Probabilistic Weighted Automata*

Krishnendu Chatterjee[1], Laurent Doyen[2**], and Thomas A. Henzinger[3]

[1] Institute of Science and Technology (IST), Austria
[2] Université Libre de Bruxelles (ULB), Belgium
[3] École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Abstract. Nondeterministic weighted automata are finite automata with numerical weights on transitions. They define quantitative languages L that assign to each word w a real number $L(w)$. The value of an infinite word w is computed as the maximal value of all runs over w, and the value of a run as the supremum, limsup, liminf, limit average, or discounted sum of the transition weights.

We introduce probabilistic weighted automata, in which the transitions are chosen in a randomized (rather than nondeterministic) fashion. Under almost-sure semantics (resp. positive semantics), the value of a word w is the largest real v such that the runs over w have value at least v with probability 1 (resp. positive probability).

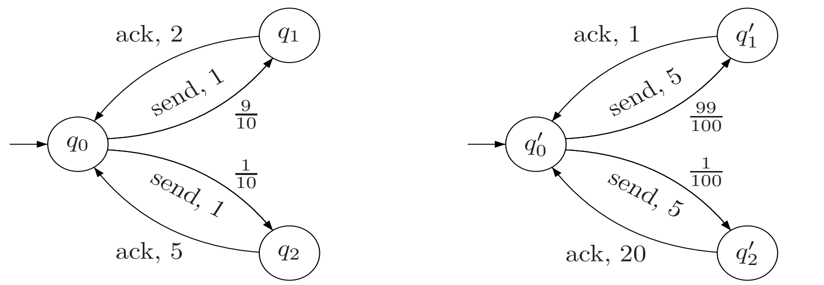
We study the classical questions of automata theory for probabilistic weighted automata: emptiness and universality, expressiveness, and closure under various operations on languages. For quantitative languages, emptiness and universality are defined as whether the value of some (resp. every) word exceeds a given threshold. We prove some of these questions to be decidable, and others undecidable. Regarding expressive power, we show that probabilities allow us to define a wide variety of new classes of quantitative languages, except for discounted-sum automata, where probabilistic choice is no more expressive than nondeterminism. Finally, we give an almost complete picture of the closure of various classes of probabilistic weighted automata for the following pointwise operations on quantitative languages: max, min, sum, and numerical complement.

1 Introduction

In formal design, specifications describe the set of correct behaviors of a system. An implementation satisfies a specification if all its behaviors are correct. If we view a behavior as a word, then a specification is a language, i.e., a set of words. Languages can be specified using finite automata, for which a large number of results and techniques are known; see [20, 24]. We call them *boolean languages*

* This research was supported in part by the Swiss National Science Foundation under the Indo-Swiss Joint Research Programme, by the European Network of Excellence on Embedded Systems Design (ArtistDesign), by the European projects Combest, Quasimodo, and Gasics, by the PAI program Moves funded by the Belgian Federal Government, and by the CFV (Federated Center in Verification) funded by the F.R.S.-FNRS.

** Postdoctoral researcher of the F.R.S.-FNRS.



(a) Low reliability but cheap. (b) High reliability but expensive.

Fig. 1. Two specifications of a channel.

because a given behavior is either good or bad according to the specification. Boolean languages are useful to specify functional requirements.

In a generalization of this approach, we consider *quantitative languages* L, where each word w is assigned a real number $L(w)$. The value of a word can be interpreted as the amount of some resource (e.g., memory or power) needed to produce it, or as a quality measurement for the corresponding behavior [5, 6]. Therefore, quantitative languages are useful to specify nonfunctional requirements such as resource constraints, reliability properties, or levels of quality (such as quality of service). Note that a boolean language L is a special case of quantitative language that assigns value 1 to the words in L and value 0 to the words not in L.

Quantitative languages can be defined using nondeterministic weighted automata, i.e., finite automata with numerical weights on transitions [13, 17]. In [7], we studied quantitative languages of infinite words and defined the value of an infinite word w as the maximal value of all runs of an automaton over w (if the automaton is nondeterministic, then there may be many runs over w). The value of a run r is a function of the infinite sequence of weights that appear along r. There are several natural functions to consider, such as Sup, LimSup, LimInf, limit average, and discounted sum of weights. For example, peak power consumption can be modeled as the maximum of a sequence of weights representing power usage; energy use, as a discounted sum; average response time, as a limit average [4, 5].

In this paper, we present *probabilistic* weighted automata as a new model defining quantitative languages. In such automata, nondeterministic choice is replaced by probability distributions on successor states. The value of an infinite word w is defined to be the maximal value v such that the set of runs over w with value at least v has either positive probability (*positive semantics*), or probability 1 (*almost-sure semantics*). This simple definition combines in a general model the natural quantitative extensions of logics and automata [14, 15, 7], and the probabilistic models of automata for which boolean properties have been studied [22, 3, 2]. Note that the probabilistic Büchi and coBüchi automata of [2] are a special case of probabilistic weighted automata with weights 0 and 1 only (and the value of an infinite run computed as LimSup or LimInf, respectively). While quantitative objectives are standard in the branching-time context of stochastic games [23, 16, 18, 5, 11, 19], we are not aware of any model combining probabilities and weights in the linear-time context of words and languages, though such a model is very natural for the specification of quantitative properties. Consider the specification of two types of communication channels given in Fig. 1. One has low cost (sending costs 1 unit) and low reliability (a failure occurs in 10% of the cases and entails an increased cost for the operation), while the second is expensive (sending costs 5 units), but the reliability is high (though the cost of a failure is prohibitive). In the figure, we omit the self-loops with cost 0 in state q_0 and q_0' over *ack*, and in q_1, q_2, q_1', q_2' over *send*. Natural questions can be formulated in this framework, such as whether the average cost of every word $w \in \{send, ack\}^\omega$ is really smaller in the low-cost channel, or to construct a probabilistic weighted automaton that assigns to each infinite word $w \in \{send, ack\}^\omega$ the minimum of the average cost of the two types of channels (the answers are yes for both the questions for Fig. 1). In this paper, we attempt a comprehensive study of such fundamental questions, about the expressive power, closure properties, and decision problems for probabilistic weighted automata. We focus on the positive and the almost-sure semantics. In future work, we will consider another semantics where the value of a word w is defined to be the expectation of the values of the runs over w.

First, we compare the expressiveness of the various classes of probabilistic and nondeterministic weighted automata over infinite words. For LimSup, LimInf, and limit average, we show that a wide variety of new classes of quantitative languages can be defined using probabilities, which are not expressible using nondeterminism. Our results rely on reachability properties of closed recurrent sets in Markov chains. For discounted sum, we show that probabilistic weighted automata under the positive semantics have the same expressive power as nondeterministic weighted automata, while under the almost-sure semantics, they have the same expressive power as weighted automata with universal branching, where the value of a word is the minimal (instead of maximal) value of all runs. The question of whether the positive semantics of weighted limit-average automata is more expressive than nondeterminism remains open.

Second, we give an almost complete picture of the closure of probabilistic weighted automata under the pointwise operations of maximum, minimum, and sum for quantitative languages. We also consider the numerical complement L^c of a quantitative language L defined by $L^c(w) = 1 - L(w)$ for all words w.[1] Note that maximum and minimum provide natural generalization of the classical union and intersection operations on boolean languages, and they define the least upper bound and greatest lower bound for the pointwise natural order on quantitative languages (where $L_1 \leq L_2$ if $L_1(w) \leq L_2(w)$ for all words w). The numerical complement applied to boolean languages also defines the usual complement operation.

[1] One can define $L^c(w) = k - L(w)$ for any constant k without changing our results.

Closure under max trivially holds for the positive semantics, and closure under min for the almost-sure semantics. Only LimSup-automata under positive semantics and LimInf-automata under almost-sure semantics are closed under all four operations; these results extend corresponding results for the boolean case [1]. To establish the closure properties of limit-average automata, we characterize the expected limit-average reward of Markov chains. Our characterization answers all closure questions except for the language sum in the case of positive semantics, which we leave open. Note that expressiveness results and closure properties are tightly connected. For instance, because they are closed under max, the LimInf-automata with positive semantics are no more expressive than to LimInf-automata with almost-sure semantics and to LimSup-automata with positive semantics; and because they are not closed under complement, the LimSup-automata with almost-sure semantics and LimInf-automata with positive semantics have incomparable expressive powers.

Third, we investigate the emptiness and universality problems for probabilistic weighted automata, which ask to decide if some (resp. all) words have a value above a given threshold. Using our expressiveness results, as well as [1, 9], we establish some decidability and undecidability results for Sup, LimSup, and LimInf automata; in particular, emptiness and universality are undecidable for LimSup-automata with positive semantics and for LimInf-automata with almost-sure semantics, while the question is open for the emptiness of LimInf-automata with positive semantics and for the universality of LimSup-automata with almost-sure semantics. We also prove the decidability of emptiness for probabilistic discounted-sum automata with positive semantics, while the universality problem is as hard as for nondeterministic discounted-sum automata, for which no decidability result is known. We leave open the case of limit average. Due to lack of space, we omit detailed proofs; they can be found in [10].

2 Definitions

A *quantitative language* over a finite alphabet Σ is a function $L : \Sigma^\omega \to \mathbb{R}$. A boolean language (or a set of infinite words) is a special case where $L(w) \in \{0, 1\}$ for all words $w \in \Sigma^\omega$. Nondeterministic weighted automata define the value of a word as the maximal value of a run [7]. In this paper, we study probabilistic weighted automata as generator of quantitative languages.

Value functions. We consider the following value functions $\mathsf{Val} : \mathbb{Q}^\omega \to \mathbb{R}$ to define quantitative languages. Given an infinite sequence $v = v_0 v_1 \dots$ of rational numbers, define

- $\mathsf{Sup}(v) = \sup\{v_n \mid n \geq 0\}$;
- $\mathsf{LimSup}(v) = \limsup_{n\to\infty} \; v_n = \lim_{n\to\infty} \sup\{v_i \mid i \geq n\}$;
- $\mathsf{LimInf}(v) = \liminf_{n\to\infty} \; v_n = \lim_{n\to\infty} \inf\{v_i \mid i \geq n\}$;
- $\mathsf{LimAvg}(v) = \liminf_{n\to\infty} \; \frac{1}{n} \cdot \sum_{i=0}^{n-1} v_i$;

– for $0 < \lambda < 1$, $\mathsf{Disc}_\lambda(v) = \sum_{i=0}^{\infty} \lambda^i \cdot v_i$.

Given a finite set S, a *probability distribution* over S is a function $f : S \to [0,1]$ such that $\sum_{s \in S} f(s) = 1$. We denote by $\mathcal{D}(S)$ the set of all probability distributions over S.

Probabilistic weighted automata. A *probabilistic weighted automaton* is a tuple $A = \langle Q, \rho_I, \Sigma, \delta, \gamma \rangle$, where

- Q is a finite set of states;
- $\rho_I \in \mathcal{D}(Q)$ is the initial probability distribution;
- Σ is a finite alphabet;
- $\delta : Q \times \Sigma \to \mathcal{D}(Q)$ is a probabilistic transition function;
- $\gamma : Q \times \Sigma \times Q \to \mathbb{Q}$ is a weight function.

The automaton A is *deterministic* if $\rho_I(q_I) = 1$ for some $q_I \in Q$, and for all $q \in Q$ and $\sigma \in \Sigma$, there exists $q' \in Q$ such that $\delta(q, \sigma)(q') = 1$.

A *run* of A over a finite (resp. infinite) word $w = \sigma_1 \sigma_2 \dots$ is a finite (resp. infinite) sequence $r = q_0 \sigma_1 q_1 \sigma_2 \dots$ of states and letters such that (i) $\rho_I(q_0) > 0$, and (ii) $\delta(q_i, \sigma_{i+1})(q_{i+1}) > 0$ for all $0 \le i < |w|$. We denote by $\gamma(r) = v_0 v_1 \dots$ the sequence of weights that occur in r where $v_i = \gamma(q_i, \sigma_{i+1}, q_{i+1})$ for all $0 \le i < |w|$. The probability of a finite run $r = q_0 \sigma_1 q_1 \sigma_2 \dots \sigma_k q_k$ over a finite word $w = \sigma_1 \dots \sigma_k$ is $\mathbb{P}^A(r) = \rho_I(q_0) . \prod_{i=1}^{k} \delta(q_{i-1}, \sigma_i)(q_i)$. For a finite run r, let $\mathsf{Cone}(r)$ denote the set of infinite runs r' such that r is a prefix of r'. The set of cones forms the basis for the Borel sets for runs. For each $w \in \Sigma^\omega$, the function $\mathbb{P}^A(\cdot)$ defines a unique probability measure over Borel sets of runs of A over w.

Given a value function $\mathsf{Val} : \mathbb{Q}^\omega \to \mathbb{R}$, we say that the probabilistic Val-automaton A generates the quantitative languages defined for all words $w \in \Sigma^\omega$ by $L_A^{=1}(w) = \sup\{\eta \mid \mathbb{P}^A(\{r \in \mathsf{Run}^A(w) \text{ such that } \mathsf{Val}(\gamma(r)) \ge \eta\}) = 1\}$ under the *almost-sure* semantics, and $L_A^{>0}(w) = \sup\{\eta \mid \mathbb{P}^A(\{r \in \mathsf{Run}^A(w) \text{ such that } \mathsf{Val}(\gamma(r)) \ge \eta\}) > 0\}$ under the *positive* semantics. In classical (non-probabilistic) semantics, the value of a word is defined either as the maximal value of the runs (i.e., $L_A^{\max}(w) = \sup\{\mathsf{Val}(\gamma(r)) \mid r \in \mathsf{Run}^A(w)\}$ for all $w \in \Sigma^\omega$) and the automaton is then called *nondeterministic*, or as the minimal value of the runs, and the automaton is then called *universal* [8]. Note that the above four semantics coincide for deterministic weighted automata (because then every word has exactly one run), and that Büchi and coBüchi automata [2] are special cases of respectively LimSup- and LimInf-automata, where all weights are either 0 or 1.

Reducibility. A class $\mathcal{C}$ of weighted automata is *reducible* to a class $\mathcal{C}'$ of weighted automata if for every $A \in \mathcal{C}$ there exists $A' \in \mathcal{C}'$ such that $L_A = L_{A'}$, i.e., $L_A(w) = L_{A'}(w)$ for all words w. Reducibility relationships for (non)deterministic weighted automata are given in [7].

Composition. Given two quantitative languages $L, L' : \Sigma^\omega \to \mathbb{R}$, we denote by $\max(L, L')$ (resp. $\min(L, L')$ and $L + L'$) the quantitative language that assigns $\max\{L(w), L'(w)\}$ (resp. $\min\{L(w), L'(w)\}$ and $L(w) + L'(w)$) to each word

$w \in \Sigma^\omega$. The language $1 - L$ is called the *complement* of L. The max, min and complement operators for quantitative languages generalize respectively the union, intersection and complement operator for boolean languages. The closure properties of (non)deterministic weighted automata are given in [9].

Notation. The first letter in acronyms for classes of automata can be N(ondeterministic), D(eterministic), U(niversal), Pos for the language in the positive semantics, or As for the language in the almost-sure semantics. For $X, Y \in \{$N, D, U, Pos, As$\}$, we sometimes use the notation $\genfrac{}{}{0pt}{}{\mathrm{X}}{\mathrm{Y}}$ for classes of automata where the X and Y versions are reducible to each other. For Büchi and coBüchi automata, we use the classical acronyms NBW, DBW, NCW, etc. When the type of an automaton A is clear from the context, we often denote its language simply by $L_A(\cdot)$ or even $A(\cdot)$, instead of $L_A^{=1}$, $L_A^{\max}$, etc.

Remark. We sometimes use automata with weight functions $\gamma : Q \to \mathbb{Q}$ that assign a weight to states instead of transitions. This is a convenient notation for weighted automata in which from each state, all outgoing transitions have the same weight. In pictorial descriptions of probabilistic weighted automata, the transitions are labeled with probabilities, and states with weights.

3 Expressive Power of Probabilistic Weighted Automata

We complete the picture given in [7] about reducibility for nondeterministic weighted automata, by adding the relations with probabilistic automata. The results for LimInf, LimSup, and LimAvg are summarized in Fig. 2s, and for Sup- and Disc-automata in Theorems 1 and 6.

As for probabilistic automata over finite words, the quantitative languages definable by probabilistic and (non)deterministic Sup-automata coincide.

Theorem 1. DSup *is as expressive as* PosSup *and* AsSup.

In many of our results, we use the following definitions and properties related to Markov chains. A *Markov chain* $M = (S, E, \delta)$ consists of a finite set S of states, a set E of edges, and a probabilistic transition function $\delta : S \to \mathcal{D}(S)$. For all $s, t \in S$, there is an edge $(s, t) \in E$ iff $\delta(s)(t) > 0$. A *closed recurrent set* C of states in M is a bottom strongly connected set of states in the graph (S, E). The proof of the Lemma 1 relies on the following basic properties [21]. Lemma 1 will be used in the proof of some of the following results.

1. *Property 1.* Given a Markov chain M, and a start state s, with probability 1, the set of closed recurrent states is reached from s in finite time. Hence for any $\epsilon > 0$, there exists k_0 such that for all $k > k_0$, for all starting state s, the set of closed recurrent states are reached with probability at least $1 - \epsilon$ in k steps.
2. *Property 2.* If a closed recurrent set C is reached, and the limit of the expectation of the average weights of C is α, then for all $\epsilon > 0$, there exists a k_0 such that for all $k > k_0$ the expectation of the average weights for k steps is at least $\alpha - \epsilon$.

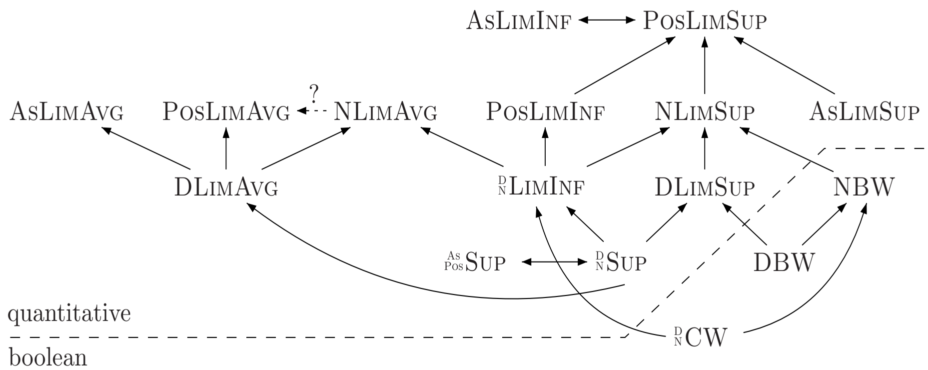


Fig. 2. Reducibility relation. $\mathcal{C}$ is reducible to $\mathcal{C}'$ if $\mathcal{C} \to \mathcal{C}'$. Classes that are not connected by an arrow are incomparable. Reducibility for the dashed arrow is open. The Disc-automata are incomparable with the automata in the figure. Their reducibility relation is given in Theorem 6.

Lemma 1. *Let A be a probabilistic weighted automaton with alphabet $\Sigma = \{a, b\}$. Consider the Markov chain arising from A on input b^ω (we refer to this as the b-Markov chain) and the a-Markov chain is defined symmetrically. The following assertions hold:*

1. *If for all closed recurrent sets C in the b-Markov chain, the expected limit-average value is at least 1, then there exists j such that for all closed recurrent sets arising from A on input $(b^j \cdot a)^\omega$ the expected limit-average reward is positive.*
2. *If for all closed recurrent sets C in the b-Markov chain, the expected limit-average value is at most 0, then there exists j such that for all closed recurrent sets arising from A on input $(b^j \cdot a)^\omega$ the expected limit-average reward is strictly less than 1.*
3. *If for all closed recurrent sets C in the b-Markov chain, the expected limit-average value is at most 0,and if for all closed recurrent sets C in the a-Markov chain, the expected limit-average value is at most 0, then there exists j such that for all closed recurrent sets arising from A on input $(b^j \cdot a^j)^\omega$ the expected limit-average reward is strictly less than 1/2.*

Proof. We present the proof of the first part. Let β be the maximum absolute value of the weights of A. From any state $s \in A$, there is a path of length at most n to a closed recurrent set C in the b-Markov chain, where n is the number of states of A. Hence if we choose $j > n$, then any closed recurrent set in the Markov chain arising on the input $(b^j \cdot a)^\omega$ contains closed recurrent sets of the b-Markov chain. For $\epsilon > 0$, there exists k_ϵ such that from any state $s \in A$, for all $k > k_\epsilon$, on input b^k from s, the closed recurrent sets of the b-Markov chain is reached with probability at least $1 - \epsilon$ (by property 1). If all closed recurrent

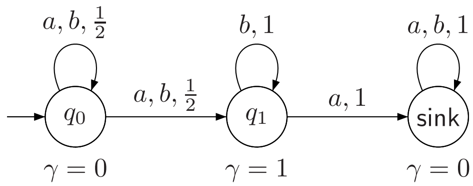


Fig. 3. PosLimAvg for Lemma 2.

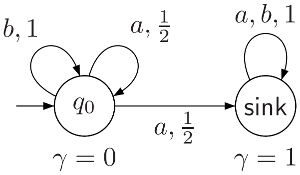


Fig. 4. AsLimAvg for Lemma 3.

sets in the b-Markov chain have expected limit-average value at least 1, then by property 2 it follows that for all $\epsilon > 0$, there exists l_ϵ such that for all $l > l_\epsilon$, from all states s of a closed recurrent set on the input b^l the expected average of the weights is at least $1 - \epsilon$, (i.e., expected sum of the weights is $l - l \cdot \epsilon$). Consider $0 < \epsilon \leq \min\{1/4, 1/(20 \cdot \beta)\}$, we choose $j = k + l$, where $k = k_\epsilon > 0$ and $l > \max\{l_\epsilon, k\}$. Observe that by our choice $j + 1 \leq 2l$. Consider a closed recurrent set in the Markov chain on $(b^j \cdot a)^\omega$ and we obtain a lower bound on the expected average reward as follows: with probability $1 - \epsilon$ the closed recurrent set of the b-Markov chain is reached within k steps, and then in the next l steps at the expected sum of the weights is at least $l - l \cdot \epsilon$, and since the worst case weight is $-\beta$ we obtain the following bound on the expected sum of the rewards:

$$(1-\epsilon) \cdot (l - l \cdot \epsilon) - \epsilon \cdot \beta \cdot (j+1) \geq \frac{l}{2} - \frac{l}{10} = \frac{2l}{5}$$

Hence the expected average reward is at least $1/5$ and hence positive. ■

3.1 Probabilistic LimAvg-automata

We consider the alphabet $\Sigma = \{a, b\}$ and we define the boolean language L_F of finitely many a's, i.e., $L_F(w) = 1$ if $w \in \Sigma^\omega$ consists of finitely many a's, and $L_F(w) = 0$ otherwise. We also consider the language L_I of words with infinitely many a's, i.e., the complement of L_F.

Lemma 2. *Consider the language L_F of finitely many a's. The following assertions hold.*

1. *There is no* NLimAvg *that specifies L_F.*
2. *There exists a* PosLimAvg *that specifies L_F (see Fig. 3).*
3. *There is no* AsLimAvg *that specifies L_F.*

Proof. We present the proof of the third part. Assume that there exists an AsLimAvg automaton A that specifies L_F. Consider the Markov chain M that arises from A if the input is only b (i.e., on b^ω), we refer to it as the b-Markov chain. If there is a closed recurrent set C in M that can be reached from the starting state in A (reached by any sequence of a and b's in A), then the expected

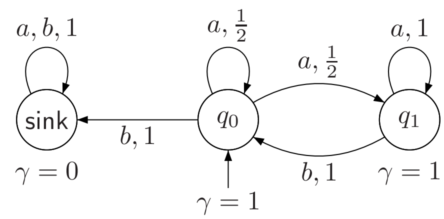


Fig. 5. A probabilistic weighted automaton (PosLimAvg, PosLimSup, or PosLimInf) for Lemma 4.

limit-average reward in C must be at least 1 (otherwise, if there is a closed recurrent set C in M with limit-average reward less than 1, we can construct a finite word w that will reach C with positive probability in A, and then follow w by b^ω yielding $A(w \cdot b^\omega) < 1$). Thus any closed recurrent set in M has limit-average reward at least 1 and by Lemma 1 there exists j such that the $A((b^j \cdot a)^\omega) > 0$. It follows that A cannot specify L_F. ∎

Lemma 3. *Consider the language L_I of infinitely many a's. The following assertions hold.*

1. *There is no* NLimAvg *that specifies L_I.*
2. *There is no* PosLimAvg *that specifies L_I.*
3. *There exists an* AsLimAvg *that specifies L_I (see Fig. 4).*

Lemma 4. *There exists a language L such that: (a) there exists a* PosLimAvg*, a* PosLimSup *and a* PosLimInf *that specifies L (see Fig. 5); and (b) there is no* NLimAvg*, no* NLimSup *and no* NLimInf *that specifies L.*

The next theorem summarizes the results for limit-average automata obtained in this section.

Theorem 2. AsLimAvg *is incomparable in expressive power with* PosLimAvg *and* NLimAvg*, and* NLimAvg *is not as expressive as* PosLimAvg*.*

Open question. Whether NLimAvg is reducible to PosLimAvg or NLimAvg is incomparable to PosLimAvg (i.e., whether there is a language expressible by NLimAvg but not by PosLimAvg) remains open.

3.2 Probabilistic LimInf- and LimSup-automata

To compare the expressiveness of probabilistic LimInf- and LimSup-automata, we use and extend results from [1, 7], Lemma 3 and 4, and the notion of *determinism in the limit* [12, 25]. A nondeterministic weighted automaton A is *deterministic in the limit* if for all states s of A with weight greater than the minimum weight, all states t reachable from s have deterministic transitions.

Lemma 5. *For every* NLimSup A*, there exists a* NLimSup B *that is deterministic in the limit and specifies the same quantitative language.*

Lemma 5 is useful to translate a nondeterministic automaton into a probabilistic one with positive semantics. The next lemma presents the results about reducibility of liminf automata.

Lemma 6. *The following assertions hold: (a) both* AsLimInf *and* PosLimInf *are as expressive as* NLimInf*; (b) there exists an* AsLimInf *that specifies the language* L_I*, there is no* NLimInf *and there is no* PosLimInf *that specifies* L_I*; (c)* AsLimInf *is as expressive as* PosLimInf*.*

As a corollary of Lemma 4 and Lemma 6, we get the following theorem.

Theorem 3. AsLimInf *is strictly more expressive than* PosLimInf*; and* PosLimInf *is strictly more expressive than* NLimInf*.*

The following lemma presents the results about reducibility of limsup automata.

Lemma 7. *The following assertions hold: (a)* NLimSup *and* AsLimSup *are not as expressive as* PosLimSup*; (b)* PosLimSup *is as expressive as* NLimSup*; (c)* PosLimSup *is as expressive as* AsLimSup*; (d)* AsLimSup *is not as expressive as* NLimSup*.*

Theorem 4. AsLimSup *and* NLimSup *are incomparable in expressive power, and* PosLimSup *is strictly more expressive than* AsLimSup *and* NLimSup*.*

The above theorem summarizes the reducibility results for limsup automata. Finally, we establish the reducibility relation between probabilistic LimSup- and LimInf-automata.

Theorem 5. AsLimInf *and* PosLimSup *have the same expressive power;* AsLimSup *and* PosLimInf *have incomparable expressive power.*

Proof. This result is an easy consequence of the fact that an automaton interpreted as AsLimInf specifies the complement of the language of the same automaton interpreted as PosLimSup (and similarly for AsLimSup and PosLimInf), and from the fact that AsLimInf and PosLimSup are closed under complement, while AsLimSup and PosLimInf are not (see Lemma 13). ■

3.3 Probabilistic Disc-automata

For probabilistic discounted-sum automata, the nondeterministic and the positive semantics have the same expressive power. Intuitively, this is because the run with maximal value can be approached arbitrarily close by a finite run, and therefore the set of infinite runs sharing that finite run as a prefix has positive probability. This also shows that the positive semantics does not depend on the actual values of the probabilities, but only on whether they are positive or not. Analogous results hold for the universal semantics and the almost-sure semantics.

Theorem 6. *The following assertions hold: (a)* NDisc *and* PosDisc *have the same expressive power; (b)* UDisc *and* AsDisc *have the same expressive power.*

Proof. (a) Let $A = \langle Q, \rho_I, \Sigma, \delta_A, \gamma \rangle$ be a NDisc, and let $v_{\min}, v_{\max}$ be its minimum and maximum weights respectively. Consider the PosDisc $B = \langle Q, \rho_I, \Sigma, \delta_B, \gamma \rangle$ where $\delta_B(q, \sigma)$ is the uniform probability distribution over the set of states q' such that $(q, \sigma, \{q'\}) \in \delta_A$. Let $r = q_0 \sigma_1 q_1 \sigma_2 \dots$ be a run of A (over $w = \sigma_1 \sigma_2 \dots$) with value η. For all $\epsilon > 0$, we show that $\mathbb{P}^B(\{r \in \mathsf{Run}^B(w) \mid \mathsf{Val}(\gamma(r)) \geq \eta - \epsilon\}) > 0\}$. Let $n \in \mathbb{N}$ such that $\frac{\lambda^n}{1-\lambda} \cdot (v_{\max} - v_{\min}) \leq \epsilon$, and let $r_n = q_0 \sigma_1 q_1 \sigma_2 \dots \sigma_n q_n$. The discounted sum of the weights in r_n is at least $\eta - \frac{\lambda^n}{1-\lambda} \cdot (v_{\max})$. The probability of the set of runs over w that are continuations of r_n is positive, and the value of all these runs is at least $\eta - \frac{\lambda^n}{1-\lambda} \cdot (v_{\max} - v_{\min})$, and therefore at least $\eta - \epsilon$. This shows that $L_B(w) \geq \eta$, and thus $L_B(w) \geq L_A(w)$. Note that $L_B(w) \leq L_A(w)$ since there is no run in A (nor in B) over w with value greater than $L_A(w)$. Hence $L_B = L_A$.

Now, we prove that PosDisc is reducible to NDisc. Given a PosDisc $B = \langle Q, \rho_I, \Sigma, \delta_B, \gamma \rangle$, we construct a NDisc $A = \langle Q, \rho_I, \Sigma, \delta_A, \gamma \rangle$ where $(q, \sigma, \{q'\}) \in \delta_A$ if and only if $\delta_B(q, \sigma)(q') > 0$, for all $q, q' \in Q$, $\sigma \in \Sigma$. By analogous arguments as in the first part of the proof, it is easy to see that $L_B = L_A$.

(b) The complement of the quantitative language specified by an UDisc (resp. AsDisc) can be specified by a NDisc (resp. PosDisc). Then, the result follows from Part a) (essentially, given an UDisc, we obtain easily a NDisc for the complement, then an equivalent PosDisc, and finally an AsDisc for the complement of the complement, i.e., the original quantitative language). ■

4 Closure Properties of Probabilistic Weighted Automata

We consider the closure properties of the probabilistic weighted automata under the operations max, min, complement, and sum.

Closure under max **and** min**.** The closure under max holds for the positive semantics (and under min for the almost-sure semantics) using initial nondeterminism (Lemma 8), while a synchronized product can be used for AsLimSup and PosLimInf (Lemma 9). In Lemma 10, we use the closure under intersection of probabilistic Büchi automata [2], and the closure under max of PosLimSup.

Lemma 8. PosLimSup*,* PosLimInf*, and* PosLimAvg *are closed under* max*; and* AsLimSup*,* AsLimInf*, and* AsLimAvg *are closed under* min*.*

Lemma 9. AsLimSup *is closed under* max*;* PosLimInf *is closed under* min*.*

Lemma 10. PosLimSup *is closed under* min*;* AsLimInf *is closed under* max*.*

		max	min	comp.	sum	emptiness	universality
positive	PosSup	✓	✓	×	✓	✓	✓
	PosLimSup	✓	✓	✓	✓	×	×
	PosLimInf	✓	✓	×	✓	✓	✓
	PosLimAvg	✓	×	×	?	?	?
	PosDisc	✓	×	×	✓	✓	? (1)
almost-sure	AsSup	✓	✓	×	✓	✓	✓
	AsLimSup	✓	✓	×	✓	✓	✓
	AsLimInf	✓	✓	✓	✓	×	×
	AsLimAvg	×	✓	×	×	?	?
	AsDisc	×	✓	×	✓	? (1)	✓

The universality problem for NDisc can be reduced to (1).
It is not known whether this problem is decidable.

Table 1. Closure properties and decidability of emptiness and universality.

The closure properties of LimAvg-automata in the positive semantics rely on the following lemma.

Lemma 11. *Consider the alphabet $\Sigma = \{a, b\}$, and consider the languages L_a and L_b that assign the long-run average number of a's and b's, respectively. Then the following assertions hold:*

1. *There is no* PosLimAvg *that specifies the language $L_m = \min\{L_a, L_b\}$.*
2. *There is no* PosLimAvg *that specifies the language $L^* = 1 - \max\{L_a, L_b\}$.*

Lemma 12. PosLimAvg *is not closed under* min*, and* AsLimAvg *is not closed under* max*.*

Proof. The result for PosLimAvg follows from Lemma 11. We show that AsLimAvg is not closed under max. Consider the alphabet $\Sigma = \{a, b\}$ and the quantitative languages L_a and L_b that assign the long-run average number of a's and b's, respectively. There exist DLimAvg (and hence AsLimAvg) to specify L_a and L_b. We show that $L_m = \max(L_a, L_b)$ cannot be specified by an AsLimAvg. By contradiction, assume that A is an AsLimAvg with set of states Q that specifies L_m. Consider any closed recurrent set of the a-Markov chain of A. The expected limit-average of the weights of the recurrent set must be 1, as if we consider the word $w^* = w_C \cdot a^\omega$ where w_C is a finite word to reach C in A, the value of w^* in L_m is 1. Hence, the limit-average of the weights of all the reachable a-closed recurrent set C in A is 1.

Given $\epsilon > 0$, there exists j_ϵ such that the following properties hold:

1. from any state of A, given the word a^{j_ϵ} with probability $1 - \epsilon$ an a-closed recurrent set is reached (by property 1 for Markov chains);
2. once an a-closed recurrent set is reached, given the word a^{j_ϵ}, (as a consequence of property 2 for Markov chains) we can show that the following

properties hold: (a) the expected average of the weights is at least $j_\epsilon \cdot (1-\epsilon)$, and (b) the probability distribution of the states is with ϵ of the probability distribution of the states for the word $a^{2\cdot j_\epsilon}$ (this holds as the probability distribution of states on words a^j converges to the probability distribution of states on the word a^ω).

Let $\beta > 1$ be a number that is greater than the absolute maximum value of weights in A. We chose $\epsilon > 0$ such that $\epsilon < \frac{1}{40\cdot\beta}$. Let $j = 2 \cdot j_\epsilon$ (such that j_ϵ satisfies the properties above). Consider the word $(a^j \cdot b^{3j})^\omega$ and the answer by A must be $\frac{3}{4}$, as $L_m((a^j \cdot b^{3j})^\omega) = \frac{3}{4}$. Consider the word $\widehat{w} = (a^{2j} \cdot b^{3j})^\omega$ and consider a closed recurrent set in the Markov chain obtain from A on $\widehat{w}$. We obtain the following lower bound on the expected limit-average of the weights: (a) with probability at least $1-\epsilon$, after $j/2$ steps, a-closed recurrent sets are reached; (b) the expected average of the weights for the segment between a^j and a^{2j} is at least $j \cdot (1-\epsilon)$; and (c) the difference in probability distribution of the states after a^j and a^{2j} is at most ϵ. Since the limit-average of the weights of $(a^j \cdot b^{3j})^\omega$ is $\frac{3}{4}$, the lower bound on the limit-average of the weights is as follows

$$\begin{aligned}(1-3\cdot\epsilon)\cdot(\tfrac{3\cdot j + j\cdot(1-\epsilon)}{5j}) - 3\cdot\epsilon\cdot\beta &= (1-\epsilon)\cdot(\tfrac{4}{5}-\tfrac{\epsilon}{5}) - 3\cdot\epsilon\cdot\beta \\ &\geq \tfrac{4}{5} - \epsilon - 3\cdot\epsilon\cdot\beta \geq \tfrac{4}{5} - 4\cdot\epsilon\cdot\beta \\ &\geq \tfrac{4}{5} - \tfrac{1}{10} \geq \tfrac{7}{10} > \tfrac{3}{5}.\end{aligned}$$

It follows that $A((a^{2j} \cdot b^{3j})^\omega) > \frac{3}{5}$. This contradicts that A specifies L_m. ■

Closure under complement and sum. We now consider closure under complement and sum.

Lemma 13. PosLimSup *and* AsLimInf *are closed under complement; all other classes of probabilistic weighted automata are not closed under complement.*

Proof. We give the proof for limit average. The fact that PosLimAvg is not closed under complement follows from Lemma 11. We now show that AsLimAvg is not closed under complement. Consider the DLimAvg A over alphabet $\Sigma = \{a, b\}$ that consists of a single self-loop state with weight 1 for a and 0 for b. Notice that $A(w.a^\omega) = 1$ and $A(w.b^\omega) = 0$ for all $w \in \Sigma^*$. To obtain a contradiction, assume that there exists a AsLimAvg B such that $B = 1 - A$. For all finite words $w \in \Sigma^*$, let $B(w)$ be the expected average weight of the finite run of B over w. Fix $0 < \epsilon < \frac{1}{2}$. For all finite words w, there exists a number n such that the average number of a's in $w.b^n$ is at most ϵ, and there exists a number m such that $B(w.a^m) \leq \epsilon$ (since $B(w.a^\omega) = 0$). Hence, we can construct a word $w = b^{n_1}a^{m_1}b^{n_2}a^{m_2}\ldots$ such that $A(w) \leq \epsilon$ and $B(w) \leq \epsilon$. Since $B = 1 - A$, this implies that $1 \leq 2\epsilon$, a contradiction. ■

Lemma 14. *The* Sup-, LimSup-, LimInf-, *and* Disc-*automata are closed under sum under both the positive and almost-sure semantics.* AsLimAvg *is not closed under sum.*

Theorem 7. *The closure properties for probabilistic weighted automata under max, min, complement, and sum are summarized in Table 1.*

Open question. Whether PosLimAvg is closed under sum remains open.

5 Decision Problems

We conclude the paper with some decidability and undecidability results for classical decision problems about quantitative languages (see Table 1). Most of them are direct corollaries of the results in [1]. Given a weighted automaton A and a rational number $\nu \in \mathbb{Q}$, the *quantitative emptiness problem* asks whether there exists a word $w \in \Sigma^\omega$ such that $L_A(w) \geq \nu$, and the *quantitative universality problem* asks whether $L_A(w) \geq \nu$ for all words $w \in \Sigma^\omega$.

Theorem 8. *The emptiness and universality problems for* PosSup*,* AsSup*,* AsLimSup*, and* PosLimInf *are decidable.*

Theorem 9. *The emptiness and universality problems for* PosLimSup *and* AsLimInf *are undecidable.*

Finally, by Theorem 6 and the decidability of emptiness for NDisc, we get the following result.

Theorem 10. *The emptiness problem for* PosDisc *and the universality problem for* AsDisc *are decidable.*

Note that by Theorem 6, the universality problem for NDisc (which is not know to be decidable) can be reduced to the universality problem for PosDisc and to the emptiness problem for AsDisc.

Language inclusion. Given two weighted automata A and B, the *quantitative language-inclusion problem* asks whether for all words $w \in \Sigma^\omega$ we have $L_A(w) \geq L_B(w)$ and the *quantitative language-equivalence problem* asks whether for all words $w \in \Sigma^\omega$ we have $L_A(w) = L_B(w)$. It follows from our results that the language-inclusion problem is decidable for PosSup and AsSup, and is undecidable for PosLimSup and AsLimInf. The decidability of language inclusion for PosLimInf and AsLimSup remains open; the problem is also open for the respective boolean cases (i.e., for PosCW and AsBW). The decidability of language inclusion for PosLimAvg, AsLimAvg, PosDisc, and AsDisc also remains open as either the universality or the emptiness problem (or both) remain open in the respective cases. The situation for language equivalence is the same as for language inclusion.

References

1. C. Baier, N. Bertrand, and M. Größer. On decision problems for probabilistic Büchi automata. In *Proc. of FoSSaCS*, LNCS 4962, pages 287–301. Springer, 2008.

2. C. Baier and M. Größer. Recognizing omega-regular languages with probabilistic automata. In *Proc. of LICS: Logic in Comp. Science*, pages 137–146. IEEE, 2005.
3. Vincent D. Blondel and Vincent Canterini. Undecidable problems for probabilistic automata of fixed dimension. *Theory Comput. Syst.*, 36(3):231–245, 2003.
4. A. Chakrabarti, K. Chatterjee, T. A. Henzinger, O. Kupferman, and R. Majumdar. Verifying quantitative properties using bound functions. In *Proc. of CHARME*, LNCS 3725, pages 50–64. Springer, 2005.
5. A. Chakrabarti, L. de Alfaro, T. A. Henzinger, and M. Stoelinga. Resource interfaces. In *Proc. of EMSOFT*, LNCS 2855, pages 117–133. Springer, 2003.
6. K. Chatterjee, L. de Alfaro, M. Faella, T.A. Henzinger, R. Majumdar, and M. Stoelinga. Compositional quantitative reasoning. In *Proc. of QEST*, pages 179–188. IEEE Computer Society Press, 2006.
7. K. Chatterjee, L. Doyen, and T. A. Henzinger. Quantitative languages. In *Proc. of CSL: Computer Science Logic*, LNCS 5213, pages 385–400. Springer, 2008.
8. K. Chatterjee, L. Doyen, and T. A. Henzinger. Alternating weighted automata. In *Proc. of FCT: Fundamentals of Computation Theory*. Springer, 2009. To appear.
9. K. Chatterjee, L. Doyen, and T. A. Henzinger. Expressiveness and closure properties for quantitative languages. In *Proc. of LICS: Logic in Computer Science*. IEEE Comp. Soc. Press, 2009. To appear.
10. K. Chatterjee, L. Doyen, and T. A. Henzinger. Probabilistic weighted automata, 2009. http://infoscience.epfl.ch/record/133665.
11. K. Chatterjee, M. Jurdziński, and T.A. Henzinger. Quantitative stochastic parity games. In *Proc. of SODA*, pages 114–123. ACM Press, 2004.
12. C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.
13. K. Culik II and J. Karhumäki. Finite automata computing real functions. *SIAM J. Comput.*, 23(4):789–814, 1994.
14. L. de Alfaro, M. Faella, and M. Stoelinga. Linear and branching metrics for quantitative transition systems. In *Proc. of ICALP*, LNCS 3142, pages 97–109. Springer, 2004.
15. M. Droste and D. Kuske. Skew and infinitary formal power series. *Theor. Comput. Sci.*, 366(3):199–227, 2006.
16. A. Ehrenfeucht and J. Mycielski. Positional strategies for mean payoff games. *Int. Journal of Game Theory*, 8(2):109–113, 1979.
17. Z. Ésik and W. Kuich. An algebraic generalization of omega-regular languages. In *Proc. of MFCS*, LNCS 3153, pages 648–659. Springer, 2004.
18. J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer, 1997.
19. H. Gimbert. *Jeux positionnels*. PhD thesis, Université Paris 7, 2006.
20. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games*, Lecture Notes in Computer Science 2500. Springer, 2002.
21. J. G. Kemeny, J. L. Snell, and A. W. Knapp. *Denumerable Markov Chains*. D. Van Nostrand Company, 1966.
22. Michael O. Rabin. Probabilistic automata. *Information and Control*, 6(3):230–245, 1963.
23. L. S. Shapley. Stochastic games. In *Proc. of the National Acadamy of Science USA*, volume 39, pages 1095–1100, 1953.
24. M. Y. Vardi. An automata-theoretic approach to linear temporal logic. In *Proc. of Banff Higher Order Workshop*, LNCS 1043, pages 238–266. Springer, 1995.
25. Moshe Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *Proc. of FOCS*, pages 327–338. IEEE, 1985.