# Probabilistic Weighted Automata

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

[1] University of California, Santa Cruz
[2] Université Libre de Bruxelles, Belgium
[3] EPFL, Lausanne, 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 a word is naturally computed as the maximal value of a run, and the value of a run as the maximum, 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 fashion. Under almost-sure semantics (resp. positive semantics), the value of a word $w$ is the largest $v$ such that the runs produced by the automaton over $w$ have value at least $v$ with probability 1 (resp. positive probability). The main results of this paper are a comparison of the expressive power of the various classes of probabilistic and nondeterministic weighted automata over infinite words. In particular, for limit average, we show that probabilities define a wide variety of new classes of quantitative languages, while for discounted sum, probabilities do not bring more expressive power than nondeterminism.
We next study the closure properties of probabilistic weighted automata. For quantitative languages $L_1$ and $L_2$, we consider the operations $max(L_1, L_2)$, $min(L_1, L_2)$, $1 - L_1$, and the sum $L_1 + L_2$. Finally, we give (un)decidability results for the emptiness and universality problems.

## 1 Introduction

In formal design, specifications describe the set of correct behaviours of a system. An implementation satisfies a specification if all its behaviours are correct. If we view a behaviour 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 e.g. [10, 11]. We call them *boolean languages* because a given behaviour is either good or bad according to the specification. Boolean languages are useful to specify computational (or functional) requirements.

In a generalization of this approach, we consider *quantitative languages* where each word is assigned a real number. The value of a word can be interpreted as the amount of some resource (e.g., memory consumption, or power consumption) needed to produce it, or as a quality measurement of the corresponding behaviour. Therefore, quantitative languages are useful to specify non-purely computational requirements such as resource constraints, reliability requirements, or level of quality.

Nondeterministic weighted automata (i.e., finite automata with numerical weights on transitions) have been used to define quantitative languages over infinite words [8, 9, 6]. In [6], we defined the value of an infinite word $w$ as the maximal value of all runs over $w$ (if the automaton is nondeterministic, then there may be many runs over $w$), and the value of a run $r$ is a function of the infinite sequence of weights that appear along $r$. We consider several functions, such as $\mathsf{Sup}$, $\mathsf{LimSup}$, $\mathsf{LimInf}$, 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 can be modeled as the discounted sum; average response time as the limit average [3, 4].

In this paper, we consider probabilistic weighted automata as generator of quantitative languages. The value of an infinite word $w$ is defined as 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). The probabilistic Büchi and coBüchi automata of [2] are a special case of probabilistic weighted automata with weights 0 and 1 only (and value computed as $\mathsf{LimSup}$ and $\mathsf{LimInf}$). We are not aware of any other model combining probabilities and weights for the specification of quantitative languages.

As a continuation of [5, 6], we consider fundamental questions about the expressive power, the closure properties, and the emptiness and universality problems for probabilistic weighted automata.

We compare the expressive power of the various classes of probabilistic and nondeterministic weighted automata over infinite words. For $\mathsf{LimSup}$, $\mathsf{LimInf}$, and limit average, we show that a wide variety of new classes of quantitative languages can be defined with probabilitic automata, but are not expressible using nondeterminism. The results are based 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 the nondeterministic ones. Under the almost-sure semantics, they have the same expressive power as the automata with universal branching, where the value of a word is the minimal (instead of maximal) value of a run. On the other hand, some questions remain open about expressiveness, for instance which of limit-average automata with almost-sure or positive probability is more expressive, or whether they are incomparable.

We next study the closure properties of probabilistic weighted automata. We consider the operations of *maximum*, *minimum*, and *sum* defined, for quantitative languages $L_1$ and $L_2$, as the quantitative language that assigns $\max(L_1(w), L_2(w))$, $\min(L_1(w), L_2(w))$, and $L_1(w) + L_2(w)$ to each word $w$. The *complement* $L^c$ of a quantitative language $L$ is defined by $L^c(w) = 1 - L(w)$ for all words $w$. Note that closure under *max* always holds for the positive semantics, and closure under *min* always holds for the almost-sure semantics. The closure properties of $\mathsf{Sup}$-, $\mathsf{LimSup}$-, and $\mathsf{LimInf}$-automata are obtained as an extension of known results for the boolean finite automata [1], and for discounted sum as a corollary of our results about expressiveness and [5]. Only $\mathsf{LimSup}$-automata under positive semantics and $\mathsf{LimInf}$-automata under almost-sure semantics are closed under the four operations. To establish the closure properties of limit-average automata, we study the expected limit-average reward of Markov chains. This approach solves all closure questions except for the sum in the *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 can be reduced 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 expressiveness.

Finally, and for the sake of completeness, we consider the classical emptiness and universality problems which ask to decide if some (resp. all) words have value above a given threshold. Using our results about expressiveness and [1, 5], we establsih decidability and undecidability results for Sup-, LimSup-, LimInf, and discounted sum. We leave the question open for limit average.

♣ **Add some more positive/optimistic statement ??** ♣

## 2 Definitions

A quantitative language over a finite alphabet $\Sigma$ 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 [6]. 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

- $\mathsf{Sup}(v) = \sup\{v_n \mid n \geq 0\}$;
- $\mathsf{LimSup}(v) = \limsup\limits_{n\to\infty} \; v_n = \lim\limits_{n\to\infty} \sup\{v_i \mid i \geq n\}$;
- $\mathsf{LimInf}(v) = \liminf\limits_{n\to\infty} \; v_n = \lim\limits_{n\to\infty} \inf\{v_i \mid i \geq n\}$;
- $\mathsf{LimAvg}(v) = \liminf\limits_{n\to\infty} \; \dfrac{1}{n} \displaystyle\sum_{i=0}^{n-1} v_i$;
- For $0 < \lambda < 1$, $\mathsf{Disc}_\lambda(v) = \displaystyle\sum_{i=0}^{\infty} \lambda^i \cdot v_i$;

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

**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 distribution;
- $\Sigma$ 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.

We can define a *non-probabilistic* automaton from $A$ by ignoring the probability values, and saying that $q$ is initial if $\rho_I(q) > 0$, and $(q, \sigma, q')$ is an edge of $A$ if $\delta(q, \sigma)(q') > 0$. The automaton $A$ is *deterministic* if 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 $0 \leq i < |w|$. 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 $0 \leq i < |w|$.

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 $\mathbb{P}^A(r) = \rho_I(q_0) \cdot \prod_{i=1}^{k} \delta(q_{i-1}, \sigma_i)(q_i)$. For each $w \in \Sigma^\omega$, the function $\mathbb{P}^A(\cdot)$ defines a unique probability measure over Borel sets of (infinite) runs of $A$ over $w$. Given a function $f$ that maps each run to a real number, we denote by $\mathbb{E}^A_w(f)$ the expected value of $f$ over the runs of $A$ over $w$.

Given a value function $\mathsf{Val} : \mathbb{Q}^\omega \to \mathbb{R}$, we say that the probabilistic $\mathsf{Val}$-automaton $A$ generates the *quantitative language* defined for all words $w \in \Sigma^\omega$ by $L_A^{=1}(w) = \sup\{\eta \mid \mathbb{P}^A(\{r \in \mathsf{Run}^A(w) \mid \mathsf{Val}(\gamma(r)) \geq \eta\}) = 1\}$ under the almost-sure semantics, and $L_A^{>0}(w) = \sup\{\eta \mid \mathbb{P}^A(\{r \in \mathsf{Run}^A(w) \mid \mathsf{Val}(\gamma(r)) \geq \eta\}) > 0\}$ under the positive semantics. For non-probabilistic automata, the value of a word is either 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 the minimal value of the runs, qnd the automaton is then called *universal*.

Note that Büchi and coBüchi automata ([2]) are special cases of respectively $\mathsf{LimSup}$- and $\mathsf{LimInf}$-automata, where all weights are either 0 or 1.

**Notations.** The first letter in acronyms for classes of automata can be N(ondeterministic), D(eterministic), U(niversal), Z for the language in the positive semantics, or As for the language in the almost-sure semantics. We use the notations $^{\text{D}}_{\text{N}}$ to denote the classes of automata whose deterministic version has the same expressiveness as their nondeterministic version. 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.

**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 (finite or infinite) words $w$. Reducibility relationships for (non)deterministic weighted automata are given in [6].

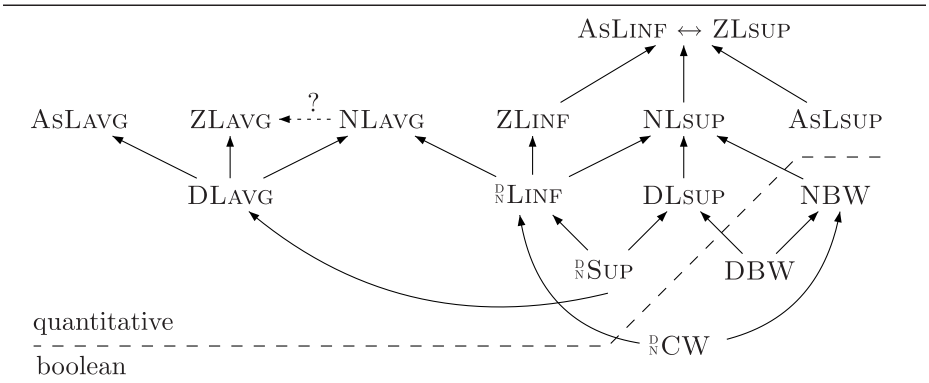


**Fig. 1.** Reducibility relation. $\mathcal{C}$ is reducible to $\mathcal{C}'$ if $\mathcal{C} \to \mathcal{C}'$. Classes that are not connected by an arrow are incomparable. Reducibility for the dashed arrow is open.

**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 [5].

**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. For automata with weights in $\{0, 1\}$, the states with weights 1 are *accepting* states; runs with infinitely many 1's in Büchi automata, or finitely many 1's in coBüchi automata are *accepting* runs.

## 3 Expressive Power of Probabilistic Weighted Automata

We complete the picture given in [6] about reducibility for nondeterministic weighted automata, by adding the relations with probabilistic automata. Fig. 1 summarizes the results.

### 3.1 Probabilistic Sup-automata

**Theorem 1.** ZSup *and* AsSup *are reducible to* DSup.

**Proof.** It is easy to see that ZSup-automata define the same language when interpreted as NSup-automata, and the same holds for AsSup and USup. The result then follows from [6, Theorem 9]. ■

### 3.2 Probabilistic LimAvg-automata

We consider the alphabet $\Sigma$ consisting of letters $a$ and $b$, i.e., $\Sigma = \{a, b\}$. We define the language $L_F$ of finitely many $a$'s, i.e., for an infinite word $w$ if $w$ consists of infinitely many $a$'s, then $L_F(w) = 0$, otherwise $L_F(w) = 1$. We also consider the language $L_I$ of words with infinitely many $a$'s (it is the complement of $L_F$). Many of our results would consider *Markov chains* and *closed recurrent states* in Markov chains. We define them below.

**Definition 1 (Markov chain and closed recurrent states).** *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 $(S, E)$.*

We will use the following two key properties of closed recurrent states.

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 $\alpha$, 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 $\alpha - \epsilon$.

The above properties are the basic properties of finite state Markov chains and closed recurrent states [**?**].

**Lemma 1.** *Let $A$ be a probabilistic weighted automata with alphabet $\Sigma = \{a, b\}$. Consider the Markov chain arising of $A$ on input $b^\omega$ (we refer to this as the $b$-Markov chain) and we use similar notation for the $a$-Markov chain. The following assertions hold:*

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

**Proof.** We present the proof in three parts.

1. Let $\beta$ 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 $k_\epsilon$ 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 $1-\epsilon$. If all closed recurrent sets in the $b$-Markov chain have expected limit-average value at least 1, then for all $\epsilon > 0$, there exists $l_\epsilon$ 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.
2. The proof is similar to the previous result.
3. The proof is also similar to the first result. The only difference is that we use a long enough sequence of $b^j$ such that with high probability a closed recurrent set in the $b$-Markov chain is reached and then stay long enough in the closed recurrent set to approach the expected sum of rewards to 0, and then present a long enough sequence of $a^j$ such that with high probability a closed recurrent set in the $a$-Markov chain is reached and then stay long enough in the closed recurrent set to approach the expected sum of rewards to 0. The calculation is similar to the first part of the proof.

Thus we obtain the desired result. ■

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

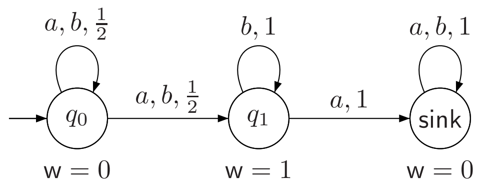


**Fig. 2.** A ZLavg for Lemma 2.

1. *The language can be expressed as a* NLavg*.*
2. *The language can be expressed as a* ZLavg*.*
3. *The language cannot be expressed as* AsLavg*.*

**Proof.** We present the three parts of the proof.

1. The result follows from the results of [6, Theorem 12] where the explicit construction of a NLavg to express $L_F$ is presented.
2. A ZLavg automaton $A$ to express $L_F$ is as follows (see Fig. 2):
   (a) *States and weight function.* The set of states of the automaton is $\{q_0, q_1, sink\}$, with $q_0$ as the starting state. The weight function $\mathsf{w}$ is as follows: $\mathsf{w}(q_0) = \mathsf{w}(sink) = 0$ and $\mathsf{w}(q_1) = 1$.
   (b) *Transition function.* The probabilistic transition function is as follows:
      (i) from $q_0$, given $a$ or $b$, the next states are $q_0, q_1$, each with probability $1/2$;
      (ii) from $q_1$ given $b$ the next state is $q_1$ with probability 1, and from $q_1$ given $a$ the next state is *sink* with probability 1; and
      (iii) from *sink* state the next state is *sink* with probability 1 on both $a$ and $b$. (it is an absorbing state).

   Given the automaton $A$ consider any word $w$ with infinitely many $a$'s then, the automata reaches sink state in finite time with probability 1, and hence $A(w) = 0$. For a word $w$ with finitely many $a$'s, let $k$ be the last position that an $a$ appears. Then with probability $1/2^k$, after $k$ steps, the automaton only visits the state $q_1$ and hence $A(w) = 1$. Hence there is a ZLavg for $L_F$.
3. We show that $L_F$ cannot be expressed as an AsLavg. Consider an AsLavg automaton $A$. Consider the Markov chain that arises from $A$ if the input is only $b$ (i.e., on $b^\omega$), we refer to it as the $b$-Markov chain. If there is a closed recurrent set $C$ that can be reached from the starting state (reached by any sequence of $a$ and $b$'s), then the limit-average reward (in probabilistic sense) in $C$ must be at least 1 (otherwise, if there is a closed recurrent set $C$ with limit-average reward less than 1, we can construct a finite word $w$ that with positive probability will reach $C$, and then follow $w$ by $b^\omega$ and we will have $A(w \cdot b^\omega) < 1$). Hence any closed recurrent set on the $b$-Markov chain has limit-average reward at least 1 and by Lemma 1 there exists $j$ such that the $A((b^j \cdot a)^\omega) > 0$. Hence it follows that $A$ cannot express $L_F$.

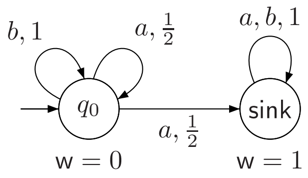


**Fig. 3.** An AsLAVG for Lemma 3.

Hence the result follows. ■

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

1. *The language cannot be expressed as an* NLAVG.
2. *The language cannot be expressed as a* ZLAVG.
3. *The language can be expressed as* AsLAVG.

**Proof.** We present the three parts of the proof.

1. It was shown in the proof of [6, Theorem 13] that NLAVG cannot express $L_I$.
2. We show that $L_I$ is not expressible by a ZLAVG. Consider a ZLAVG $A$ and consider the $b$-Markov chain arising from $A$ under the input $b^\omega$. All closed recurrent sets $C$ reachable from the starting state must have the limit-average value at most 0 (otherwise we can construct an word $w$ with finitely many $a$'s such that $A(w) > 0$). Since all closed recurrent set in the $b$-Markov chain has limit-average reward that is 0, using Lemma 1 we can construct a word $w = (b^j \cdot a)^\omega$, for a large enough $j$, such that $A(w) < 1$. Hence the result follows.
3. We now show that $L_I$ is expressible as an AsLAVG. The automaton $A$ is as follows (see Fig. 3):
   (a) *States and weight function.* The set of states are $\{q_0, sink\}$ with $q_0$ as the starting state. The weight function is as follows: $\mathsf{w}(q_0) = 0$ and $\mathsf{w}(sink) = 1$.
   (b) *Transition function.* The probabilistic transition function is as follows:
      (i) from $q_0$ given $b$ the next state is $q_0$ with probability 1;
      (ii) at $q_0$ given $a$ the next states are $q_0$ and $sink$ each with probability 1/2; (iii) the $sink$ state is an absorbing state.

   Consider a word $w$ with infinitely many $a$'s, then the probability of reaching the sink state is 1, and hence $A(w) = 1$. Consider a word $w$ with finitely many $a$'s, and let $k$ be the number of $a$'s, and then with probability $1/2^k$ the automaton always stay in $q_0$, and hence $A(w) = 0$.

Hence the result follows. ■

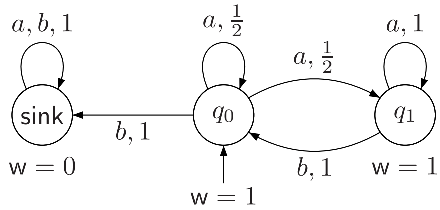


**Fig. 4.** A probabilistic automaton (ZLAVG, ZLSUP, or ZLINF) for Lemma 4.

**Lemma 4.** *There exists a language that can be expressed by* ZLAVG*,* ZLSUP *and* ZLINF*, but not by* NLAVG*,* NLSUP *or* NLINF*.*

**Proof.** Consider an automaton $A$ as follows (see Fig. 4):

1. *States and weight function.* The set of states are $\{p, q, \mathit{sink}\}$ with $p$ as the starting state. The weight function is as follows: $\mathsf{w}(p) = \mathsf{w}(q) = 1$ and $\mathsf{w}(\mathit{sink}) = 0$.
2. *Transition function.* The probabilistic transition is as follows:
   (i) from $p$ if the input letter is $a$, then the next states are $p$ and $q$ with probability $1/2$;
   (ii) from $p$ if the input letter is $b$, then the next state is *sink* with probability 1;
   (iii) from $q$, if the input letter is $b$, then the next state is $p$ with probability 1;
   (iv) from $q$, if the input letter is $a$, then the next state is $q$ with probability 1; and
   (v) the state *sink* is an absorbing state.

If we consider the automaton $A$, and interpret it as a ZLAVG, ZLSUP, or ZLINF, then it accepts the following language:

$$L_z = \{a^{k_1} b a^{k_2} b a^{k_3} b \dots \mid k_1, k_2, \dots \in \mathbb{N}_{\geq 1} \cdot \prod_{i=1}^{\infty} (1 - \frac{1}{2^{k_i}}) > 0\};$$

i.e., $A(w) = 1$ if $w \in L_z$ and $A(w) = 0$ if $w \notin L_z$: the above claim follows from the argument following Lemma 5 of [2]. We now show that $L_z$ cannot be expressed as NLAVG, NLSUP or NLINF. Consider a non-deterministic automaton $A$. Suppose there is a cycle $C$ in $A$ such that average of the rewards in $C$ is positive. If no such cycle exists, then clearly $A$ cannot express $L_z$ as there exists word for which $L_z(w) = 1$. Consider a cycle $C$ such that average of the rewards is positive, and let the cycle be formed by a finite word $w_C = a_0 a_1 \dots a_n$ and there must exist at least one index $0 \leq i \leq n$ such that $a_i = b$. Hence the word can be expressed as $w_C = a^{j_1} b a^{j_2} b \dots a^{j_k} b$, and hence there exists a finite word $w_R$ (that reaches the cycle) such that $A(w_R \cdot w_C^{\omega}) > 0$. This contradicts that $A$ is an automaton to express $L_z$ as $L_z(w_R \cdot w_C^\omega) = 0$. Simply exchanging the average reward of the cycle by the maximum reward (resp. minimum reward) shows that $L_z$ is not expressible by a NLsup (resp. NLinf). ■

**Theorem 2.** AsLavg *is incomparable in expressive power with* ZLavg *and* NLavg, *and* NLavg *cannot express all languages expressible by* ZLavg.

**Open question.** Whether NLavg is reducible to ZLavg or NLavg is incomparable to ZLavg (i.e., there is a language expressible by NLavg but not by a ZLavg) remains open.

### 3.3 Probabilistic LimInf-automata

**Lemma 5.** NLinf *is reducible to both* AsLinf *and* ZLinf.

**Proof.** It was shown in [6] that NLinf is reducible to DLinf. Since DLinf are special cases of AsLinf and ZLinf the result follows. ■

**Lemma 6.** *The language* $L_I$ *is expressible by an* AsLinf, *but cannot be expressed as a* NLinf *or a* ZLinf.

**Proof.** It was shown in [6] that the language $L_I$ is not expressible by NLinf. If we consider the automaton $A$ of Lemma 3 and interpret it as an AsLinf, then the automaton $A$ expresses the language $L_I$. The proof of the fact that ZLinf cannot express $L_I$ is similar to the the proof of Lemma 3 (part(2)) and instead of the average reward of the closed recurrent set $C$, we need to consider the minimum reward of the closed recurrent set $C$. ■

**Lemma 7.** ZLinf *is reducible to* AsLinf.

**Proof.** Let $A$ be a ZLinf and we construct a AsLinf $B$ such that $B$ is equivalent to $A$. Let $V$ be the set of weights that appear in $A$ and let $v_1$ be the least value in $V$. For each weight $v \in V$, consider the ZCW $A^v$ that is obtained from $A$ by considering all states with weight at least $v$ as accepting states. It follows from the results of [1] that ZCW is reducible to AsCW (it was shown in [1] that AsBW is reducible to ZBW and it follows easily that dually ZCW is reducible to AsCW). Let $D^v$ be an AsCW that is equivalent to $A^v$. We construct a ZLinf $B^v$ from $D^v$ by assigning weights $v$ to the accepting states of $D^v$ and the minimum weight $v_1$ to all other states. Consider a word $w$, and we consider the following cases.

1. If $A(w) = v$, then for all $v' \in V$ such that $v' \leq v$ we have $D^{v'}(w) = 1$, (i.e., the ZCW $A^{v'}$ and the AsCW $D^{v'}$ accepts $w$).

2. For $v \in V$, if $D^v(w) = 1$, then $A(w) \geq v$

It follows from above that $A = \max_{v \in V} B^v$. We will show later that AsLinf is closed under max (Lemma 17) and hence we can construct an AsLinf $B$ such that $B = \max_{v \in V} B^v$. Thus the result follows. ∎

**Theorem 3.** *We have the following strict inclusion*

$$\text{NLinf} \subsetneq \text{ZLinf} \subsetneq \text{AsLinf}$$

**Proof.** The fact that NLinf is reducible to ZLinf follows from Lemma 5, and the fact the ZLinf is not reducible to NLinf follows from Lemma 4. The fact that ZLinf is reducible to AsLinf follows from Lemma 7 and the fact that AsLinf is not reducible to ZLinf follows from Lemma 6. ∎

### 3.4 Probabilistic LimSup-automata

**Lemma 8.** NLsup *and* ZLsup *are not reducible to* AsLsup.

**Proof.** The language $L_F$ of finitely many $a$'s can be expressed as a non-deterministic Büchi automata, and hence as a NLsup. We will show that NLsup is reducible to ZLsup. It follows that $L_F$ is expressible as NLsup and ZLsup. The proof of the fact that AsLsup cannot express $L_F$ is similar to the the proof of Lemma 2 (part(3)) and instead of the average reward of the closed recurrent set $C$, we need to consider the maximum reward of the closed recurrent set $C$. ∎

**Deterministic in limit NLsup.** Consider an automaton $A$ that is a NLsup. Let $v_1 < v_2 < \ldots < v_k$ be the weights that appear in $A$. We call the automaton $A$ *deterministic in the limit* if for all states $s$ with weight greater than $v_1$, all states $t$ reachable from $s$ are deterministic.

**Lemma 9.** *For every* NLsup *$A$, there exists a* NLsup *$B$ that is deterministic in the limit and equivalent to $B$.*

**Proof.** From the results of [7] it follows that a NBW $A$ can be reduced to an equivalent NBW $B$ such that $B$ is deterministic in the limit. Let $A$ be a NLsup, and let $V$ be the set of weights that appear in $A$. and let $V = \{v_1, \ldots, v_k\}$ with $v_1 < v_2 < \cdots < v_k$. For each $v \in V$, consider the NBW $A_v$ whose (boolean) language is the set of words $w$ such that $L_A(w) \geq v$, by declaring to be accepting the states with weight at least $v$. Let $B_v$ be the deterministic in the limit NBW that is equivalent to $A_v$. The automaton $B$ that is deterministic in the limit and is equivalent to $A$ is obtained as the automaton that by initial non-determinism chooses between the $B_v$'s, for $v \in V$. ∎

**Lemma 10.** NLsup *is reducible to* ZLsup.

**Proof.** Given a NLsup $A$, consider the NLsup $B$ that is deterministic in the limit and equivalent to $B$. By assigning equal probabilities to all out-going transitions from a state we obtain a ZLsup $C$ that is equivalent to $B$ (and hence $A$). The result follows. ■

**Lemma 11.** AsLsup *is reducible to* ZLsup.

**Proof.** Consider a AsLsup $A$ and let the weights of $A$ be $v_1 < v_2 \ldots < v_l$. For $1 \leq i \leq l$ consider the AsBW obtained from $A$ with the set of state with reward at least $v_i$ as the Büchi states. It follows from the results of [1] that AsBW is reducible to ZBW. Let $B_i$ be the ZBW that is equivalent to $A_i$. Let $C_i$ be the automaton such that all Büchi states of $B_i$ is assigned weight $v_i$ and all other states are assigned $v_1$. Consider the automata $C$ that goes with equal probability to the starting states of $C_i$, for $1 \leq i \leq l$, and we interpret $C$ as a ZLsup. Consider a word $w$, and let $A(w) = v_j$ for some $1 \leq j \leq l$, i.e., given $w$, the set of states with reward at least $v_j$ is visited infinitely often with probability 1 in $A$. Hence the ZBW $B_i$ accepts $w$ with positive probability, and since $C$ chooses $C_i$ with positive probability, it follows that given $w$, in $C$ the weight $v_j$ is visited infinitely often with positive probability, i.e., $C(w) \geq v_j$. Moreover, given $w$, for all $v_k > v_l$, the set of states with weight at least $v_k$ is visited infinitely often with probability 0 in $A$. Hence for all $k > j$, the automata $B_k$ accepts $w$ with probability 0. Thus $C(w) < v_k$ for all $v_k > v_j$. Hence $C(w) = A(w)$ and thus AsLsup is reducible to ZLsup. ■

**Lemma 12.** AsLsup *is not reducible to* NLsup.

**Proof.** It follows from [1] that for $0 < \lambda < 1$ the following language $L_\lambda$ can be expressed by a AsBW and hence by AsLsup:

$$L_\lambda = \{a^{k_1}ba^{k_2}ba^{k_3}b \ldots \mid k_1, k_2, \cdots \in \mathbb{N}_{\geq 1}. \prod_{i=1}^{\infty}(1 - \lambda^{k_i}) > 0\}.$$

It follows from argument similar to Lemma 4 that there exists $0 < \lambda < 1$ such that $L_\lambda$ cannot be expressed by a NLsup. Hence the result follows. ■

**Theorem 4.** AsLsup *and* NLsup *are incomparable in expressive power, and* ZLsup *is more expressive than* AsLsup *and* NLsup.

**Lemma 13.** ZCW *is reducible to* ZBW.

**Proof.** Let $A = \langle Q, q_I, \Sigma, \delta, C\rangle$ be a ZCW with the set $C \subseteq Q$ of accepting states. We construct a ZBW $\overline{A}$ as follows:

1. The set of states is $Q \cup \overline{Q}$ where $\overline{Q} = \{\overline{q} \mid q \in Q\}$ is a copy of the states in $Q$;
2. $q_I$ is the initial state;
3. The transition function is as follows, for all $\sigma \in \Sigma$:
   (a) for all states $q, q' \in Q$, we have $\overline{\delta}(q, \sigma, q') = \overline{\delta}(q, \sigma, \overline{q'}) = \frac{1}{2} \cdot \delta(q, \sigma, q')$, i.e., the state $q'$ and its copy $\overline{q'}$ are reached with half of the original transition probability;
   (b) the states $\overline{q} \in \overline{Q}$ such that $q \notin C$ are abosrbing states (i.e., $\overline{\delta}(\overline{q}, \sigma, \overline{q}) = 1$);
   (c) for all states $q \in C$ and $q' \in Q$, we have $\overline{\delta}(\overline{q}, \sigma, \overline{q'}) = \delta(q, \sigma, q')$, i.e., the transition function in the copy automaton follows that of $A$ for states that are copy of the accepting states.
4. The set of accepting states is $\overline{C} = \{\overline{q} \in \overline{Q} \mid q \in C\}$.

We now show that the language of the ZCW $A$ and the language of ZBW $\overline{A}$ coincides. Consider a word $w$ such that $A(w) = 1$. Let $\alpha$ be the probability that given the word $w$ evenutally always states in $C$ are visited in $A$, and since $A(w) = 1$ we have $\alpha > 0$. In other words, as limit $k$ tends to $\infty$, the probability that after $k$ steps only states in $C$ are visited is $\alpha$. Hence there exists $k_0$ such that the probability that after $k_0$ steps only states in $C$ are visited is at least $\frac{\alpha}{2}$. In the automaton $\overline{A}$, the probability to reach states of $\overline{Q}$ after $k_0$ steps has probability $p = 1 - \frac{1}{2^{k_0}} > 0$. Hence with positive probability (at least $p \cdot \frac{\alpha}{2}$) the automaton visits infinitely often the states of $\overline{C}$, and hence $\overline{A}(w) = 1$. Observe that since every state in $\overline{Q} \setminus \overline{C}$ is absorbing and non-accepting), it follows that if we consider an accepting run $\overline{A}$, then the run must eventually always visits states in $\overline{C}$ (i.e., the copy of the accepting states $C$). Hence it follows that for a given word $w$, if $\overline{A}(w) = 1$, then with positive probability eventually always states in $C$ are visited in $A$. Thus $A(w) = 1$, and the result follows. ∎

**Lemma 14.** ZLinf *is reducible to* ZLsup*, and* AsLsup *is reducible to* AsLinf.

**Proof.** We present the proof that ZLinf is reducible to ZLsup, the other proof being similar. Let $A$ be a ZLinf, and let $V$ be the set of weights that appear in $A$. For each $v \in V$, it is easy to construct a ZCW $A_v$ whose (boolean) language is the set of words $w$ such that $L_A(w) \geq v$, by declaring to be accepting the states with weight at least $v$. We then construct for each $v \in V$ a ZBW $\overline{A}_v$ that accepts the language of $A_v$ (such a ZBW can be constructed by Lemma 13). Finally, assuming that $V = \{v_1, \ldots, v_n\}$ with $v_1 < v_2 < \cdots < v_n$, we construct the ZLsup $B_i$ for $i = 1, 2, \ldots, n$ where $B_i$ is obtained from $\overline{A}_{v_i}$ by assigning weight $v_i$ to each accepting states, and $v_1$ to all the other states. The ZLsup that expresses the language of $A$ is $\max_{i=1,2\ldots,n} B_i$ and since ZLsup is closed under max (see Lemma 15), the result follows. ∎

### 3.5 Probabilistic Disc-automata

**Theorem 5.** *a)* NDisc *and* ZDisc *are reducible to each other; b)* UDisc *and* AsDisc *are reducible to each other.*

**Proof.** *a)* We first prove that NDisc is reducible to ZDisc. Let $A = \langle Q, \rho_I, \Sigma, \delta_A, \gamma \rangle$ be a NDisc, and let $v_{\min}, v_{\max}$ be its minimal and maximal weights respectively. Consider the ZDisc $B = \langle Q, \rho_I, \Sigma, \delta_B, \gamma \rangle$ where $\delta_B(q, \sigma)$ is the uniform distribution over the set of states $q'$ such that $(q, \sigma, \{q'\}) \in \delta_A$. Let $r = q_0\sigma_1q_1\sigma_2\ldots$ be a run of $A$ (over $w = \sigma_1\sigma_2\ldots$) with value $\eta$. 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 $r_n = q_0\sigma_1q_1\sigma_2\ldots\sigma_nq_n$. 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 ZDisc is reducible to NDisc. Given a ZDisc $B = \langle Q, \rho_I, \Sigma, \delta_B, \gamma \rangle$, 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)* It is easy to see that the complement of the quantitative language defined by a UDisc (resp. AsDisc) can be defined by a NDisc (resp. ZDisc). Then, the result follows from Part *a)* (essentially, given a UDisc, we obtain easily an NDisc for the complement, then an equivalent ZDisc, and finally a AsDisc for the complement of the complement, i.e., the original quantitative language). ∎

Note that a by-product of this proof is that the language of a ZDisc does not depend on the precise values of the probabilities, but only on whether they are positive or not.

## 4 Closure Properties of Probabilistic Weighted Automata

### 4.1 Closure under max and min

**Lemma 15 (Closure by initial non-determinism).** ZLsup*,* ZLinf *and* ZLavg *is closed under* max*; and* AsLsup*,* AsLinf *and* AsLavg *is closed under* min*.*

**Proof.** Given two automata $A_1$ and $A_2$ consider the automata $A$ obtained by initial non-deterministic choice of $A_1$ and $A_2$. Formally, let $q_1$ and $q_2$ be the initial states of $A_1$ and $A_2$, respectively, then in $A$ we add an initial state $q_0$ and the transition from $q_0$ is as follows: for $\sigma \in \Sigma$, consider the set $Q_\sigma = \{q \in Q_1 \cup Q_2 \mid$

| | | max | min | comp. | sum | emptiness | universality |
|---|---|---|---|---|---|---|---|
| $>0$ | ZSup | $\surd$ | $\surd$ | $\times$ | $\surd$ | $\surd$ | $\surd$ |
| | ZLsup | $\surd$ | $\surd$ | $\surd$ | $\surd$ | $\times$ | $\times$ |
| | ZLinf | $\surd$ | $\surd$ | $\times$ | $\surd$ | $\surd$(?) | $\surd$ |
| | ZLavg | $\surd$ | $\times$ | $\times$ | | | |
| | ZDisc | $\surd$ | $\times$ | $\times$ | $\surd$ | $\surd$ | (1) |
| almost-sure | AsSup | $\surd$ | $\surd$ | $\times$ | $\surd$ | $\surd$ | $\surd$ |
| | AsLsup | $\surd$ | $\surd$ | $\times$ | $\surd$ | $\surd$ | $\surd$(?) |
| | AsLinf | $\surd$ | $\surd$ | $\surd$ | $\surd$ | $\times$ | $\times$ |
| | AsLavg | $\times$ | $\surd$ | $\times$ | $\times$ | | |
| | AsDisc | $\times$ | $\surd$ | $\times$ | $\surd$ | (1) | $\surd$ |

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

**Table 1.** Closure properties and emptiness problems.

$\delta_1(q_1,\sigma)(q) > 0$ or $\delta_2(q_2,\sigma)(q) > 0\}$. From $q_0$, for input letter $\sigma$, the successors are from $Q_\sigma$ each with probability $1/|Q_\sigma|$. If $A_1$ and $A_2$ are ZLsup (resp. ZLinf, ZLavg), then $A$ is a ZLsup (resp. ZLinf, ZLavg) such that $A = \max\{A_1, A_2\}$. Similarly, if $A_1$ and $A_2$ are AsLsup (resp. AsLinf, AsLavg), then $A$ is a AsLsup (resp. AsLinf, AsLavg) such that $A = \min\{A_1, A_2\}$. ■

**Lemma 16 (Closure by synchronized product).** AsLsup *is closed under* max *and* ZLinf *is closed under* min.

**Proof.** We present the proof that AsLsup is closed under max. Let $A_1$ and $A_2$ be two probabilistic weighted automata with weight function $\mathsf{w}_1$ and $\mathsf{w}_2$, respectively. Let $A$ be the usual synchronized product of $A_1$ and $A_2$ with weight function $\mathsf{w}$ such that $\mathsf{w}((s_1, s_2)) = \max\{\mathsf{w}_1(s_1), \mathsf{w}_2(s_2)\}$. Given a path $\pi = ((s_0^1, s_0^2), (s_1^1, s_1^2), \ldots)$ in $A$ we denote by $\pi \upharpoonright 1$ the path in $A_1$ that is the projection of the first component of $\pi$ and we use similar notation for $\pi \upharpoonright 2$. Consider a word $w$, let $\max\{A_1(w), A_2(w)\} = v$. We consider the following two cases to show that $A(w) = v$.

1. W.l.o.g. let the maximum be achieved by $A_1$, i.e., $A_1(w) = v$. Let $B_i^v$ be the set of states $s_i$ in $A_i$ such that weight of $s_i$ is at least $v$. Since $A_1(w) = v$, given the word $w$, in $A_1$ the event Büchi$(B_1^v)$ holds with probability 1. Consider the following set of paths in $A$

$$\Pi^v = \{\pi \mid (\pi \upharpoonright 1) \in \text{Büchi}(B_1^v)\}.$$

   Since given $w$, the event Büchi$(B_1^v)$ holds with probability 1 in $A_1$, it follows that given $w$, the event $\Pi^v$ holds with probability 1 in $A$. The $\mathsf{w}$ function ensures that every path $\pi \in \Pi^v$ visits weights of value at least $v$ infinitely often. Hence $A(w) \geq v$.

2. Consider a weight value $v' > v$. Let $C_i^v$ be the set of states $s_i$ in $A_i$ such that the weight of $s_i$ is less than $v'$. Given the word $w$, since $A_i(w) < v'$, it follows that probability of the event coBüchi($C_i^v$) in $A_i$, given the word $w$, is positive. Hence given the word $w$, the probability of the event coBüchi($C_1^v \times C_2^v$)) is positive in $A$. It follows that $A(w) < v'$.

The result follows. If $A_1$ and $A_2$ are ZLINF, and in $A$ we assign weights such that every state in $A$ has the minimum weight of its component states, and we consider $A$ as a ZLINF, then $A = \min\{A_1, A_2\}$. The proof is similar to the result for AsLSUP. ■

**Lemma 17.** ZLSUP *is closed under* min *and* AsLINF *is closed under* max.

**Proof.** Let $A_1$ and $A_2$ be two ZLSUP. We construct a ZLSUP $A$ such that $A = \min\{A_1, A_2\}$. Let $V_i$ be the set of weights that appear in $A_i$ (for $i = 1, 2$), and let $V = V_1 \cup V_2$ and let $v_1$ be the least value in $V$. For each weight $v \in V_1 \cup V_2 = \{v_1, \ldots, v_k\}$, consider the ZBW $A_i^v$ that is obtained from $A_i$ by considering all states with weight at least $v$ as accepting states. Since ZBW is closed under intersection(by the results of [2]), we can construct a ZBW $A_{12}^v$ that is the intersection of $A_1^v$ and $A_2^v$, i.e. $A_{12}^v = A_1^v \cap A_2^v$. We construct a ZLSUP $B_{12}^v$ from $A_{12}^v$ by assigning weights $v$ to the accepting states of $A_{12}^v$ and the minimum weight $v_1$ to all other states. Consider a word $w$, and we consider the following cases.

1. If $\min\{A_1(w), A_2(w)\} = v$, then for all $v' \in V$ such that $v' \leq v$ we have $A_{12}^{v'}(w) = 1$, (i.e., the ZBW $A_{12}^{v'}$ accepts $w$).
2. If $A_{12}^v(w) = 1$, then $A_1(w) \geq v$ and $A_2(w) \geq v$, i.e., $\min\{A_1(w), A_2(w)\} \geq v$.

It follows from above that $\min\{A_1, A_2\} = \max_{v \in V} B_{12}^v$. Since ZLSUP is closed under max (by initial non-determinism), it follows that ZLSUP is closed under min. The proof of closure of AsLINF under max is similar. ■

**Lemma 18.** *Consider the alphabet $\Sigma = \{a, b\}$, and consider the languages $L_a$ and $L_b$ that assigns the long-run average number of a's and b's, respectively. Then the following assertions hold.*

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

**Proof.** To obtain a contradiction, assume that there exists a ZLAVG $A$ (for either $L_m$ or $L^*$). We first claim that if we consider the $a$-Markov or the $b$-Markov chain of $A$, then there must be either an $a$-closed recurrent set or a $b$-closed recurrent set $C$ that is reachable in $A$ such that the expected sum of the weights in $C$ is positive. Otherwise, if for all $a$-closed recurrent sets and $b$-closed recurrent sets we have that the expected sum of the weights is zero or negative, then we fool the automaton as follows. By Lemma 1, it follows that there exists a $j$ such that $A((a^j \cdot b^j)^\omega) < 1/2$, however, $L_m(w) = L^*(w) = \frac{1}{2}$, i.e., we have a contradiction. W.l.o.g., we assume that there is an $a$-closed recurrent set $C$ such that expected sum of weights of $C$ is positive. Then we present the following word $w$: a finite word $w_C$ to reach the cycle $C$, followed by $a^\omega$; the answer of the automaton is positive, *i.e.*, $L_A(w) > 0$, while $L_m(w) = L^*(w) = 0$. Hence the result follows. ■

**Lemma 19.** ZLavg *is not closed under* min *and* AsLavg *is not closed under* max.

**Proof.** The result for ZLavg follows from Lemma 18. We now show that AsLavg is not closed under max. Consider the alphabet $\Sigma = \{a, b\}$ and the quantitative languages $L_a$ and $L_b$ that assign the value of long-run average number of $a$'s and $b$'s, respectively. There exists DLavg (and hence AsLavg) for $L_a$ and $L_b$. We show that $L_m = \max(L_a, L_b)$ cannot be expressed by an AsLavg. By contradiction, assume that $A$ is an AsLavg with set of states $Q$ that defines $L_m$. Consider any $a$-closed recurrent $C$ in $A$. 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 $C$, 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 $j_\epsilon$ such that the following properties hold:

1. from any state of $A$, given the word $a^{j_\epsilon}$ with probability $1 - \epsilon$ an $a$-closed recurrent set is reached;
2. once an $a$-closed recurrent set is reached, given the word $a^{j_\epsilon}$, 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 $\epsilon$ 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 $a^\omega$).

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 $j_\epsilon$ 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 $\epsilon$. 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

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

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

### 4.2 Closure under complement

**Lemma 20.** AsLsup *and* ZLinf *are not closed under complement.*

*Proof.* It follows from Lemma 8 that the language $L_F$ of finitely many $a$'s is not expressible by an AsLsup, whereas the complement $L_I$ of infinitely many $a$'s is expressible as a DBW and hence as a AsLsup. It follows from Lemma 6 that language $L_I$ is not expressible as an ZLinf, whereas its complement $L_F$ is expressible by a DCW and hence a ZLinf. ∎

**Lemma 21.** ZLsup *and* AsLinf *are closed under complement.*

**Proof.** We first present the proof for ZLsup. Let $A$ be a ZLsup, and let $V$ be the set of weights that appear in $A$. For each $v \in V$, it is easy to construct a ZBW $A_v$ whose (boolean) language is the set of words $w$ such that $L_A(w) \geq v$, by declaring to be accepting the states with weight at least $v$. We then construct for each $v \in V$ a ZBW $\bar{A}_v$ (with accepting states) that accepts the (boolean) complement of the language accepted by $A_v$ (such a ZBW can be constructed since ZBW is closed under complementation by the results of [1]). Finally, assuming that $V = \{v_1, \ldots, v_n\}$ with $v_1 < v_2 < \cdots < v_n$, we construct the ZLsup $B_i$ for $i = 2, \ldots, n$ where $B_i$ is obtained from $\bar{A}_{v_i}$ by assigning weight $-v_{i-1}$ to each accepting states, and $-v_n$ to all the other states. The complement of $L_A$ is then $\max\{L_{B_2}, \ldots, L_{B_n}\}$ which is accepted by a ZLsup (since ZLsup is closed under max). The result for AsLinf is similar and it uses the closure of AsCW under complementation which can be easily proved from the closure under complementation of ZBW. ∎

**Lemma 22.** ZLavg *and* AsLavg *are not closed under complement.*

**Proof.** The fact that ZLavg is not closed under complement follows from Lemma 18. We now show that AsLavg is not closed under complement. Consider the DLavg $A$ 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 AsLavg $B$ 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 $n_w$ such that the average number of $a$'s in $w.b^{n_w}$ is at most $\epsilon$, and there exists a number $m_w$ such that $B(w.a^{m_w}) \leq \epsilon$ (since $B(w.a^\omega) = 0$). Hence, we can construct a word $w = b^{n_1} a^{m_1} b^{n_2} a^{m_2} \dots$ such that $A(w) \leq \epsilon$ and $B(w) \leq \epsilon$. Since $B = 1 - A$, this implies that $1 \leq 2\epsilon$, a contradiction. ■

### 4.3 Closure under sum

**Lemma 23.** ZLsup *and* AsLsup *are closed under sum.*

**Proof.** Given two ZLsup (resp. AsLsup) $A_1$ and $A_2$, we construct a ZLsup (resp. AsLsup) $A$ for the sum of their languages as follows. For a pair $(v_1, v_2)$ of weights ($v_i$ in $A_i$, for $i = 1, 2$), consider a copy of the synchronized product of $A_1$ and $A_2$. We attach a bit $b$ whose range is $\{1, 2\}$ to each state to remember that we expect $A_b$ to visit the guessed weight $v_b$. Whenever this occurs, the bit $b$ is set to $3 - b$, and the weight of the state is $v_1 + v_2$. All other states (*i.e.* when $b$ is unchanged) have weight $\min\{v_1 + v_2 \mid v_1 \in V_1 \wedge v_2 \in V_2\}$. Let the automata constructed be $A_{(v_1,v_2)}$. Then $A = \max_{(v_1,v_2)} A_{(v_1,v_2)}$. Since ZLsup (resp. AsLsup) is closed under max the result follows. ■

**Lemma 24.** ZLinf *and* AsLinf *are closed under sum.*

**Proof.** Given two ZLinf (resp. AsLinf) $A_1$ and $A_2$, we construct a ZLinf (resp. AsLinf) $A$ for the sum of their languages as follows. For $i = 1, 2$, let $V_i$ be the set of weights that appear in $A_i$. Let $v_{\min} = \min\{v_1 + v_2 \mid v_1 \in V_1 \wedge v_2 \in V_2\}$. For $v_1 \in V_1$ and $v_2 \in V_2$, for $i = 1, 2$, consider the ZCW (resp. AsCW) $A_{v_i}$ obtained from $A_i$ by making all states with weights at least $v_i$ as accepting states. Let $A_{(v_1,v_2)}$ be the ZCW (resp. AsCW) such that $A_{(v_1,v_2)} = A_{v_1} \cap A_{v_2}$: such an ZCW (resp. AsCW) exists since ZCW (resp. AsCW) is closed under intersection. In other words, for a word $w$ we have $A_{(v_1,v_2)}(w) = 1$ iff $A_1(w) \geq v_1$ and $A_2(w) \geq v_2$. Let $\overline{A}_{(v_1,v_2)}$ be the ZLinf (resp. AsLinf) obtained from $A_{(v_1,v_2)}$ by assigning weight $v_1 + v_2$ to all accepting states and weight $v_{\min}$ to all other states. Then the automaton for the sum of $A_1$ and $A_2$ (denoted as $A_1 + A_2$) is $\max_{(v_1,v_2) \in V_1 \times V_2} \overline{A}_{(v_1,v_2)}$. Since ZLinf (resp. AsLinf) is closed under max the result follows. ■

**Lemma 25.** AsLavg *is not closed under sum.*

**Proof.** Consider the alphabet $\Sigma = \{a, b\}$, and consider the DLAVG-definable languages $L_a$ and $L_b$ that assigns to each word $w$ the long-run average number of $a$'s and $b$'s in $w$ respectively. Let $L_+ = L_a + L_b$. We show that $L_+$ is not expressible by ASLAVG. Assume towards contradiction that $L_+$ is defined by an ASLAVG $A$ with set of states $Q$ (we assume w.l.o.g that every state in $Q$ is reachable). Let $\beta > 1$ be greater than the maximum absolute value of the weights in $A$.

First, we claim that from every state $q \in Q$, if we consider the automaton $A_q$ with $q$ as starting state then $A_q(a^\omega) = 1$: this follows since if we consider a finite word $w_q$ to reach $q$, then $L_+(w_q \cdot a^\omega) = 1$ and hence $A(w_q \cdot a^\omega) = 1$. It follows that from any state $q$, as $k$ tends to $\infty$, the expected average of the weights converges almost-surely to 1. This implies if we consider the $a$-Markov chain arising from $A$, then from any state $q$, for all closed recurrent set $C$ of states reachable from $q$, the expected average of the weights of $C$ is 1. Hence for every $\gamma > 0$ there exists a natural number $k_0^\gamma$ such that from any state $q$, for all $k > k_0^\gamma$ given the word $a^k$ the expected average of the weights is at least $\frac{1}{2}$ with probability $1 - \gamma$, and for the first $k_0^\gamma$ steps the expected average of the weights is at least $-\beta$. The same result holds if we consider as input a sequence of $b$'s instead of $a$'s.

Consider the word $w$ generated inductively by the following procedure: (a) $w_0$ is the empty word; (b) we generate $w_{i+1}$ from $w_i$ as follows: $(i)$ the sequence of letters added to $w_i$ to obtain $w_{i+1}$ is at least $i$; $(ii)$ first we generate a long enough sequence $w'_{i+1}$ of $a$'s after $w_i$ such that the average number of $b$'s in $w_i \cdot w'_{i+1}$ falls below $\frac{1}{i}$; $(iii)$ then generate a long enough sequence $w''_{i+1}$ of $b$'s such that the average number of $a$'s in $w_i \cdot w'_{i+1} \cdot w''_{i+1}$ falls below $\frac{1}{i}$; $(iv)$ the word $w_{i+1} = w_i \cdot w'_{i+1} \cdot w''_{i+1}$. The word $w$ is the limit of these sequences. For $\gamma > 0$, consider $i \geq 6 \cdot k_0^\gamma \cdot \beta$ (where $k_0^\gamma$ satisfies the properties described above for $\gamma$). By construction for $i > 6 \cdot k_0^\gamma \cdot \beta$, the length of $w_i$ is at least $6 \cdot k_0 \cdot \beta$, and hence it follows that in the segment constructed between $w_i$ and $w_{i+1}$, for all $|w_i| \leq \ell \leq |w_{i+1}|$ with probability at least $1 - \gamma$ the expected average of the weights is at least

$$\frac{\frac{\ell - k_0^\gamma}{2} - k_0^\gamma \cdot \beta}{\ell} \geq \frac{1}{2} - \frac{2 \cdot k_0^\gamma \cdot \beta}{\ell} \geq \frac{1}{2} - \frac{1}{3} \geq \frac{1}{6}.$$

Hence for all $\gamma > 0$, the expected average of the weights is at least $\frac{1}{6}$ with probability at least $1 - \gamma$. Since this holds for all $\gamma > 0$, it follows that the expected average of the weights is at least $\frac{1}{6}$ almost-surely, (i.e., $A(w) \geq \frac{1}{6}$). We have $L_a(w) = L_b(w) = 0$ and thus $L_+(w) = 0$, while $A(w) \geq \frac{1}{6}$. Thus we have a contradiction. ■

**Lemma 26.** ZDISC *and* ASDISC *are closed under sum.*

**Proof.** The result for ZDISC follows from Theorem 5 and **??** and the fact that NDISC and UDISC are closed under sum (which is easy to prove using a

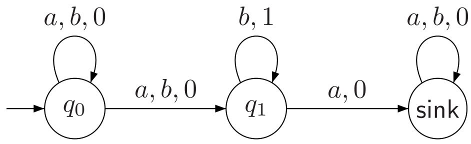


**Fig. 5.** A nondeterministic limit-average automaton.

synchronized product of automata where the weight of a joint transition is the sum of the weights of the corresponding transitions. ■

## 5 Comparison of finite max, deterministic in the limit and non-deterministic automata

**LimSup automata.** For LimSup automata, (a) since DLsup is closed under max it follows the expressive power of deterministic automata and finite max of deterministic automata is the same; (b) the expressive power of non-deterministic and the deterministic in the limit is the same; and (c) deterministic in the limit automata is more expressive than finite max of deterministic automata.

**Lemma 27.** *For limit-average automata the following assertions hold:*

1. *Finite* max *of deterministic automata is more expressive that deterministic limit-average automata.*
2. *Deterministic in the limit non-deterministic limit-average automata is more expressive that finite* max *of deterministic limit-average automata.*

**Proof.** We present the proof in two parts.

1. It follows from the results of [6] that DLavg is not closed under max, and hence the result follows.
2. Consider the language $L_F$ of finitely many $a$'s. It is also easy to see that the NLavg shown in Fig. 5 defines $L_F$, and also the NLavg shown is deterministic in the limit. We show that $L_F$ cannot be defined by finite max of DLavg's to prove the desired claim. Consider $\ell$ deterministic DLavg $A_1, A_2, \ldots, A_\ell$, and let $A = \max\{A_1, A_2, \ldots, A_\ell\}$. Towards contradiction we assume that $A$ defines the language $L_F$. We assume without loss of generality that every state $q_i \in Q_i$ is reachable from the starting $q_i^0$ for $A_i$ by a finite word $w_q^i$. For every finite word $w_f$, we have $A(w_f \cdot b^\omega) = 1$, and hence it follows that for every finite word $w_f$, there must exist a component automata $A_i$ such from the state $q_i$ reachable from $q_i^0$ by $w_f$, the $b$-cycle $C_i$ reachable from $q_i$ (the $b$-cycle is the cycle that can be executed with $b$'s

only) satisfy the condition that sum of the weights of the cycle is at least $|C_i|$. Let $\beta$ be the maximum absolute value of the weights that appear in any $A_i$, and let $|Q_M|$ be the maximum number of states in any automata. Let $j = \lceil 2 \cdot (2 \cdot |Q_M| + 2 \cdot |Q_M| \cdot \beta + \beta + 1) \rceil$ and consider the word $w = (b^j \cdot a)^\omega$. Since for every finite word $w_f$, there exist an automaton $A_i$ such that the $b$-cycle reachable $C_i$ from the state after the word $w_f$ has average weight at least 1, it follows that there exist an automaton $A_i$ such that the $b$-cycle $C_i$ executed infinitely often on the word $(b^j \cdot a)^\omega$ has average weight at least 1. A lower bound on the average of the weights in the unique run of $A_i$ over $(b^j \cdot a)$ is as follows: consider the set of states that appear infinitely often in the run, then it can have a prefix of length at most $Q_i$ whose sum of weights is at least $-|Q_i| \cdot \beta$, then it goes through $b$-cycle $C_i$ for at least $j - 2 \cdot |Q_i|$ steps with sum of weights at least $(j - 2 \cdot |Q_i|)$ (since the $b$-cycle $C_i$ have average weight at least 1), then again a prefix of length at most $|Q_i|$ without completing the cycle (with sum of weights at least $-|Q_i| \cdot \beta$), and then weight for $a$ is at least $-\beta$. Hence the average is at least

$$\frac{(j - 2 \cdot |Q_i|) - 2 \cdot |Q_i| \cdot \beta - \beta}{j+1} \geq 1 - \frac{2 \cdot |Q_i| + 2 \cdot |Q_i| \cdot \beta + \beta + 1}{j} \geq 1 - \frac{1}{2} = \frac{1}{2};$$

we used above by choice of $j$ we have $\frac{2 \cdot |Q_i| + 2 \cdot |Q_i| \cdot \beta + \beta + 1}{j} \leq \frac{1}{2}$ since $|Q_i| \leq |Q_M|$. Hence we have $A((b^j \cdot a)^\omega) \geq \frac{1}{2}$ contradicting that $A$ defines the language $L_F$.

■

## 6 Decision problems

We conclude the paper with (un)decidability results for classical decision problems about quantitative languages. 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 6.** *The emptiness and universality problems for* ZSup *and* AsSup *are decidable.*

*Proof.* By Theorem 1, these problems reduce to emptiness of DSup which is decidable ([6, Theorem 1]).

The following theorems are trivial corollaries of [1, Theorem 2].

**Theorem 7.** *The emptiness problem for* ZLsup *and the universality problem for* AsLsup *are undecidable.*

It is easy to obtain the following result as a straightforward generalization of [1, Theorem 6].

**Theorem 8.** *The emptiness problem for* AsLsup *and the universality problem for* ZLinf *are decidable.*

The following result is a particular case of [1, Corollary 3].

**Theorem 9.** *The emptiness problem for* AsLinf *and the universality problem fotr* ZLsup *are undecidable.*

Finally, by Theorem 5 and the decidability of emptiness for NDisc, we get the following result.

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

Note that by Theorem 5, the universality problem for NDisc (which is not know to be decidable) can be reduced to the universality problem for ZDisc and to the emptiness problem for AsDisc.

## References


1. Christel Baier, Nathalie Bertrand, and Marcus Größer. On decision problems for probabilistic Büchi automata. In *Proc. of FoSSaCS: Foundations of Software Science and Computational Structures*, LNCS 4962, pages 287–301. Springer, 2008.
2. Christel Baier and Marcus Größer. Recognizing omega-regular languages with probabilistic automata. In *Proc. of LICS: Logic in Computer Science*, pages 137–146. IEEE, 2005.
3. A. Chakrabarti, K. Chatterjee, T. A. Henzinger, O. Kupferman, and R. Majumdar. Verifying quantitative properties using bound functions. In *CHARME*, LNCS 3725, pages 50–64. Springer, 2005.
4. A. Chakrabarti, L. de Alfaro, T. A. Henzinger, and M. Stoelinga. Resource interfaces. In *EMSOFT*, LNCS 2855, pages 117–133. Springer, 2003.
5. K. Chatterjee, L. Doyen, and T. A. Henzinger. Expressiveness and closure properties for quantitative languages. Technical Report MTC-REPORT-2008-004, http://infoscience.epfl.ch/record/116423, EPFL, 2008.
6. K. Chatterjee, L. Doyen, and T. A. Henzinger. Quantitative languages. In *Proc. of CSL: Computer Science Logic*, LNCS 5213, pages 385–400. Springer-Verlag, 2008.
7. Costas Courcoubetis and Mihalis Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.
8. Karel Culik II and Juhani Karhumäki. Finite automata computing real functions. *SIAM J. Comput.*, 23(4):789–814, 1994.
9. Zoltán Ésik and Werner Kuich. An algebraic generalization of omega-regular languages. In *MFCS*, LNCS 3153, pages 648–659. Springer, 2004.
10. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games*, Lecture Notes in Computer Science 2500. Springer-Verlag, 2002.
11. Moshe Y. Vardi. An automata-theoretic approach to linear temporal logic. In *Proc. of Banff Higher Order Workshop*, Lecture Notes in Computer Science 1043, pages 238–266. Springer, 1995.