$ is α^k, where $k = \min\{j \in \mathbb{N} \mid s_j \in T\}$. Hence, the linear semantics $[\phi_{\alpha\text{-}reach}]^{\text{blin}_\alpha}$ associates the value α^k to traces that reach T in k steps. The same formula $\phi_{\alpha\text{-}reach}$, evaluated on a game, yields the maximum value of $[\phi_{\alpha\text{-}reach}]^{\text{blin}_\alpha}$ that player 1 can achieve. Similarly, the value of $\phi_{\alpha\text{-}safety}$ on the first state of a trace $s_0, s_1, s_2, \ldots$ is $1 - \alpha^k$, where $k = \min\{j \in \mathbb{N} \mid s_j \notin T\}$. Hence, the linear semantics $[\phi_{\alpha\text{-}safety}]^{\text{blin}_\alpha}$ associates the value $1 - \alpha^k$ to traces that stay in T for k steps. The same formula $\phi_{\alpha\text{-}safety}$, evaluated on a game, yields the maximum value of $[\phi_{\alpha\text{-}safety}]^{\text{blin}_\alpha}$ that player 1 can achieve. Again, this correspondence holds for a set of formulas that includes the solution formulas of games with parity conditions.

1.5 Quantitative Equivalence Relations

The frameworks for probabilistic and discounted verification are complemented by quantitative equivalence relations [HK97,DGJP99,vBW01b]. We show that, just as CTL and CTL* characterize ordinary bisimulation [Mil90], so probabilistic and discounted μ-calculus characterize probabilistic and discounted bisimulation [dAHM03].

Credits. This paper is based on joint work with Thomas A. Henzinger and Rupak Majumdar on the connection between games, μ-calculus, and linear properties [dAHM01a,dAM01,dAHM03]. I would like to thank Marco Faella, Rupak Majumdar, Mariëlle Stoelinga, and an anonymous reviewer for reading a preliminary version of this work and for providing many helpful comments and suggestions.

2 Preliminaries

2.1 The μ-Calculus

Syntax. Let $\mathcal{P}$ be a set of predicate symbols, $\mathcal{V}$ be a set of variables, and $\mathcal{F}$ be a set of function symbols. The formulas of μ-calculus are generated by the grammar

$$\phi ::= p \mid x \mid \neg\phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid f(\phi) \mid \mu x.\phi \mid \nu x.\phi, \tag{1}$$

for predicates $p \in \mathcal{P}$, variables x, and functions $f \in \mathcal{F}$. In the two quantifications $\mu x.\phi$ and $\nu x.\phi$, we require that all occurrences of x in ϕ have even polarity, that is, they occur in the scope of an even number of negations ($\neg$). We assume that for each function $f \in \mathcal{F}$ there is *dual* function $\text{Dual}(f) \in \mathcal{F}$, with $\text{Dual}(\text{Dual}(f)) = f$. Given a closed formula ϕ of μ-calculus, the following transformations enable us to push all negations to the predicates:

$$\neg(\phi_1 \vee \phi_2) \rightsquigarrow (\neg\phi_1) \wedge (\neg\phi_2) \qquad \neg(\mu x.\phi) \rightsquigarrow \nu x.\neg\phi[\neg x/x] \tag{2}$$

$$\neg(\phi_1 \wedge \phi_2) \rightsquigarrow (\neg\phi_1) \vee (\neg\phi_2) \qquad \neg(\nu x.\phi) \rightsquigarrow \mu x.\neg\phi[\neg x/x] \tag{3}$$

$$\neg f(\phi) \rightsquigarrow \text{Dual}(f)(\neg\phi), \tag{4}$$

where $\phi[\neg x/x]$ denotes the formula in which all free occurrences of x are replaced by $\neg x$. We will be particularly interested in the of formulas in *EJ-form.* These formulas take their name from the authors of [EJ91], where it was shown that they suffice for solving turn-based games; as we will see, these formulas can be uniformly used for solving boolean, probabilistic, and discounted control problems with respect to parity conditions. For $f \in \mathcal{F}$, a formula ϕ is in *EJ-form* if it can be written as

$$\phi ::= \gamma_n x_n.\gamma_{n-1} x_{n-1} \cdots \gamma_0 x_0. \bigvee_{i=0}^{n} (\chi_i \wedge f(x_i)),$$

$$\chi ::= p \mid \neg\chi \mid \chi \vee \chi \mid \chi \wedge \chi,$$

where for $0 \leq i \leq n$, we have $x_i \in \mathcal{V}$, and where $f \in \mathcal{F}$ and $p \in \mathcal{P}$. For $0 \leq i \leq n$, the fixpoint quantifier γ_i is ν if i is even, and is μ if i is odd. A fixpoint formula ϕ is in *strongly deterministic form* [dAHM01a] iff ϕ consists of a string of fixpoint quantifiers followed by a quantifier-free part ψ generated by the following grammar:

$$\psi \ ::= \ p \mid \neg p \mid \psi \vee \psi \mid p \wedge \psi \mid \neg p \wedge \psi \mid f(\chi),$$

$$\chi \ ::= \ x \mid \chi \vee \chi.$$

Semantics. The semantics of μ-calculus is defined in terms of lattices. A *lattice* $\mathbb{L} = (E, \preceq)$ consists of a set E of elements and of a partial order $\preceq$ over E, such that every pair of elements $v_1, v_2 \in E$ has a unique greatest lower bound $v_1 \sqcap v_2$ and least upper bound $v_1 \sqcup v_2$. A lattice is *complete* if every (not necessarily finite) non-empty subset of E has a greatest lower bound and a least upper bound in E. A *value lattice* is a complete lattice together with a *negation* operator $\sim$, satisfying $\sim\sim v = v$ for all $v \in E$, and $\sim \sqcap E' = \bigsqcup\{\sim v \mid v \in E'\}$ for all $E' \subseteq E$ [Ros90, chapter 6]. A μ-calculus *interpretation* $(\mathbb{L}, [\![\cdot]\!])$ consists of a value lattice $\mathbb{L} = (E, \preceq)$ and of an interpretation $[\![\cdot]\!]$ that maps every predicate $p \in \mathcal{P}$ to a lattice element $[\![p]\!] \in E$, and that maps every function $f \in \mathcal{F}$ to a function $[\![f]\!] \in (E \mapsto E)$. We require that for all $f \in \mathcal{F}$ and all $v \in E$, we have $\sim[\![f]\!](v) = [\![\mathrm{Dual}(f)]\!](\sim v)$. A *variable environment* is a function $e : \mathcal{V} \mapsto E$ that associates a lattice element $e(x) \in E$ to each variable $x \in \mathcal{V}$. For $x \in \mathcal{V}$, $v \in E$, and a variable environment e, we denote by $e[x := v]$ the variable environment defined by $e[x := v](x) = v$, and $e[x := v](y) = e(y)$ for $x \neq y$. Given an interpretation $\mathcal{I} = (\mathbb{L}, [\![\cdot]\!])$, and a variable environment e, every μ-calculus formula ϕ specifies a lattice element $[\![\phi]\!]_e^{\mathcal{I}} \in E$, defined inductively as follows, for $p \in \mathcal{P}$, $f \in \mathcal{F}$, and $x \in \mathcal{V}$:

$$[\![p]\!]_e^{\mathcal{I}} = [\![p]\!] \qquad [\![\phi_1 \vee \phi_2]\!]_e^{\mathcal{I}} = [\![\phi_1]\!]_e^{\mathcal{I}} \sqcup [\![\phi_2]\!]_e^{\mathcal{I}}$$

$$[\![\neg p]\!]_e^{\mathcal{I}} = \sim[\![p]\!]_e^{\mathcal{I}} \qquad [\![\phi_1 \wedge \phi_2]\!]_e^{\mathcal{I}} = [\![\phi_1]\!]_e^{\mathcal{I}} \sqcap [\![\phi_2]\!]_e^{\mathcal{I}}$$

$$[\![x]\!]_e^{\mathcal{I}} = e(x) \qquad [\![\mu x.\phi]\!]_e^{\mathcal{I}} = \sqcap\{v \in E \mid v = [\![\phi]\!]_{e[x:=v]}^{\mathcal{I}}\}$$

$$[\![f(\phi)]\!]_e^{\mathcal{I}} = [\![f]\!]([\![\phi]\!]_e^{\mathcal{I}}) \qquad [\![\nu x.\phi]\!]_e^{\mathcal{I}} = \sqcup\{v \in E \mid v = [\![\phi]\!]_{e[x:=v]}^{\mathcal{I}}\}.$$

All right-hand-side (semantic) operations are performed over the value lattice $\mathbb{L}$. It is easy to show that if $\phi \rightsquigarrow \phi'$ by (2)–(4), then $[\![\phi]\!]_e^{\mathcal{I}} = [\![\phi']\!]_e^{\mathcal{I}}$. A μ-calculus formula ϕ is *closed* if all its variables are bound by one of the μ or ν fixpoint quantifiers. If ϕ is closed, then the value $[\![\phi]\!]_e^{\mathcal{I}}$ does not depend on e, and we write simply $[\![\phi]\!]^{\mathcal{I}}$.

2.2 Game Structures

We develop the theory for stochastic game structures. For a finite set A, we denote by $\mathrm{Distr}(A)$ the set of probability distributions over A. A (two-player stochastic) *game structure* $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta \rangle$ consists of the following components [AHK02,dAHK98]:

- A finite set S of states.
- A finite set $\mathcal{M}$ of moves.
- Two move assignments Γ_1, Γ_2: $S \mapsto 2^{\mathcal{M}} \setminus \emptyset$. For $i \in \{1, 2\}$, the assignment Γ_i associates with each state $s \in S$ the nonempty set $\Gamma_i(s) \subseteq \mathcal{M}$ of moves available to player i at state s.
- A probabilistic transition function δ: $S \times \mathcal{M}^2 \mapsto \mathrm{Distr}(S)$, that gives the probability $\delta(s, a_1, a_2)(t)$ of a transition from s to t when player 1 plays move a_1 and player 2 plays move a_2.

At every state $s \in S$, player 1 chooses a move $a_1 \in \Gamma_1(s)$, and simultaneously and independently player 2 chooses a move $a_2 \in \Gamma_2(s)$. The game then proceeds to the successor state $t \in S$ with probability $\delta(s, a_1, a_2)(t)$. We denote by $\tau(s, a_1, a_2) = \{t \in S \mid \delta(s, a_1, a_2)(t) > 0\}$ the set of *destination states* when actions a_1, a_2 are chosen at s. In general, the players can randomize their choice of moves at a state. We denote by $\mathcal{D}_i(s) \subseteq \mathrm{Distr}(\mathcal{M})$ the set of move distributions available to player $i \in \{1, 2\}$ at $s \in S$, defined by

$$\mathcal{D}_i(s) = \{\zeta \in \mathrm{Distr}(\mathcal{M}) \mid \zeta(a) > 0 \text{ } \textit{implies} \text{ } a \in \Gamma_i(s)\}.$$

For $s \in S$ and $\zeta_1 \in \mathcal{D}_1(s)$, $\zeta_2 \in \mathcal{D}_2(s)$, we denote by $\hat{\delta}(s, \zeta_1, \zeta_2)$ the next-state probability distribution, defined for all $t \in S$ by

$$\hat{\delta}(s, \zeta_1, \zeta_2)(t) = \sum_{a_1 \in \Gamma_1(s)} \sum_{a_2 \in \Gamma_2(s)} \delta(s, a_1, a_2)(t)\, \zeta_1(a_1)\, \zeta_2(a_2).$$

A (randomized) *strategy* π_i for player $i \in \{1, 2\}$ is a mapping $\pi_i : S^+ \mapsto \mathrm{Distr}(\mathcal{M})$ that associates with every sequence of states $\overline{s} \in S^+$ the move distribution $\pi_i(\overline{s})$ used by player i when the past history of the game is $\overline{s}$; we require that $\pi_i(\overline{s}s) \in \mathcal{D}_i(s)$ for all $\overline{s} \in S^*$ and $s \in S$. We indicate with Π_i the set of all strategies for player $i \in \{1, 2\}$.

Given an initial state $s \in S$ and two strategies $\pi_1 \in \Pi_1$ and $\pi_2 \in \Pi_2$, we define the set $\mathit{Outcomes}(s, \pi_1, \pi_2) \subseteq S^\omega$ to consist of all the sequences of states $s_0, s_1, s_2, \ldots$ such that $s_0 = s$ and such that for all $k \geq 0$ there are moves $a_1^k, a_2^k \in \mathcal{M}$ such that $\pi_1(s_0, \ldots, s_k)(a_1^k) > 0$, $\pi_2(s_0, \ldots, s_k)(a_2^k) > 0$, and

$s_{k+1} \in \tau(s_k, a_1^k, a_2^k)$. Given a trace $\sigma = s_0, s_1, s_2, \ldots \in S^\omega$, we denote by σ_k its k-th state s_k.

An initial state $s \in S$ and two strategies $\pi_1 \in \Pi_1$ and $\pi_2 \in \Pi_2$ uniquely determine a stochastic process $(S^\omega, \Omega, \mathrm{Pr}_s^{\pi_1,\pi_2})$ where $\Omega \subseteq 2^{S^\omega}$ is the set of measurable sets, and where $\mathrm{Pr}_s^{\pi_1,\pi_2} : \Omega \mapsto [0,1]$ assigns a probability to each measurable set [KSK66,FV97]. In particular, for a measurable set of traces $A \in \Omega$, we denote by $\mathrm{Pr}_s^{\pi_1,\pi_2}(A)$ the probability that the game follows a trace in A, and given a measurable function $f : S^\omega \mapsto \mathbb{R}$, we denote by $\mathrm{E}_s^{\pi_1,\pi_2}(f)$ the expectation of f in $(S^\omega, \Omega, \mathrm{Pr}_s^{\pi_1,\pi_2})$.

We denote by $S_{\mathcal{G}}$, $\mathcal{M}_{\mathcal{G}}$, $\Gamma_1^{\mathcal{G}}$, $\Gamma_2^{\mathcal{G}}$, and $\delta_{\mathcal{G}}$ the individual components of a game structure $\mathcal{G}$.

Special Classes of Game Structures. Transition systems, turn-based games, and Markov decision processes are special cases of deterministic game structures. A game structure $\mathcal{G}$ is *deterministic* if for all states $s, t \in S_{\mathcal{G}}$ and all moves $a_1, a_2 \in \mathcal{M}_{\mathcal{G}}$ we have $\delta_{\mathcal{G}}(s, a_1, a_2)(t) \in \{0, 1\}$. The structure $\mathcal{G}$ is *turn-based* if at every state at most one player can choose among multiple moves; that is, if for all states $s \in S_{\mathcal{G}}$, there exists at most one $i \in \{1, 2\}$ with $|\Gamma_i^{\mathcal{G}}(s)| > 1$. The turn-based deterministic game structures coincide with the games of [BL69, Con92,Tho95]. For $i \in \{1, 2\}$, the structure $\mathcal{G}$ is *player-i* if at every state only player i can choose among multiple moves; that is, if $|\Gamma_{3-i}^{\mathcal{G}}(s)| = 1$ for all states $s \in S$. Player-1 and player-2 structures (called collectively *one-player* structures) coincide with Markov decision processes [Der70]. The player-i deterministic game structures coincide with transition systems: in every state, each available move of player i determines a unique successor state.

3 Boolean Verification and Control

Given a game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta \rangle$, a *linear property* of $\mathcal{G}$ is a subset $\Phi \subseteq S^\omega$ of its state sequences. Given a linear property $\Phi \subseteq S^\omega$, we let

$$\langle 1 \rangle_{\mathcal{G}}^{\mathrm{b}} \Phi = \{ s \in S \mid \exists \pi_1 \in \Pi_1 . \forall \pi_2 \in \Pi_2 . \mathit{Outcomes}(s, \pi_1, \pi_2) \subseteq \Phi \} \quad (5)$$

$$\langle 2 \rangle_{\mathcal{G}}^{\mathrm{b}} \Phi = \{ s \in S \mid \exists \pi_2 \in \Pi_2 . \forall \pi_1 \in \Pi_1 . \mathit{Outcomes}(s, \pi_1, \pi_2) \subseteq \Phi \} . \quad (6)$$

The set $\langle 1 \rangle_{\mathcal{G}}^{\mathrm{b}} \Phi$ is the set of states from which player 1 can ensure that the game outcome is in Φ; the set $\langle 2 \rangle_{\mathcal{G}}^{\mathrm{b}} \Phi$ is the symmetrically defined set for player 2. We consider the control problems of computing the sets (5) and (6). We note that for player-1 deterministic game structures, computing (5) corresponds to solving the *existential* verification problem "is there a trace in Φ?", and for player-2 game structures computing (5) corresponds to solving the *universal* verification problem "are all traces in Φ?". We review the well-known solution of these control problems for the case in which Φ is a reachability property, a safety property, and a parity property. For a subset $T \subseteq S$ of states, the *safety* property $\Box T = \{ s_0, s_1, s_2, \ldots \in S^\omega \mid \forall k . s_k \in T \}$ consists of all traces that stay always in T, and the *reachability* property $\Diamond T = \{ s_0, s_1, s_2, \ldots \in S^\omega \mid \exists k . s_k \in T \}$ consists of all traces that contain a state in T. Consider any tuple $\mathcal{A} = \langle T_1, T_2, \ldots, T_m \rangle$ such that $T_1, T_2, \ldots, T_m$ form a partition of S into $m > 0$ mutually disjoint subsets. Given a trace $\sigma = s_0, s_1, s_2, \ldots \in S^\omega$, we denote by $\mathit{Index}(\sigma, \mathcal{A})$ the largest $i \in \{1, \ldots, m\}$ such that $s_k \in T_i$ for infinitely many $k \in \mathbb{N}$. Then, the *parity* property $\mathit{Parity}(\mathcal{A})$ is defined as $\mathit{Parity}(\mathcal{A}) = \{\sigma \in S^\omega \mid \mathit{Index}(\sigma, \mathcal{A}) \text{ is even}\}$. The relevance of parity properties is due to the fact that any ω-regular property can be specified with a deterministic automaton with a parity accepting condition [Tho90]. Hence, we can transform any verification problem with respect to an ω-regular condition into a verification problem with respect to a parity condition by means of a simple automaton product construction (see for instance [dAHM01a]).

3.1 Boolean μ-Calculus

For all three classes of properties (safety, reachability, and parity), the solution of the boolean control problems (5)–(6) can be written in μ-calculus interpreted over the lattice of subsets of states. Precisely, given a set S of states, the set $\mathcal{BMC}_S$ of *boolean μ-calculus* formulas consists of all μ-calculus formulas defined over the set of predicates $\mathcal{P}_S = 2^S$ and the set of functions $\mathcal{F}_b = \{\mathit{pre}_1, \mathit{dpre}_1, \mathit{pre}_2, \mathit{dpre}_2\}$, where $\mathrm{Dual}(\mathit{pre}_1) = \mathit{dpre}_1$ and $\mathrm{Dual}(\mathit{pre}_2) = \mathit{dpre}_2$. Given a game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta \rangle$, we interpret the formulas of $\mathcal{BMC}_S$ over the lattice $\mathbb{L}(2^S, \subseteq)$ of subsets of S, ordered according to set inclusion. Negation is set complementation: for all $T \subseteq S$, we let $\sim T = S \setminus T$. The predicates are interpreted as themselves: for all $p \in \mathcal{P}$, we let $[\![p]\!]^{\mathrm{b}}_{\mathcal{G}} = p$. The functions pre_1, dpre_1, pre_2, and dpre_2 are called *predecessor operators,* and they are interpreted as follows:

$$[\![\mathit{pre}_1]\!]^{\mathrm{b}}_{\mathcal{G}}(X) = \{s \in S \mid \exists a_1 \in \Gamma_1(s).\forall a_2 \in \Gamma_2(s).\tau(s, a_1, a_2) \subseteq X\} \tag{7}$$

$$[\![\mathit{dpre}_1]\!]^{\mathrm{b}}_{\mathcal{G}}(X) = \{s \in S \mid \forall a_1 \in \Gamma_1(s).\exists a_2 \in \Gamma_2(s).\tau(s, a_1, a_2) \cap X \neq \emptyset\} \tag{8}$$

$$[\![\mathit{pre}_2]\!]^{\mathrm{b}}_{\mathcal{G}}(X) = \{s \in S \mid \exists a_2 \in \Gamma_2(s).\forall a_1 \in \Gamma_1(s).\tau(s, a_1, a_2) \subseteq X\} \tag{9}$$

$$[\![\mathit{dpre}_2]\!]^{\mathrm{b}}_{\mathcal{G}}(X) = \{s \in S \mid \forall a_2 \in \Gamma_2(s).\exists a_1 \in \Gamma_1(s).\tau(s, a_1, a_2) \cap X \neq \emptyset\}. \tag{10}$$

Intuitively, $[\![\mathit{pre}_1]\!]^{\mathrm{b}}_{\mathcal{G}}(X)$ is the set of states from which player 1 can force a transition to X in $\mathcal{G}$, and $[\![\mathit{dpre}_1]\!]^{\mathrm{b}}_{\mathcal{G}}(X)$ is the set of states from which player 1 is unable to avoid a transition to X in $\mathcal{G}$. The functions pre_2 and dpre_2 are interpreted symmetrically. We denote by $\mathit{bin}(\mathcal{G}) = (\mathbb{L}(2^S, \subseteq), [\![\cdot]\!]^{\mathrm{b}}_{\mathcal{G}})$ the resulting interpretation for μ-calculus. For a closed formula $\phi \in \mathcal{BMC}_S$, we write $[\![\phi]\!]^{\mathrm{b}}_{\mathcal{G}}$ rather than $[\![\phi]\!]^{\mathit{bin}(\mathcal{G})}$, and we omit $\mathcal{G}$ when clear from the context. For a game $\mathcal{G}$, a subset $T \subseteq S_{\mathcal{G}}$ of states and player $i \in \{1, 2\}$, we have then

$$\langle i \rangle^{\mathrm{b}}_{\mathcal{G}} \Diamond T = [\![\mu x.(T \vee \mathit{pre}_i(x))]\!]^{\mathrm{b}}_{\mathcal{G}}. \tag{11}$$

This formula can be understood by considering its iterative computation: we have that $[\![\mu x.(T \vee \mathit{pre}_i(x))]\!]^{\mathrm{b}}_{\mathcal{G}} = \lim_{k \to \infty} X_k$, where $X_0 = \emptyset$ and, for $k \in \mathbb{N}$, where $X_{k+1} = T \cup [\![\mathit{pre}_i]\!]^{\mathrm{b}}_{\mathcal{G}}(X_k)$: it is easy to show by induction that the set that

X_k consists of the states of S from which player i can force the game to T in at most k steps. The equation (11) then follows by taking the limit $k \to \infty$. Similarly, for safety properties we have, for $i \in \{1, 2\}$:

$$\langle i \rangle^{\mathrm{b}}_{\mathcal{G}} \Box T = [\![\nu x.(T \wedge pre_i(x))]\!]^{\mathrm{b}}_{\mathcal{G}}. \tag{12}$$

Again, to understand this formula it helps to consider its iterative computation. We have $[\![\nu x.(T \wedge pre_1(x))]\!]^{\mathrm{b}}_{\mathcal{G}} = \lim_{k \to \infty} X_k$, where $X_0 = S$ and, for $k \in \mathbb{N}$, where $X_{k+1} = T \cap [\![pre_1]\!]^{\mathrm{b}}_{\mathcal{G}}(X_k)$; it is easy to see that X_k consists of the states of $\mathcal{G}$ from which player 1 can guarantee that the game stays in T for at least k steps. The equation (12) then follows by taking the limit $k \to \infty$. The solution of control and verification problems for parity properties is given by the following result.

Theorem 1. [EJ91] *For all game structures $\mathcal{G}$, all partitions $\langle T_1, T_2, \dots, T_m \rangle$ of $S_{\mathcal{G}}$, and all $i \in \{1, 2\}$, we have*

$$\langle i \rangle^{\mathrm{b}}_{\mathcal{G}} Parity(\langle T_1, \dots, T_m \rangle) = [\![\gamma_m x_m \cdots \gamma_1 x_1 . \bigvee_{j=1}^{m} (T_j \wedge pre_i(x_j))]\!]^{\mathrm{b}}_{\mathcal{G}} \tag{13}$$

Given an EJ-form μ-calculus formula $\phi = \gamma_m x_m \cdots \gamma_1 x_1 . \bigvee_{j=1}^{m} (T_j \wedge pre_1(x_j))$, we define the parity property $PtyOf(\phi)$ corresponding to ϕ by $PtyOf(\phi) = Parity(\langle T_1, \dots, T_m \rangle)$. With this notation, we can restate (13) as follows.

Corollary 1. *For all game structures $\mathcal{G}$, all $i \in \{1, 2\}$ and all closed EJ-form μ-calculus formulas $\phi \in \mathcal{BMC}_{S_{\mathcal{G}}}$ containing only the function symbol pre_i, we have $\langle i \rangle^{\mathrm{b}}_{\mathcal{G}} PtyOf(\phi) = [\![\phi]\!]^{\mathrm{b}}_{\mathcal{G}}$.*

Lack of Determinacy. In boolean μ-calculus, the operators pre_1 and pre_2 are not the dual one of the other. This implies that boolean control problems are not determined: for $\Phi \subseteq S^\omega$, the equality $S \setminus \langle 1 \rangle^{\mathrm{b}}_{\mathcal{G}} \Phi = \langle 2 \rangle^{\mathrm{b}}_{\mathcal{G}} (S^\omega \setminus \Phi)$ does *not* hold for all game structures $\mathcal{G}$ and all properties Φ. Intuitively, the fact that player 1 is unable to ensure the control goal Φ does not entail that player 2 is able to ensure the control goal $\neg\Phi$. For example, there are game structures $\mathcal{G}$ where for some $T \subseteq S_{\mathcal{G}}$ we have

$$\begin{aligned} S_{\mathcal{G}} \setminus \left(\langle 1 \rangle^{\mathrm{b}}_{\mathcal{G}} \Diamond T \right) &= [\![\neg \mu x.(T \vee pre_1(x))]\!]^{\mathrm{b}}_{\mathcal{G}} = [\![\nu x.(\neg T \wedge dpre_1(x))]\!]^{\mathrm{b}}_{\mathcal{G}} \\ &\neq [\![\nu x.(\neg T \wedge pre_2(x))]\!]^{\mathrm{b}}_{\mathcal{G}} = \langle 2 \rangle^{\mathrm{b}}_{\mathcal{G}} \Box (\neg T). \end{aligned}$$

An example is the game structure MATCHBIT: as explained in the introduction we have $s_{try} \not\in \langle 1 \rangle^{\mathrm{b}} \Diamond \{s_{goal}\}$; on the other hand, it can be easily seen that $s_{try} \not\in \langle 2 \rangle^{\mathrm{b}} \Box \{s_{try}\}$.

3.2 The Linear Semantics of Boolean μ-Calculus

Theorem 1 establishes a basic connection between linear parity properties and verification algorithms expressed in μ-calculus. Here, we shall develop a connection between linear properties *expressed in μ-calculus,* and their verification algorithms also expressed in μ-calculus. To this end, we provide a linear semantics for μ-calculus, obtained by evaluating μ-calculus on linear traces.

A trace $\sigma \in S^\omega$ gives rise to an interpretation $\mathcal{I}^b_\sigma = (\mathbb{L}(2^{\mathbb{N}}, \subseteq), [\![\cdot]\!]^{\rm b}_\sigma)$ for μ-calculus, where $\mathbb{L}(2^{\mathbb{N}}, \subseteq)$ is the lattice of subsets of natural numbers ordered according to set inclusion, and where all predicates $p \in 2^S$ are interpreted as the sets of indices of states in p, i.e., $[\![p]\!]^{\rm b}_\sigma = \{k \in \mathbb{N} \mid \sigma_k \in p\}$. The definitions (7)–(10) can be simplified, since every location of the trace has a single successor: for all $i \in \{1, 2\}$ and all $X \subseteq \mathbb{N}$ we let $[\![pre_i]\!]^{\rm b}_\sigma(X) = [\![dpre_i]\!]^{\rm b}_\sigma(X) = \{k \in \mathbb{N} \mid k+1 \in X\}$. We define the *boolean linear semantics* $[\phi]^{\text{blin}_S}$ over the set of states S of a closed μ-calculus formula $\phi \in \mathcal{BMC}_S$ to consist of all traces whose first state is in the semantics of ϕ: specifically, $[\phi]^{\text{blin}_S} = \{\sigma \in S^\omega \mid \sigma_0 \in [\![\phi]\!]^{\mathcal{I}^b_\sigma}\}$. In contrast, we call the semantics $[\![\phi]\!]^{\rm b}_{\mathcal{G}}$ defined over a game structure $\mathcal{G}$ the *branching* semantics of the μ-calculus formula ϕ. The following lemma states that for formulas in EJ-form, the parity property corresponding to the formula coincides with the linear semantics of the formula.

Lemma 1. *For all sets of states S, all $i \in \{1, 2\}$ and all closed EJ-form μ-calculus formulas $\phi \in \mathcal{BMC}_S$ containing only the function symbol pre_i, we have $PtyOf(\phi) = [\phi]^{\text{blin}_S}$.*

This leads easily to the following result, which relates the linear and branching semantics of EJ-form formulas.

Corollary 2. *For all game structures $\mathcal{G}$, all $i \in \{1, 2\}$ and all closed EJ-form μ-calculus formulas $\phi \in \mathcal{BMC}_{S_\mathcal{G}}$ containing only the function symbol pre_i, we have $[\![\phi]\!]^{\rm b}_{\mathcal{G}} = \langle i \rangle^{\rm b}_{\mathcal{G}} PtyOf(\phi) = \langle i \rangle^{\rm b}_{\mathcal{G}} [\phi]^{\text{blin}_{S_\mathcal{G}}}$.*

In fact, the relationship between the linear and branching semantics holds for all μ-calculus formulas in strongly deterministic form.

Theorem 2. [dAHM01a] *For all game structures $\mathcal{G}$, all closed μ-calculus formulas $\phi \in \mathcal{BMC}_{S_\mathcal{G}}$, and all players $i \in \{1, 2\}$, if ϕ is in strongly deterministic form and contains only the function symbol pre_i, then $[\![\phi]\!]^{\rm b}_{\mathcal{G}} = \langle i \rangle^{\rm b}_{\mathcal{G}} [\phi]^{\text{blin}_{S_\mathcal{G}}}$.*

We will see that the linear and branching semantics of strongly deterministic (and in particular, EJ-form) formulas are related in all the settings considered in this paper, namely, in the boolean, probabilistic, and discounted settings. The linear and branching semantics of formulas are not always related, as the following example demonstrates.

Example 1. [dAHM01a] Consider the formula $\phi = \mu x.(pre_1(x) \vee \nu y.(B \wedge pre_1(y)))$, where $B \subseteq S$ is a set of states. The linear semantics $[\phi]^{\text{blin}_S}$ consists of all the traces $\sigma = s_0, s_1, s_2, \ldots$ for which there is a $k \in \mathbb{N}$ such that $s_i \in B$ for all $i \geq k$, that is, of all the traces that eventually enter B, and never leave it again; using temporal-logic notation, we indicate this set of traces by $[\Diamond\Box B]_S$. In fact, we have $s_k \in [\![\nu y.(B \wedge pre_1(y))]\!]^{\mathcal{I}^b_\sigma}$ only if $s_i \in B$ for all $i \geq k$, and we have that $s_0 \in [\![\phi]\!]^{\mathcal{I}^b_\sigma}$ iff there is $k \in \mathbb{N}$ such that $s_k \in [\![\nu y.(B \wedge pre_1(y))]\!]^{\mathcal{I}^b_\sigma}$. However, consider a deterministic player-2 structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta \rangle$ with $S = \{s, t, u\}$, $\mathcal{M} = \{a, b, \bullet\}$, and $\Gamma_2(s) = \{a, b\}$, $\Gamma_2(t) = \Gamma_2(u) = \{a\}$, and transition relation given by $\tau(s, \bullet, a) = \{s\}$, $\tau(s, \bullet, b) = \{t\}$, $\tau(t, \bullet, a) = \{u\}$, $\tau(u, \bullet, a) = \{u\}$, where

• is the single move available to player 1. For $B = \{s, u\}$ it is easy to see that $\langle 1\rangle^{\mathrm{b}}_{\mathcal{G}}[\phi]^{\mathrm{blin}_S} = \langle 1\rangle^{\mathrm{b}}_{\mathcal{G}}[\Diamond\Box B]^{\mathrm{blin}_S} = \{s, t, u\}$, while $[\![\phi]\!]^{\mathrm{b}}_{\mathcal{G}} = \{t, u\}$. ∎

In [dAHM01a], it is shown that in general the linear and branching semantics of μ-calculus formulas are related on all game structures iff they are related on all player-1 and all player-2 game structures.

4 Probabilistic Verification and Control

The boolean control problem asks whether a player can guarantee that all outcomes are in a desired linear property. The *probabilistic* control problem asks what is the *maximal probability* with which a player can guarantee that the outcome of the game belongs to the desired linear property. Given a game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta\rangle$ and a property $\Phi \subseteq S^\omega$, we consider the two *probabilistic control problems* consisting in computing the functions $\langle 1\rangle^{\mathrm{p}}_{\mathcal{G}}\Phi, \langle 2\rangle^{\mathrm{p}}_{\mathcal{G}}\Phi : S \mapsto [0, 1]$ defined by:

$$\langle 1\rangle^{\mathrm{p}}_{\mathcal{G}}\Phi = \lambda s \in S. \sup_{\pi_1 \in \Pi_1} \inf_{\pi_2 \in \Pi_2} \mathrm{Pr}^{\pi_1,\pi_2}_s(\Phi) \tag{14}$$

$$\langle 2\rangle^{\mathrm{p}}_{\mathcal{G}}\Phi = \lambda s \in S. \sup_{\pi_2 \in \Pi_2} \inf_{\pi_1 \in \Pi_1} \mathrm{Pr}^{\pi_1,\pi_2}_s(\Phi). \tag{15}$$

where $\lambda s \in S.f(s)$ is the usual λ-calculus notation for a function that maps each $s \in S$ into $f(s)$.

4.1 Probabilistic μ-Calculus

For the case in which Φ is a reachability, safety, or parity property, we can compute the functions (14), (15) using a *probabilistic* interpretation of μ-calculus [dAM01]. Precisely, given a set S of states, the set $\mathcal{PMC}_S$ of *probabilistic μ-calculus* formulas consists of all μ-calculus formulas defined over the set of predicates $\mathcal{P}_S = 2^S$ and the set of functions $\mathcal{F}_q = \{pre_1, pre_2\}$, where $\mathrm{Dual}(pre_1) = pre_2$. Given a game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta\rangle$, we interpret these formulas over the lattice $\mathbb{L}(S \mapsto [0, 1], \leq)$ of functions $S \mapsto [0, 1]$, ordered pointwise: for $f, g : S \mapsto [0, 1]$ and $s \in S$, we have $(f \sqcup g)(s) = \max\{f(s), g(s)\}$ and $(f \sqcap g)(s) = \min\{f(s), g(s)\}$. Negation is defined by $\sim f = \lambda s \in S.1 - f(s)$. The predicates are interpreted as *characteristic functions:* for all $p \in \mathcal{P}_S$, we let $[\![p]\!]^{\mathrm{p}}_{\mathcal{G}} = \chi(p)$, where $\chi(p)$ is defined for all $s \in S$ by $\chi(p)(s) = 1$ if $s \in p$, and $\chi(p)(s) = 0$ otherwise. The interpretations of pre_1 and pre_2 are defined as follows, for $X : S \mapsto [0, 1]$:

$$[\![pre_1]\!]^{\mathrm{p}}_{\mathcal{G}}(X) = \lambda s \in S. \sup_{\zeta_1 \in \mathcal{D}_1(s)} \inf_{\zeta_2 \in \mathcal{D}_2(s)} \mathrm{E}_\circ(X \mid s, \zeta_1, \zeta_2) \tag{16}$$

$$[\![pre_2]\!]^{\mathrm{p}}_{\mathcal{G}}(X) = \lambda s \in S. \sup_{\zeta_2 \in \mathcal{D}_2(s)} \inf_{\zeta_1 \in \mathcal{D}_1(s)} \mathrm{E}_\circ(X \mid s, \zeta_1, \zeta_2) \tag{17}$$

where $\mathrm{E}_\circ(X \mid s, \zeta_1, \zeta_2) = \sum_{t \in S} \hat{\delta}(s, \zeta_1, \zeta_2)(t)\, X(t)$ is the next-step expectation of X, given that player 1 and player 2 choose their moves according to distributions ζ_1 and ζ_2, respectively. Intuitively, $[\![pre_1]\!]^{\mathrm{p}}_{\mathcal{G}}(X)$ is the function that associates with each $s \in X$ the maximal expectation of X that player 1 can achieve in one step. In particular, for $T \subseteq S$, $[\![pre_i]\!]^{\mathrm{p}}_{\mathcal{G}}(\chi(T))$ is the maximal probability with which player i can force a transition to T in one step. We note that, unlike in the boolean case, in probabilistic μ-calculus the operators pre_1 and pre_2 are dual, so that the calculus requires only two predecessor operators, rather than four. The duality follows directly from the minimax theorem [vN28]: for all $X : S \mapsto [0,1]$ and all $s \in S$, we have

$$\begin{aligned} 1 - [\![pre_1]\!]^{\mathrm{p}}_{\mathcal{G}}(X)(s) &= 1 - \sup\nolimits_{\zeta_1 \in \mathcal{D}_1(s)} \inf\nolimits_{\zeta_2 \in \mathcal{D}_2(s)} \mathrm{E}_\circ(X \mid s, \zeta_1, \zeta_2) \\ &= \inf\nolimits_{\zeta_1 \in \mathcal{D}_1(s)} \sup\nolimits_{\zeta_2 \in \mathcal{D}_2(s)} 1 - \mathrm{E}_\circ(X \mid s, \zeta_1, \zeta_2) \\ &= \sup\nolimits_{\zeta_2 \in \mathcal{D}_2(s)} \inf\nolimits_{\zeta_1 \in \mathcal{D}_1(s)} \mathrm{E}_\circ(\sim X \mid s, \zeta_1, \zeta_2) \\ &= [\![pre_2]\!]^{\mathrm{p}}_{\mathcal{G}}(\sim X)(s). \end{aligned}$$

We denote by $prb(\mathcal{G}) = (\mathbb{L}(S \mapsto [0,1], \leq), [\![\cdot]\!]^{\mathrm{p}}_{\mathcal{G}})$ the resulting interpretation for μ-calculus. For a closed formula $\phi \in \mathcal{BMC}_S$, we write $[\![\phi]\!]^{\mathrm{p}}_{\mathcal{G}}$ rather than $[\![\phi]\!]^{prb(\mathcal{G})}$, and we omit $\mathcal{G}$ when clear from the context. The solutions to probabilistic control problems with respect to reachability, safety, and parity properties can then be written in μ-calculus as stated by the following theorem.

Theorem 3. [dAM01] *For all game structures $\mathcal{G}$, all $i \in \{1,2\}$, all $T \subseteq S_{\mathcal{G}}$ and all partitions $\mathcal{A} = \langle T_1, T_2, \ldots, T_m \rangle$ of $S_{\mathcal{G}}$, we have:*

$$\langle i \rangle^{\mathrm{p}}_{\mathcal{G}} \Diamond T = [\![\mu x.(T \vee pre_i(x))]\!]^{\mathrm{p}}_{\mathcal{G}} \tag{18}$$

$$\langle i \rangle^{\mathrm{p}}_{\mathcal{G}} \Box T = [\![\nu x.(T \wedge pre_i(x))]\!]^{\mathrm{p}}_{\mathcal{G}} \tag{19}$$

$$\langle i \rangle^{\mathrm{p}}_{\mathcal{G}} Parity(\langle T_1, \ldots, T_m \rangle) = [\![\gamma_m x_m \cdots \gamma_1 x_1 . \bigvee_{i=1}^{m} (T_i \wedge pre_i(x_i))]\!]^{\mathrm{p}}_{\mathcal{G}}. \tag{20}$$

The above solution formulas are the analogous to (11), (12), and (13), even though the proof of their correctness requires different arguments. The argument for reachability games is as follows. The fixpoint (18) can be computed iteratively by $[\![\mu x.(T \vee pre_i(x))]\!]^{\mathrm{p}}_{\mathcal{G}} = \lim_{k \to \infty} X_k$, where $X_0 = \lambda s.0$ and, for $k \in \mathbb{N}$, where $X_{k+1} = \chi(T) \sqcup [\![pre_i]\!]^{\mathrm{p}}_{\mathcal{G}}(X_k)$. It is then easy to show by induction that $X_k(s)$ is the maximal probability with which player i can reach T from $s \in S$ in at most k steps. In fact, (18) is simply a restatement in μ-calculus of the well-known fixpoint characterization of the solution of positive stochastic games (see, e.g., [FV97]). The solution (19) can also be understood in terms of the iterative evaluation of the fixpoint. We have $[\![\nu x.(T \wedge pre_i(x))]\!]^{\mathrm{p}}_{\mathcal{G}} = \lim_{k \to \infty} X_k$, where $X_0 = \lambda s.1$, and for $x \in \mathbb{N}$, where $X_{k+1} = \chi(T) \sqcap [\![pre_i]\!]^{\mathrm{p}}_{\mathcal{G}}(X_k)$. It can be shown by induction that $X_k(s)$ is equal to the maximal probability of staying in T for at least k steps that player i can achieve from $s \in S$. The detailed arguments can be found in [dAM01].

We note that on deterministic turn-based structures (and their special cases, such as transition systems), the boolean and probabilistic control problems are equivalent, as are the boolean and probabilistic μ-calculi. Indeed, for all deterministic turn-based game structures $\mathcal{G}$ with set of states S, and for all properties $\Phi \subseteq S^\omega$, all $i \in \{1, 2\}$, and all closed μ-calculus formulas ϕ containing only functions pre_1 and pre_2, we have that $\chi(\langle i \rangle^{\mathrm{b}}_{\mathcal{G}} \Phi) = \langle i \rangle^{\mathrm{p}}_{\mathcal{G}} \Phi$ and $\chi([\![\phi]\!]^{\mathrm{b}}_{\mathcal{G}}) = [\![\phi]\!]^{\mathrm{p}}_{\mathcal{G}}$.

Determinacy. As a consequence of the duality between the pre_1 and pre_2 operators, probabilistic control problems are determined, unlike their boolean counterparts: in particular, [Mar98] proves that for all games $\mathcal{G}$, all sets $\Phi \subseteq S^\omega_{\mathcal{G}}$ in the Borel hierarchy, and all $s \in S_{\mathcal{G}}$, we have $1 - \langle 1 \rangle^{\mathrm{p}}_{\mathcal{G}} \Phi(s) = \langle 2 \rangle^{\mathrm{p}}_{\mathcal{G}} (S^\omega \setminus \Phi)(s)$. While the proof of this result requires advanced arguments, the case in which Φ is a parity property follows elementarily from our μ-calculus solution formula (20), and from the duality of pre_1 and pre_2. In fact, consider a partition $\mathcal{A} = \langle T_1, T_2, \ldots, T_m \rangle$ of S. Letting $U_1 = \emptyset$ and $U_{i+1} = T_i$ for $1 \leq i \leq m$, we have:

$$\begin{aligned} 1 - \langle 1 \rangle^{\mathrm{p}}_{\mathcal{G}} \mathit{Parity}(\langle T_1, \ldots, T_m \rangle) &= 1 - [\![\gamma_m x_m \cdots \gamma_1 x_1 . \textstyle\bigvee_{j=1}^m (T_j \wedge pre_1(x_j))]\!]^{\mathrm{p}}_{\mathcal{G}} \\ &= [\![\gamma_{m+1} x_m \cdots \gamma_2 x_1 . \textstyle\bigvee_{j=1}^m (T_j \wedge pre_2(x_j))]\!]^{\mathrm{p}}_{\mathcal{G}} \\ &= \langle 2 \rangle^{\mathrm{p}}_{\mathcal{G}} \mathit{Parity}(\langle U_1, \ldots, U_{m+1} \rangle) \\ &= \langle 2 \rangle^{\mathrm{p}}_{\mathcal{G}} (S^\omega \setminus \mathit{Parity}(\langle T_1, \ldots, T_m \rangle)). \end{aligned}$$

4.2 The Linear Semantics of Probabilistic μ-Calculus

The solution (20) of parity control problems can be restated as follows. For player $i \in \{1, 2\}$, all game structures $\mathcal{G}$, and all EJ-form formulas ϕ containing only the function symbol pre_i, we have $[\![\phi]\!]^{\mathrm{p}}_{\mathcal{G}} = \langle i \rangle^{\mathrm{p}}_{\mathcal{G}} \mathit{PtyOf}(\phi)$. Using Lemma 1, we can therefore relate the linear and branching semantics of ϕ as follows.

Theorem 4. *For all game structures $\mathcal{G}$, all $i \in \{1, 2\}$, and all closed μ-calculus formulas $\phi \in \mathcal{PMC}_{S_{\mathcal{G}}}$ in EJ-form containing only the function symbol pre_i, we have that $[\![\phi]\!]^{\mathrm{p}}_{\mathcal{G}} = \langle i \rangle^{\mathrm{p}}_{\mathcal{G}} [\phi]^{\mathrm{blins}_{S_{\mathcal{G}}}}$.*

This theorem relates the branching semantics $[\![\phi]\!]^{\mathrm{p}}_{\mathcal{G}}$ of *probabilistic* μ-calculus with the linear semantics $[\phi]^{\mathrm{blins}_{S_{\mathcal{G}}}}$ of *boolean* μ-calculus. In order to relate branching and linear semantics of *probabilistic* μ-calculus, we define a probabilistic linear semantics $[\cdot]^{\mathrm{plins}_S}$ of probabilistic μ-calculus.

A trace $\sigma \in S^\omega$ gives rise to an interpretation $\mathcal{I}^p_\sigma = (\mathbb{L}(N \mapsto [0,1], \leq), [\![\cdot]\!]^{\mathrm{p}}_\sigma)$ for μ-calculus, where $(\mathbb{L}(N \mapsto [0,1], \leq)$ is the lattice of functions $\mathbb{N} \mapsto \{0, 1\}$ ordered pointwise, where a predicate $p \in 2^S$ is interpreted as its characteristic function, i.e., for all $k \geq 0$ we have $[\![p]\!]^{\mathrm{p}}_\sigma(k) = 1$ if $\sigma_k \in p$, and $[\![p]\!]^{\mathrm{p}}_\sigma(k) = 0$ if $\sigma_k \notin p$. Similarly to the boolean case, the definitions (16)–(17) can be simplified, since every state of the trace has a single successor. For all $X : \mathbb{N} \mapsto [0, 1]$ and $i \in \{1, 2\}$ we let $[\![pre_i]\!]^{\mathrm{p}}_\sigma(X) = \lambda k. X(k + 1)$. Given a closed μ-calculus formula

$\phi \in \mathcal{PMC}_S$, we define the *probabilistic linear semantics* $[\phi]^{\text{plin}_S} : S^\omega \mapsto [0,1]$ of ϕ over the set of states S by taking the value of ϕ over the first state of the trace: specifically, we let $[\phi]^{\text{plin}_S}(\sigma) = [\![\phi]\!]^{\mathcal{I}^p_\sigma}(0)$.

In the definitions (14), (15) of the probabilistic control and verification problems, the property Φ is a set of traces. To complete our connection with the probabilistic linear semantics $[\cdot]^{\text{plin}_S}$, we need to define a probabilistic version of these problems. Let $h : S^\omega \mapsto [0,1]$ be a function that is measurable in the probability space $(S^\omega, \Omega, \Pr_s^{\pi_1,\pi_2})$, for all $\pi_1 \in \Pi_1$ and $\pi_2 \in \Pi_2$. We define:

$$\langle 1\rangle^{\text{q}}_{\mathcal{G}} h = \lambda s \in S \,.\, \sup_{\pi_1 \in \Pi_1} \inf_{\pi_2 \in \Pi_2} \mathrm{E}_s^{\pi_1,\pi_2}(h)$$

$$\langle 2\rangle^{\text{q}}_{\mathcal{G}} h = \lambda s \in S \,.\, \sup_{\pi_2 \in \Pi_2} \inf_{\pi_1 \in \Pi_1} \mathrm{E}_s^{\pi_1,\pi_2}(h).$$

The relationship between the probabilistic linear and branching semantics is then expressed by the following theorem.

Theorem 5. *For all game structure $\mathcal{G}$, all $i \in \{1,2\}$, and all closed μ-calculus formulas $\phi \in \mathcal{PMC}_S$ in strongly deterministic form and containing only the function symbol pre$_i$, we have that $[\![\phi]\!]^{\text{p}}_{\mathcal{G}} = \langle i\rangle^{\text{q}}_{\mathcal{G}} [\phi]^{\text{plin}_{S_\mathcal{G}}}$.*

For player 1, the above theorem states that for all $s \in S_\mathcal{G}$,

$$[\![\phi]\!]^{\text{p}}_{\mathcal{G}}(s) = \sup_{\pi_1 \in \Pi_1} \inf_{\pi_2 \in \Pi_2} \mathrm{E}_s^{\pi_1,\pi_2}([\phi]^{\text{plin}_{S_\mathcal{G}}}). \tag{21}$$

This equation can be read as follows: the value of a control μ-calculus formula $[\![\phi]\!]^{\text{p}}_{\mathcal{G}}$ is equal to the maximal expectation that player 1 can guarantee for the same formula, evaluated on linear traces. The theorem relates not only the branching and the linear probabilistic semantics, but also a global optimization problem to a local one. In (21) the right-hand side represents a global optimization problem: player 1 is trying to maximize the value of the function $[\phi]^{\text{plin}_S}$ over traces, and player 2 is trying to oppose this. On the left-hand side, on the other hand, the optimization is local, being performed through the evaluation of the operator $[\![pre_1]\!]^{\text{p}}_{\mathcal{G}}$ at all states of $\mathcal{G}$.

5 Discounted Verification and Control

In the boolean and probabilistic settings, properties are specified as ω-regular languages, and algorithms are encoded as fixpoint expressions in the μ-calculus. The main theorems, such as Theorem 1 and Theorem 3, express the relationship between the properties and the μ-calculus fixpoints that solve the verification and control problems. The correspondence between the branching and linear semantics serves mainly to clarify the relationship between the local optimization that takes place in the branching semantics, and the global optimization that takes place in the linear semantics. In the discounted setting, on the other hand, we choose not have an independent notion of discounted property: rather, discounted properties are specified by the linear semantics of formulas of the discounted μ-calculus. The main results for the discounted setting concern thus the relationship between the linear semantics (used to express properties) and the branching semantics (which represents algorithms) of discounted μ-calculus, as well as the relationship between the discounted setting and the undiscounted one. As both properties and algorithms are defined in terms of the μ-calculus, we begin by introducing discounted μ-calculus.

5.1 Discounted μ-Calculus

Given a set S of states and a set Υ of *discount factors,* the set $\mathcal{DMC}_{S,\Upsilon}$ of discounted μ-calculus formulas consists of all the formulas defined over the set of predicates $\mathcal{P}_S = 2^S$ and the set of functions

$$\mathcal{F}_\Upsilon = \{\ \alpha pre_i,\ (1-\alpha) + \alpha pre_i\ \mid i \in \{1,2\}, \alpha \in \Upsilon\},$$

where $\text{Dual}(\alpha pre_1) = (1-\alpha) + \alpha pre_2$ and $\text{Dual}(\alpha pre_2) = (1-\alpha) + \alpha pre_1$. As in the probabilistic case, given a game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta\rangle$, we interpret these formulas over the lattice $\mathbb{L}(S \mapsto [0,1], \leq)$ of functions $S \mapsto [0,1]$, ordered pointwise. Again, we define negation by $\sim f = \lambda s \in S.1 - f(s)$. The interpretation of predicates and functions is parameterized by a *discount factor interpretation* $\eta : \Upsilon \mapsto [0,1]$, that assigns to each discount factor $\alpha \in \Upsilon$ its value $\eta(\alpha) \in [0,1]$. As in the probabilistic semantics, we interpret the predicates $p \in \mathcal{P}$ as their characteristic function, i.e., $[\![p]\!]^{\mathrm{d}}_{\mathcal{G},\eta} = \chi(p)$. For all $\eta \in (\Upsilon \mapsto [0,1])$ and all $i \in \{1,2\}$, we let:

$$[\![\alpha pre_i]\!]^{\mathrm{d}}_{\mathcal{G},\eta}(X) = \lambda s \in S.\eta(\alpha)[\![pre_i]\!]^{\mathrm{p}}_{\mathcal{G}}(X)(s) \tag{22}$$

$$[\![(1-\alpha) + \alpha pre_i]\!]^{\mathrm{d}}_{\mathcal{G},\eta}(X) = \lambda s \in S.(1-\eta(\alpha)) + \eta(\alpha)[\![pre_i]\!]^{\mathrm{p}}_{\mathcal{G}}(X)(s). \tag{23}$$

Thus, the discounted interpretation of αpre_i is equal to the probabilistic interpretation of pre_i, discounted by a factor α; the discounted interpretation of $(1-\alpha) + \alpha pre_i$ is equal to the probabilistic interpretation of pre_i, discounted by a factor of α, and with $1-\alpha$ added to it. We denote by $disc(\mathcal{G},\eta) = (\mathbb{L}(S \mapsto [0,1], \leq), [\![\cdot]\!]^{\mathrm{d}}_{\mathcal{G},\eta})$ the resulting semantics for the μ-calculus, and we write $[\![\cdot]\!]^{\mathrm{d}}_{\mathcal{G},\eta}$ for $[\![\cdot]\!]^{disc(\mathcal{G},\eta)}$, omitting $\mathcal{G}$ when clear from the context.

While (22) is the expected definition, (23) requires some justification. Consider a game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta\rangle$. First, notice that this definition ensures that pre_1 and $(1-\alpha) + \alpha pre_2$ are dual: in fact, for $s \in S$ we have

$$\begin{aligned}
1 - [\![\alpha pre_1]\!]^{\mathrm{d}}_{\eta}(X)(s) &= 1 - \eta(\alpha)[\![pre_1]\!]^{\mathrm{p}}(X)(s) \\
&= 1 - \eta(\alpha) + \eta(\alpha) - \eta(\alpha)[\![pre_1]\!]^{\mathrm{p}}(X)(s) \\
&= (1-\eta(\alpha)) + \eta(\alpha)\big[1 - [\![pre_1]\!]^{\mathrm{p}}(X)(s)\big] \\
&= (1-\eta(\alpha)) + \eta(\alpha)[\![pre_2]\!]^{\mathrm{p}}(\sim X)(s) \\
&= [\![(1-\alpha) + \alpha pre_2]\!]^{\mathrm{d}}_{\eta}(\sim X)(s).
\end{aligned}$$

The definitions (22) and (23) can also be justified by showing how the resulting predecessor operators can be used to solve discounted reachability and safety games in a way analogous to (18) and (19). Let $T \subseteq S$ be a set of target states, and fix a player $i \in \{1, 2\}$. Consider a *discounted reachability game,* in which player i gets the payoff $\eta(\alpha)^k$ when the target T is reached after k steps, and the payoff 0 if T is not reached. The maximum payoff that player i can guarantee is given by

$$[\![\mu x.(T \vee \alpha pre_i(x))]\!]^{\mathrm{d}}_{\eta}. \tag{24}$$

As an example, consider again the game MATCHBIT, along with the formula $[\![\mu x.(\{s_{goal}\} \vee \alpha pre_i(x))]\!]^{\mathrm{d}}_{\eta}$, and let $r = \eta(\alpha)$. Let $X_0 = \lambda s.0$ and, for $k \in \mathbb{N}$, let $X_{k+1} = \chi(\{s_{goal}\}) \sqcup [\![\alpha pre_1]\!]^{\mathrm{d}}_{\eta}(X_k)$. We can verify that $X_0(s_{try}) = 0$, $X_1(s_{try}) = r(\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1) = \frac{r}{2}$, $X_2(s_{try}) = r(\frac{1}{2} \cdot \frac{r}{2} + \frac{1}{2} \cdot 1) = \frac{r}{2} + \frac{r^2}{4}$, and $\lim_{k \to \infty} X_k(s_{try}) = r/(2-r) = [\![\mu x.(\{s_{goal}\} \vee \alpha pre_i(x))]\!]^{\mathrm{d}}_{\eta}(s_{try})$.

Consider now a *discounted safety game,* in which player i gets the payoff $1 - \eta(\alpha)^k$ if the game stays in T for k consecutive steps, and the payoff 1 if T is never left. The maximum payoff that player i can guarantee is given by

$$[\![\nu x.(T \wedge (1-\alpha) + \alpha pre_i(x))]\!]^{\mathrm{d}}_{\eta}. \tag{25}$$

Indeed, one can verify that for all $s \in S$, we have

$$1 - [\![\mu x.(T \vee \alpha pre_1(x))]\!]^{\mathrm{d}}_{\eta}(s) = [\![\nu x.(\neg T \wedge (1-\alpha) + \alpha pre_2(x))]\!]^{\mathrm{d}}_{\eta}(s) \tag{26}$$

indicating that the payoff player 1 can guarantee in a discounted T-reachability game is equal to 1 minus the payoff that player 2 can guarantee for the discounted $\neg T$-safety game.

Above, we have informally introduced discounted reachability and safety games in terms of payoffs associated with the traces. How are these payoffs defined, for more general goals? And what is the precise definition of the games that (24) and (25) solve? To answer these questions, we introduce the linear semantics of discounted μ-calculus, and we once more relate the linear semantics to the branching one.

5.2 The Linear Semantics of Discounted μ-Calculus

A *discounted property* is the interpretation of a discounted μ-calculus formula over linear traces. Similarly to the probabilistic case, the linear semantics of discounted μ-calculus associates with each trace a number in the interval $[0, 1]$ obtained by evaluating the μ-calculus formula over the trace, and taking the value at the initial state of the trace.

Consider a set Υ of discount factors, along with a discount factor interpretation $\eta : \Upsilon \mapsto [0, 1]$. A trace $\sigma \in S^\omega$ gives rise to an interpretation $\mathcal{I}^{\eta}_{\sigma} = (\mathbb{L}(N \mapsto [0,1], \leq), [\![\cdot]\!]^{\mathrm{d}}_{\sigma,\eta})$ for the discounted μ-calculus formulas in $\mathcal{DMC}_{S,\Upsilon}$. As in the probabilistic case, all predicates $p \in 2^S$ are interpreted as their characteristic function, i.e., for all $k \geq 0$ we have $[\![p]\!]^{\mathrm{p}}_{\sigma}(k) = 1$ if $\sigma_k \in p$, and $[\![p]\!]^{\mathrm{p}}_{\sigma}(k) = 0$ if

$\sigma_k \notin p$. The definitions (22) and (23) can be simplified, since in a trace, every location has a single successor. For all $X : \mathbb{N} \mapsto [0,1]$ and $i \in \{1,2\}$ we let

$$\llbracket \alpha pre_i \rrbracket^{\mathrm{d}}_{\sigma,\eta}(X) = \lambda k.\big[\eta(\alpha)\, X(k+1)\big]$$
$$\llbracket (1-\alpha) + \alpha pre_i \rrbracket^{\mathrm{d}}_{\sigma,\eta}(X) = \lambda k.\big[(1-\eta(\alpha)) + \eta(\alpha)\, X(k+1)\big].$$

Given a closed μ-calculus formula $\phi \in \mathcal{DMC}_{S,\Upsilon}$, we then define its *discounted linear semantics* $[\phi]^{\mathrm{dlin}_{S,\eta}} : S^\omega \mapsto [0,1]$ by $[\phi]^{\mathrm{dlin}_{S,\eta}}(\sigma) = \llbracket \phi \rrbracket^{\mathcal{I}^\eta_\sigma}(0)$. A *discounted property* is the mapping $S^\omega \mapsto [0,1]$ defined by the linear semantics $[\phi]^{\mathrm{dlin}_{S,\eta}}$ of a closed discounted μ-calculus formula $\phi \in \mathcal{DMC}_{S,\Upsilon}$.

As an example, consider again a subset $T \subseteq S$ of target states, and a player $i \in \{1,2\}$. The payoff of the discounted reachability game considered informally in Sect. 5.1 can be defined by $[\mu x.(T \vee \alpha pre_i(x))]^{\mathrm{dlin}_{S,\eta}}$: indeed,

$$[\mu x.(T \vee \alpha pre_i(x))]^{\mathrm{dlin}_{S,\eta}}(\sigma) = \eta(\alpha)^k,$$

where $k = \min\{j \in \mathbb{N} \mid \sigma_j \in T\}$. The fact that (24) represents the maximum payoff that player 1 can achieve in a game structure $\mathcal{G}$ can be formalized as

$$\langle 1 \rangle^{\mathrm{q}}_{\mathcal{G}} [\mu x.(T \vee \alpha pre_i(x))]^{\mathrm{dlin}_{S,\eta}} = \llbracket \mu x.(T \vee \alpha pre_i(x)) \rrbracket^{\mathrm{d}}_{\mathcal{G},\eta}. \tag{27}$$

Similarly, the payoff of the discounted safety game considered informally in Sect. 5.1 can be defined by $[\nu x.(T \wedge (1-\alpha) + \alpha pre_i(x))]^{\mathrm{dlin}_{S,\eta}}$: indeed,

$$[\nu x.(T \wedge (1-\alpha) + \alpha pre_i(x))]^{\mathrm{dlin}_{S,\eta}}(\sigma) = 1 - \eta(\alpha)^k,$$

where $k = \min\{j \in \mathbb{N} \mid \sigma_j \notin T\}$. Also in this case, for all game structures $\mathcal{G}$ we have:

$$\langle 1 \rangle^{\mathrm{q}}_{\mathcal{G}} [\nu x.(T \wedge (1-\alpha) + \alpha pre_i(x))]^{\mathrm{dlin}_{S,\eta}} = \llbracket \nu x.(T \wedge (1-\alpha) + \alpha pre_i(x)) \rrbracket^{\mathrm{d}}_{\mathcal{G},\eta}. \tag{28}$$

The relations (27) and (28) are just two special cases of the general relation between the linear and branching semantics of discounted μ-calculus, expressed by the following theorem.

Theorem 6. *For all game structures $\mathcal{G}$, all players $i \in \{1,2\}$, all sets Υ of discount factors, all discount factor evaluations $\eta \in (\Upsilon \mapsto [0,1])$, and all closed μ-calculus formulas $\phi \in \mathcal{DMC}_{S,\Upsilon}$ in strongly deterministic form that contain only the function symbols αpre_i and $(1-\alpha) + \alpha pre_i$ for $\alpha \in \Upsilon$, we have that $\llbracket \phi \rrbracket^{\mathrm{d}}_{\mathcal{G},\eta} = \langle i \rangle^{\mathrm{q}}_{\mathcal{G}} [\phi]^{\mathrm{dlin}_{S,\eta}}$.*

This theorem is the main result about the verification of discounted properties, as it relates a discounted property $[\phi]^{\mathrm{dlin}_{S,\eta}}$ to the valuation $\llbracket \phi \rrbracket^{\mathrm{d}}_{\mathcal{G},\eta}$ computed by the verification algorithm ϕ over the game structure $\mathcal{G}$.

Determinacy. Since discounted properties are defines as the linear semantics of discounted μ-calculus formulas, the duality of discounted control problems can be stated as follows.

Theorem 7. *For all game structures $\mathcal{G}$, all sets Υ of discount factors, all discount factor evaluations $\eta \in (\Upsilon \mapsto [0,1])$, and all closed μ-calculus formulas $\phi \in \mathcal{DMC}_{S_\mathcal{G},\Upsilon}$ in strongly deterministic form that contain only the function symbols αpre_1 and $(1-\alpha)+\alpha pre_1$ for $\alpha \in \Upsilon$, we have that*

$$1 - \langle 1 \rangle^{\mathrm{q}}_{\mathcal{G}} [\phi]^{\mathrm{dlin}_{S_\mathcal{G},\eta}} = \langle 2 \rangle^{\mathcal{G}}_{q} [\neg\phi]^{\mathrm{dlin}_{S_\mathcal{G},\eta}}.$$

5.3 Relation between Discounted and Probabilistic μ-Calculus

Given $r \in [0,1]$, denote by $E_r(\Upsilon) : \Upsilon \mapsto [0,r]$ the set of all discount factor interpretations bound by r. If $\eta \in E_r(\Upsilon)$ for $r < 1$, we say that η is *contractive.* A fixpoint quantifier μx or νx occurs *guarded* in a formula ϕ if a function symbol *pre* occurs on every syntactic path from the quantifier to a quantified occurrence of the variable x. For example, in the formula $\mu x.(T \vee \alpha pre_i(x))$ the fixpoint quantifier occurs guarded; in the formula $(1-\alpha)+\alpha pre_i(\mu x.(T \vee x))$ it does not. Under a contractive discount factor interpretation, every guarded occurrence of a fixpoint quantifier defines a contractive operator on the values of the free variables that are in the scope of the quantifier. Hence, by the Banach fixpoint theorem, the fixpoint is unique. In such cases, we need not distinguish between μ and ν quantifiers, and we denote both by κ.

If $\eta(\alpha) = 1$, then both $[\![\alpha pre_i]\!]^{\mathrm{d}}_{\mathcal{G},\eta}$ and $[\![(1-\alpha)+\alpha pre_i]\!]^{\mathrm{d}}_{\mathcal{G},\eta}$ reduce to the undiscounted function $[\![pre_i]\!]^{\mathrm{p}}_{\mathcal{G}}$, for $i \in \{1,2\}$. The following theorem extends this observation to the complete μ-calculus, showing how the semantics of the discounted μ-calculus converges to the semantics of the undiscounted μ-calculus as the discount factors approach 1. To state the result, we extend the semantics of discounted μ-calculus to interpret also the functions pre_1, pre_2, letting $[\![pre_i]\!]^{\mathrm{d}}_{\mathcal{G},\eta} = [\![pre_i]\!]^{\mathrm{p}}_{\mathcal{G}}$ for all $i \in \{1,2\}$, game structures $\mathcal{G}$, and all discount interpretations η. We also let $\eta[\alpha := a]$ be the discount factor interpretation defined by $\eta[\alpha := a](\alpha) = a$ and $\eta[\alpha := a](\alpha') = \eta(\alpha')$ for $\alpha \neq \alpha'$.

Theorem 8. [dAHM03] *For all game structures $\mathcal{G}$, Let $\phi(x) \in \mathcal{DMC}_{S_\mathcal{G},\Upsilon}$ be a μ-calculus formula with free variable x, and discount factor α. The following assertions hold:*

1. *If x and α always and only occur in the context $\alpha pre_i(x)$, for $i \in \{1,2\}$, then*

$$\lim_{a \to 1} [\![\lambda x.\phi(\alpha pre_i(x))]\!]^{\mathrm{d}}_{\mathcal{G},\eta[\alpha:=a],e} = [\![\mu x.\phi(pre_i(x))]\!]^{\mathrm{d}}_{\mathcal{G},\eta,e}.$$

2. *If x and α always and only occur in the context $(1-\alpha)+\alpha pre_i(x)$, then*

$$\lim_{a \to 1} [\![\lambda x.\phi((1-\alpha)+\alpha pre_i(x))]\!]^{\mathrm{d}}_{\mathcal{G},\eta[\alpha:=a],e} = [\![\nu x.\phi(pre_i(x))]\!]^{\mathrm{d}}_{\mathcal{G},\eta,e}.$$

The remarkable fact is that the order of quantifiers in probabilistic μ-calculus corresponds to the order in which the limits are taken in discounted μ-calculus. For instance, for a game structure $\mathcal{G}$ and $T \subseteq S_{\mathcal{G}}$, let $\phi = \lambda y.\lambda x.((\neg T \wedge \alpha pre_i(x)) \vee (T \wedge (1-\beta) + \beta pre_i(y)))$. We have that

$$\lim_{a\to 1}\lim_{b\to 1} [\![\phi]\!]^{\mathrm{d}}_{\mathcal{G},\eta[\alpha:=a,\beta:=b]} = [\![\mu x.\nu y.((\neg T \wedge pre_i(x)) \vee (T \wedge pre_i(y)))]\!]^{\mathrm{p}}_{\mathcal{G}} \quad (29)$$

$$\lim_{b\to 1}\lim_{a\to 1} [\![\phi]\!]^{\mathrm{d}}_{\mathcal{G},\eta[\alpha:=a,\beta:=b]} = [\![\nu y.\mu x.((\neg T \wedge pre_i(x)) \vee (T \wedge pre_i(y)))]\!]^{\mathrm{p}}_{\mathcal{G}}. \quad (30)$$

Formula (29) is the solution of probabilistic co-Büchi games with goal $\Diamond\Box T$, while (30) is the solution of probabilistic Büchi games with goal $\Box\Diamond T$.

6 Equivalence Metrics

To complete the extension of the classical boolean framework for specification, verification, and control to the quantitative case, we show how the classical notion of bisimulation can be extended to the quantitative setting, and how our quantitative μ-calculi characterize quantitative bisimulation, just as the boolean μ-calculus, like CTL, characterizes bisimulation.

6.1 Alternating Bisimulation

In the boolean setting, and for deterministic game structures, the notion of bisimulation for games is called *alternating simulation* [AHKV98]. Fix a deterministic game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta\rangle$, along with a set $\mathcal{P} \subseteq 2^S$ of predicates. A relation $R \subseteq S \times S$ is a *player-1 alternating bisimulation* if, for all $s, t \in S$, $(s,t) \in R$ implies that $s \in p \leftrightarrow t \in p$ for all $p \in \mathcal{P}$, and if $(s,t) \in R$, then

$$\forall a_1 \in \Gamma_1(s).\exists b_1 \in \Gamma_1(t).\forall b_2 \in \Gamma_2(t).\exists a_2 \in \Gamma_2(s).\hat{R}(\tau(s,a_1,a_2), \tau(s,b_1,b_2)),$$

$$\forall b_1 \in \Gamma_1(t).\exists a_1 \in \Gamma_1(s).\forall a_2 \in \Gamma_2(s).\exists b_2 \in \Gamma_2(t).\hat{R}(\tau(s,a_1,a_2), \tau(s,b_1,b_2)),$$

where $\hat{R}(\{t_1\},\{t_2\})$ iff $(t_1,t_2) \in R$, for all $t_1, t_2 \in S$. The definition of a *player-2 alternating bisimulation* is obtained by exchanging in the above definition the roles of players 1 and 2. A relation R is an *alternating bisimulation* if it is both a player-1 and a player-2 alternating bisimulation.

To obtain the coarsest player 1 alternating bisimulation, i.e., the largest relation that is a player-1 alternating bisimulation, we can use a symbolic fixpoint approach [Mil90], which in view of our extension to the quantitative case, we state as follows. A *binary distance function* is a function $d : S \times S \mapsto \{0,1\}$ that maps each pair of states $s, t \in S$ to their distance $d(s,t) \in \{0,1\}$, and such that for all $s, t, u \in S$, we have $d(s,t) = d(t,s)$ and $d(s,t) \leq d(s,u) + d(u,t)$. For distance functions d, d' we let $d \leq d'$ iff $d(s,t) \leq d'(s,t)$ for all $s, t \in S$. We define the functor F_1 mapping binary distance functions to binary distance functions: for all binary distance functions d and all $s, t \in S$, we let $F_1(d)(s,t) = 1$ if there is $p \in \mathcal{P}$ such that $s \in p \not\leftrightarrow t \in p$, and we let

$$F_1(d)(s,t) = \max \left\{ \begin{array}{l} \max\limits_{a_1 \in \Gamma_1(s)} \min\limits_{b_1 \in \Gamma_1(t)} \max\limits_{b_2 \in \Gamma_2(t)} \min\limits_{a_2 \in \Gamma_2(s)} \hat{d}(\tau(s,a_1,a_2), \tau(s,b_1,b_2)), \\ \max\limits_{b_1 \in \Gamma_1(t)} \min\limits_{a_1 \in \Gamma_1(s)} \max\limits_{a_2 \in \Gamma_2(s)} \min\limits_{b_2 \in \Gamma_2(t)} \hat{d}(\tau(s,a_1,a_2), \tau(s,b_1,b_2)) \end{array} \right\}$$

otherwise, where $\hat{d}(\{t_1\},\{t_2\}) = d(t_1,t_2)$ for all $t_1,t_2 \in S$. A player-1 alternating bisimulation R is simply a relation whose characteristic function is a fixpoint of F_1, i.e., it is a subset $R \subseteq S \times S$ such that $\chi(R) = F_1(\chi(R))$, where $\chi(R)(s,t)$ is 0 if $(s,t) \in R$, and is 1 otherwise. In particular, the coarsest player-1 bisimulation B_1^{bin} is given by $d_1^* = \chi(B_1^{\mathrm{bin}})$, where d_1^* is least fixpoint of the functor F_1, i.e., the least distance function that satisfies $d_1^* = F_1(d_1^*)$. We define the coarsest player-2 alternating bisimulation B_2^{bin} in an analogous fashion, with respect to a functor F_2 obtained by swapping the roles of players 1 and 2 in the definition of F_1. Finally, the coarsest alternating bisimulation B^{bin} is given by $\chi(B^{\mathrm{bin}}) = d^*$, where d^* is the least distance function that satisfies both $d^* = F_1(d^*)$ and $d^* = F_2(d^*)$. When we wish to make explicit the dependence of the bisimulation relations on the game and on $\mathcal{P}$, we write $B_{\mathcal{G},\mathcal{P}}^{\mathrm{bin}}$, $B_{1,\mathcal{G},\mathcal{P}}^{\mathrm{bin}}$, and $B_{2,\mathcal{G},\mathcal{P}}^{\mathrm{bin}}$ for B^{bin}, B_1^{bin}, and B_2^{bin}. The following theorem, derived from [AHKV98], relates alternating bisimulation and boolean μ-calculus.

Theorem 9. *For a deterministic game structure $\mathcal{G}$, the following assertions hold:*

1. *For all $i \in \{1,2\}$, we have that $(s,t) \not\in B_{i,\mathcal{G},\mathcal{P}}^{\mathrm{bin}}$ iff there is a closed μ-calculus formula $\phi \in \mathcal{BMC}_{S_\mathcal{G}}$ containing only predicates in $\mathcal{P}$ and functions in $\{pre_i, dpre_i\}$ such that $s \in [\![\phi]\!]_\mathcal{G}^{\mathrm{b}}$ and $t \not\in [\![\phi]\!]_\mathcal{G}^{\mathrm{b}}$.*
2. *$(s,t) \not\in B_{\mathcal{G},\mathcal{P}}^{\mathrm{bin}}$ iff there is a closed μ-calculus formula $\phi \in \mathcal{BMC}_{S_\mathcal{G}}$ containing only predicates in $\mathcal{P}$ such that $s \in [\![\phi]\!]_\mathcal{G}^{\mathrm{b}}$ and $t \not\in [\![\phi]\!]_\mathcal{G}^{\mathrm{b}}$.*

6.2 Game Bisimulation Distance

To obtain a quantitative version of alternating bisimulation, we adapt the definition of F_i to the case of probabilistic game structures and quantitative distance functions [dAHM03]. Fix a game structure $\mathcal{G} = \langle S, \mathcal{M}, \Gamma_1, \Gamma_2, \delta \rangle$, along with a set $\mathcal{P} \subseteq 2^S$ of predicates. A *distance function* is a mapping $d : S \times S \mapsto [0,1]$ such that for all $s,t,u \in S$ we have $d(s,t) = d(t,s)$ and $d(s,t) \leq d(s,u) + d(u,t)$. We define *discounted game bisimulation* [dAHM03] with respect to a discount factor $r \in [0,1]$; the undiscounted case corresponds to $r = 1$. Given $r \in [0,1]$, we define the functor G_r mapping distance functions to distance functions: for every distance function d and all states $s,t \in S$, we define $G_r(d)(s,t) = 1$ if there is $p \in \mathcal{P}$ such that $s \in p \not\leftrightarrow t \in p$, and

$$G_r(d)(s,t) = r \cdot \max \left\{ \begin{array}{l} \sup\limits_{\zeta_1 \in \mathcal{D}_1(s)} \inf\limits_{\xi_1 \in \mathcal{D}_1(t)} \sup\limits_{\xi_2 \in \mathcal{D}_2(t)} \inf\limits_{\zeta_2 \in \mathcal{D}_2(s)} D(d)(\hat{\delta}(s,\zeta_1,\zeta_2), \hat{\delta}(t,\xi_1,\xi_2)), \\ \sup\limits_{\xi_1 \in \mathcal{D}_1(t)} \inf\limits_{\zeta_1 \in \mathcal{D}_1(s)} \sup\limits_{\zeta_2 \in \mathcal{D}_2(s)} \inf\limits_{\xi_2 \in \mathcal{D}_2(t)} D(d)(\hat{\delta}(s,\zeta_1,\zeta_2), \hat{\delta}(t,\xi_1,\xi_2)) \end{array} \right\}$$

otherwise. For a distance function d and distributions ζ_1 and ζ_2, we let $D(d)(\zeta_1, \zeta_2)$ be the extension of the function d from states to distributions [vBW01a] given by the solution to the linear program $\max \sum_{s \in Q} (\zeta_1(s) - \zeta_2(s)) k_s$ where the variables $\{k_s\}_{s \in Q}$ are subject to $k_s - k_t \leq d(s,t)$ for all $s, t \in Q$. The least distance function that is a fixpoint of G_r is called *r-discounted game bisimilarity,* and denoted B_r^{disc}. On MDPs (one-player game structures), for $r < 1$, discounted game bisimulation coincides with the discounted distance metrics of [vBW01a]. Again, we write $B_{r,\mathcal{G},\mathcal{P}}^{\mathrm{disc}}$ when we wish to make explicit the dependency of B_r^{disc} from the game $\mathcal{G}$ and from the subset of predicates $\mathcal{P}$.

By the minimax theorem [vN28], we can exchange the two middle sup and inf operators in the definition of G_r; as a consequence, it is easy to see that the definition is symmetrical with respect to players 1 and 2. Thus, there is only one version of (un)discounted game bisimulation, in contrast to the two distinct player-1 and player-2 alternating bisimulations. Indeed, comparing the definition of F_i and G_r, we see that alternating bisimulation is defined with respect to *deterministic* move distributions, and the minimax theorem does not hold if the players are forced to use deterministic distributions. The following theorem relates game bisimilarity with quantitative and discounted μ-calculus.

Theorem 10. [dAHM03] *The following assertions hold for all game structures $\mathcal{G}$.*

1. *Let $\mathcal{PMC}_{S_\mathcal{G},\mathcal{P}}$ be the set of closed μ-calculus formulas in $\mathcal{PMC}_{S_\mathcal{G}}$ that contain only predicates in $\mathcal{P}$. For all $s, t \in S_\mathcal{G}$ we have*

$$B_{1,\mathcal{G},\mathcal{P}}^{\mathrm{disc}}(s,t) = \sup_{\phi \in \mathcal{PMC}_{S_\mathcal{G},\mathcal{P}}} \left| [\![\phi]\!]_\mathcal{G}^{\mathrm{p}}(s) - [\![\phi]\!]_\mathcal{G}^{\mathrm{p}}(t) \right|.$$

2. *Let $\mathcal{DMC}_{S_\mathcal{G},\Upsilon,\mathcal{P}}$ be the set of closed μ-calculus formulas in $\mathcal{DMC}_{S_\mathcal{G},\Upsilon}$ that contain only predicates in $\mathcal{P}$. For all $s, t \in S_\mathcal{G}$ and all $r \in [0,1]$, we have*

$$B_{r,\mathcal{G},\mathcal{P}}^{\mathrm{disc}}(s,t) = \sup_{\phi \in \mathcal{DMC}_{S_\mathcal{G},\mathcal{P}}} \sup_{\eta \in E_r(\Upsilon)} \left| [\![\phi]\!]_{\mathcal{G},\eta}^{\mathrm{d}}(s) - [\![\phi]\!]_{\mathcal{G},\eta}^{\mathrm{d}}(t) \right|.$$

It is possible to extend the connection between discounted μ-calculus and equivalence relations further, including results about the stability of bisimulation and discounted μ-calculus with respect to perturbations in the game structure [DGJP02,dAHM03].

References

[AHK02] R. Alur, T.A. Henzinger, and O. Kupferman. Alternating time temporal logic. *J. ACM*, 49:672–713, 2002.

[AHKV98] R. Alur, T.A. Henzinger, O. Kupferman, and M.Y. Vardi. Alternating refinement relations. In *CONCUR 97: Concurrency Theory. 8th Int. Conf.*, volume 1466 of *Lect. Notes in Comp. Sci.*, pages 163–178. Springer-Verlag, 1998.

[BC96] G. Bhat and R. Cleaveland. Efficient model checking via the equational μ-calculus. In *Proc. 11th IEEE Symp. Logic in Comp. Sci.*, pages 304–312, 1996.

[Ber95] D.P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 1995. Volumes I and II.

[BL69] J.R. Büchi and L.H. Landweber. Solving sequential conditions by finite-state strategies. *Trans. Amer. Math. Soc.*, 138:295–311, 1969.

[Bry86] R.E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, 1986.

[Con92] A. Condon. The complexity of stochastic games. *Information and Computation*, 96:203–224, 1992.

[dAH00] L. de Alfaro and T.A. Henzinger. Concurrent omega-regular games. In *Proc. 15th IEEE Symp. Logic in Comp. Sci.*, pages 141–154, 2000.

[dAHK98] L. de Alfaro, T.A. Henzinger, and O. Kupferman. Concurrent reachability games. In *Proc. 39th IEEE Symp. Found. of Comp. Sci.*, pages 564–575. IEEE Computer Society Press, 1998.

[dAHM01a] L. de Alfaro, T.A. Henzinger, and R. Majumdar. From verification to control: Dynamic programs for omega-regular objectives. In *Proc. 16th IEEE Symp. Logic in Comp. Sci.*, pages 279–290. IEEE Press, 2001.

[dAHM01b] L. de Alfaro, T.A. Henzinger, and R. Majumdar. Symbolic algorithms for infinite-state games. In *CONCUR 01: Concurrency Theory. 12th Int. Conf.*, volume 2154 of *Lect. Notes in Comp. Sci.*, pages 536–550. Springer-Verlag, 2001.

[dAHM03] L. de Alfaro, T.A. Henzinger, and R. Majumdar. Discounting the future in systems theory. In *Proc. 30th Int. Colloq. Aut. Lang. Prog.*, Lect. Notes in Comp. Sci. Springer-Verlag, 2003.

[dAM01] L. de Alfaro and R. Majumdar. Quantitative solution of omega-regular games. In *Proc. 33rd ACM Symp. Theory of Comp.*, pages 675–683. ACM Press, 2001.

[DEP02] J. Desharnais, A. Edalat, and P. Panangaden. Bisimulation for labelled markov processes. *Information and Computation*, 179(2):163–193, 2002.

[Der70] C. Derman. *Finite State Markovian Decision Processes*. Academic Press, 1970.

[DGJP99] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. Metrics for labelled markov systems. In *CONCUR'99: Concurrency Theory. 10th Int. Conf.*, volume 1664 of *Lect. Notes in Comp. Sci.*, pages 258–273. Springer, 1999.

[DGJP02] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. The metric analogue of weak bisimulation for probabilistic processes. In *Proc. 17th IEEE Symp. Logic in Comp. Sci.*, pages 413–422, 2002.

[EJ91] E.A. Emerson and C.S. Jutla. Tree automata, mu-calculus and determinacy (extended abstract). In *Proc. 32nd IEEE Symp. Found. of Comp. Sci.*, pages 368–377. IEEE Computer Society Press, 1991.

[EL86] E.A. Emerson and C.L. Lei. Efficient model checking in fragments of the propositional μ-calculus. In *Proc. First IEEE Symp. Logic in Comp. Sci.*, pages 267–278, 1986.

[Eve57] H. Everett. Recursive games. In *Contributions to the Theory of Games III*, volume 39 of *Annals of Mathematical Studies*, pages 47–78, 1957.

[Fel83] Y.A. Feldman. A decidable propositional probabilistic dynamic logic. In *Proc. 15th ACM Symp. Theory of Comp.*, pages 298–309, 1983.

[FH82] Y.A. Feldman and D. Harel. A probabilistic dynamic logic. In *Proc. 14th ACM Symp. Theory of Comp.*, pages 181–195, 1982.

[FV97] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer-Verlag, 1997.

[HK97] M. Huth and M. Kwiatkowska. Quantitative analysis and model checking. In *Proc. 12th IEEE Symp. Logic in Comp. Sci.*, pages 111–122, 1997.

[HM00] T.A. Henzinger and R. Majumdar. A classification of symbolic transition systems. In *Proc. of 17th Annual Symp. on Theor. Asp. of Comp. Sci.*, volume 1770 of *Lect. Notes in Comp. Sci.*, pages 13–34. Springer-Verlag, 2000.

[Koz83a] D. Kozen. A probabilistic PDL. In *Proc. 15th ACM Symp. Theory of Comp.*, pages 291–297, 1983.

[Koz83b] D. Kozen. Results on the propositional μ-calculus. *Theoretical Computer Science*, 27(3):333–354, 1983.

[KSK66] J.G. Kemeny, J.L. Snell, and A.W. Knapp. *Denumerable Markov Chains*. D. Van Nostrand Company, 1966.

[Mar98] D.A. Martin. The determinacy of Blackwell games. *The Journal of Symbolic Logic*, 63(4):1565–1581, 1998.

[Mil90] R. Milner. Operational and algebraic semantics of concurrent processes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 1202–1242. Elsevier Science Publishers (North-Holland), Amsterdam, 1990.

[MP91] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems: Specification*. Springer-Verlag, New York, 1991.

[Rei80] J.H. Reif. Logic for probabilistic programming. In *Proc. 12th ACM Symp. Theory of Comp.*, pages 8–13, 1980.

[Ros90] K.I. Rosenthal. *Quantales and Their Applications*, volume 234 of *Pitman Research Notes in Mathematics Series*. Longman Scientific & Technical, Harlow, 1990.

[Sha53] L.S. Shapley. Stochastic games. *Proc. Nat. Acad. Sci. USA*, 39:1095–1100, 1953.

[Tho90] W. Thomas. Automata on infinite objects. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, chapter 4, pages 135–191. Elsevier Science Publishers (North-Holland), Amsterdam, 1990.

[Tho95] W. Thomas. On the synthesis of strategies in infinite games. In *Proc. of 12th Annual Symp. on Theor. Asp. of Comp. Sci.*, volume 900 of *Lect. Notes in Comp. Sci.*, pages 1–13. Springer-Verlag, 1995.

[vBW01a] F. van Breugel and J. Worrel. An algorithm for quantitative verification of probabilistic transition systems. In *CONCUR 01: Concurrency Theory. 12th Int. Conf.*, volume 2154 of *Lect. Notes in Comp. Sci.*, pages 336–350, 2001.

[vBW01b] F. van Breugel and J. Worrel. Towards quantitative verification of probabilistic systems. In *Proc. 28th Int. Colloq. Aut. Lang. Prog.*, volume 2076 of *Lect. Notes in Comp. Sci.*, pages 421–432. Springer-Verlag, 2001.

[vN28] J. von Neumann. Zur Theorie der Gesellschaftsspiele. *Math. Annal*, 100:295–320, 1928.