Probabilistic Automata and Probabilistic Logic

Thomas Weidner

Institut für Informatik, Universität Leipzig, D-04009 Leipzig, Germany
weidner@informatik.uni-leipzig.de

Abstract. We present a monadic second-order logic which is extended by an expected value operator and show that this logic is expressively equivalent to probabilistic automata for both finite and infinite words. We give possible syntax extensions and an embedding of our probabilistic logic into weighted MSO logic. We further derive decidability results which are based on corresponding results for probabilistic automata.

1 Introduction

Probabilistic automata, introduced already by Rabin [19], form a flourishing field. Their applications range from speech recognition [20] over prediction of climate parameters [17] to randomized distributed systems [11]. For surveys of theoretical results see the books [18,5]. Recently, the concept of probabilistic automata has been transfered to infinite words by Baier and Grösser [1]. This concept led to further research [2,7,8,9,10,22].

Though probabilistic automata admit a natural quantitative behavior, namely the acceptance probability of each word, the main research interest has been towards qualitative properties (for instance the language of all words with positive acceptance probability). We consider the behavior of a probabilistic automaton as function mapping finite or infinite words to a probability value. In spite of the paramount success of Büchi's [4] and Elgot's [14] characterizations of recognizable languages by MSO logic, no logic characterization of the behavior of probabilistic automata has been found yet.

We solve this problem by defining a probabilistic extension of MSO logic. Our *probabilistic MSO (PMSO)* logic is obtained from classical MSO logic by adding a second-order expected value operator $\mathbb{E}_p X$. In the scope of this operator, formulas $x \in X$ are considered to be true with probability p. The semantics of the expected value operator is then defined as the expected value over all sets. We illustrate our logic by an example of a communication device with probabilistic behavior, which can be modeled in PMSO.

In our main result, we establish the desired coincidence of behaviors of probabilistic automata and semantics of probabilistic MSO sentences. Our proof also yields a characterization of probabilistically recognizable word functions in terms of classical recognizable languages and Bernoulli measures. We show that every PMSO formula admits a prenex normal form, which is similar to existential MSO. Furthermore we give possible syntax extensions which do not alter the expressiveness of PMSO. Weighted MSO is another quantitative extension of MSO logic.

B. Rovan, V. Sassone, and P. Widmayer (Eds.): MFCS 2012, LNCS 7464, pp. 813–824, 2012.

As shown in [12], a restricted form of weighted MSO is expressively equivalent to weighted automata for finite and infinite words. For finite words, probabilistic automata can be viewed as special case of weighted automata. We give a direct embedding of PMSO into the restricted fragment of weighted MSO, thus obtaining PMSO as a special case of weighted MSO.

There are many decidability results already known for probabilistic automata on finite [15] and infinite words [2,9,10]. By the expressive equivalence of probabilistic automata and PMSO, we obtain these results also for PMSO. For instance, it is decidable for a given formula whether there is a finite word with positive or almost sure semantics. On the downside, it is undecidable for a given formula whether there is a finite word with semantics greater than some non-trivial cut point, or if there is an infinite word with semantics equal to one.

2 Bernoulli Measures and Probabilistic Automata

For the rest of this work let Σ be a finite alphabet. We let Σ^+ be the set of all finite, non-empty words over Σ, and Σ^ω the set of all infinite words over Σ. If $w \in \Sigma^+$ is a finite word we write $|w|$ for its length. If $w \in \Sigma^\omega$ we let $|w| = \omega$. For convenience we write Σ^∞ if either Σ^+ or Σ^ω can be used. For a word $w \in \Sigma^\infty$ let $\mathrm{dom}(w)$ be $\mathbb{N}$ if $w \in \Sigma^\omega$ and $\{1, \ldots, |w|\}$ otherwise.

For two words $u = (u_i)_{i \in \mathrm{dom}(u)} \in \Sigma_1^\infty$ and $v = (v_i)_{i \in \mathrm{dom}(v)} \in \Sigma_2^\infty$ with $\mathrm{dom}(u) = \mathrm{dom}(v)$ we define the word (u, v) as $((u_i, v_i))_{i \in \mathrm{dom}(u)} \in (\Sigma_1 \times \Sigma_2)^\infty$.

Given a set X and a subset $Y \subseteq X$ let $\mathbb{1}_Y \colon X \to \{0, 1\}$ be the characteristic function of Y, i.e. $\mathbb{1}_Y(x) = 1$ if $x \in Y$ and $\mathbb{1}_Y(x) = 0$ otherwise. In the case $X = \mathbb{N}$, $\mathbb{1}_Y$ is also interpreted as ω-word over $\{0, 1\}$. Also for $f \colon X \to \mathbb{R}$ let $\mathrm{supp}(f) = \{x \in X \mid f(x) \neq 0\}$.

A *σ-field* over a set Ω is a system $\mathcal{A}$ of subsets of Ω which includes the empty set and is closed under complement and countable union. The pair $(\Omega, \mathcal{A})$ is called a *measurable space*. A *measure* on $\mathcal{A}$ is a mapping $\mu \colon \mathcal{A} \to [0, \infty]$ such that $\mu(\emptyset) = 0$ and $\mu(\bigcup_{i \geq 1} M_i) = \sum_{i \geq 1} \mu(M_i)$ for pairwise disjoint $M_i \in \mathcal{A}$. If $\mu(\Omega) = 1$, μ is called a *probability measure*.

Let $(\Omega', \mathcal{A}')$ be another measurable space. A function $f \colon \Omega \to \Omega'$ is $\mathcal{A}$-$\mathcal{A}'$-*measurable* if $f^{-1}(M') \in \mathcal{A}$ for every $M' \in \mathcal{A}'$. Now let f be $\mathcal{A}$-$\mathcal{A}'$-measurable and μ a measure on $\mathcal{A}$. The *image measure* of μ under f is the measure $\mu \circ f^{-1}$ on $\mathcal{A}'$ defined by $(\mu \circ f^{-1})(M') = \mu(f^{-1}(M'))$ for all $M' \in \mathcal{A}'$.

A measurable function $s \colon \Omega \to \mathbb{R}$ of the form $s = \sum_{i=1}^n r_i \mathbb{1}_{M_i}$ for $r_i \geq 0$ and $M_i \in \mathcal{A}$ is called simple. The integral of s is defined by $\int s \, \mathrm{d}\mu = \sum_{i=1}^n r_i \cdot \mu M_i$. For an arbitrary $\mathcal{A}$-Borel($\mathbb{R}$)-measurable function $f \colon \Omega \to [0, \infty]$ the integral is then given by

$$\int f \, \mathrm{d}\mu = \int_\Omega f(x) \, \mu(\mathrm{d}x) = \sup \left\{ \int s \, \mathrm{d}\mu \;\middle|\; 0 \leq s \leq f,\ s \text{ simple} \right\}.$$

2.1 Bernoulli Measures

Let X be a finite set. We denote the product σ-field $\bigotimes_{i=1}^\infty \mathcal{P}(X)$ on the base set X^ω by $\mathcal{A}_X$. Then $\mathcal{A}_X$ is the σ-field generated by all cones xX^ω for $x \in X^*$. As the system of all cones is closed under intersection, any two measures that agree on all cones are equal by standard measure theory.

For a probability distribution $p = (p_x)_{x \in X}$ on X let $\mathbb{P}_p^X$ be the corresponding probability measure on $\mathcal{P}(X)$, i.e. $\mathbb{P}_p^X(\{x\}) = p_x$ for all $x \in X$. If $X = \{0,1\}$ and $p \in [0,1]$, we simply write $\mathbb{P}_p$ for $\mathbb{P}_{(1-p,p)}^{\{0,1\}}$, i.e. $\mathbb{P}_p(\{1\}) = p$.

The *Bernoulli measure* $\mathrm{B}_p^{X^\omega}$ is defined as the product measure $\bigotimes_{i=1}^{\infty} \mathbb{P}_p^X$. It is explicitly given by $\mathrm{B}_p^{X^\omega}(x_1 \cdots x_k X^\omega) = \prod_{i=1}^k p_{x_i}$, for all cones $x_1 \cdots x_k X^\omega \in \mathcal{A}_X$. Hence $\mathrm{B}_p^{X^\omega}$ is a probability measure on $\mathcal{A}_X$. In the case $X = \{0,1\}$ and for $p \in [0,1]$ we write B_p^ω for $\mathrm{B}_{(1-p,p)}^{\{0,1\}^\omega}$. For more background information on Bernoulli- and product measures see for example [16, Theorem 1.64].

A binary sequence $m = (m_i)_{i \in \mathrm{dom}(m)} \in \{0,1\}^\infty$ corresponds bijectively to the set $\mathrm{supp}(m) = \{i \in \mathrm{dom}(m) \mid m_i = 1\}$. We define A_ω as the σ-field on $\mathcal{P}(\mathbb{N})$ generated by supp, i.e. $\mathcal{A}_\omega = \mathrm{supp}(\mathcal{A}_{\{0,1\}})$. Note that A_ω is actually a σ-field, as supp is bijective. The Bernoulli measure B_p^ω is also transfered to $\mathcal{A}_\omega$ by supp, i.e. $\mathrm{B}_p^\omega \circ \mathrm{supp}^{-1}$ is a measure on $\mathcal{A}_\omega$. We will denote this measure also by B_p^ω.

Let $n \in \mathbb{N}$. It is also possible to define a Bernoulli measure on the finite σ-field $\mathcal{P}(X^n)$. The measure $\mathrm{B}_p^{X^n}$ is defined as the finite product $\bigotimes_{i=1}^{n} \mathbb{P}_p^X$. The measure B_p^n on $\{0,1\}^n$ resp. $\mathcal{P}(\{1,\dots,n\})$ is defined analogously to the infinite case: $\mathrm{B}_p^n = \mathrm{B}_{(1-p,p)}^{\{0,1\}^n}$ resp. $\mathrm{B}_p^n = \mathrm{B}_{(1-p,p)}^{\{0,1\}^n} \circ \mathrm{supp}^{-1}$.

2.2 Probabilistic Automata

A *probabilistic automaton* A is given by a quadruple (Q, δ, μ, F), where

- Q is a finite set of *states*
- $\delta\colon Q \times \Sigma \times Q \to [0,1]$ is the *transition probability function* such that $\sum_{q \in Q} \delta(r,a,q) = 1$ for every $r \in Q$ and $a \in \Sigma$
- $\mu\colon Q \to [0,1]$ is the *initial distribution* such that $\sum_{q \in Q} \mu(q) = 1$
- $F \subseteq Q$ is the set of *final states.*

For a word $w = w_1 \dots w_k \in \Sigma^+$ we define the *behavior* $\|A\|\colon \Sigma^+ \to [0,1]$ *of* A by

$$\|A\|(w) := \sum_{\substack{q_0,\dots,q_{k-1} \in Q \\ q_k \in F}} \mu(q_0) \prod_{i=1}^{k} \delta(q_{i-1}, w_i, q_i),$$

for each $w \in \Sigma^+$. It follows that $\|A\|(w) \in [0,1]$ for every $w \in \Sigma^+$.

We call a function $S\colon \Sigma^+ \to [0,1]$ *probabilistically recognizable* if there is a probabilistic automaton A such that $S = \|A\|$.

2.3 Probabilistic ω-Automata

Probabilistic ω-automata are a generalization of deterministic ω-automata. A *probabilistic Muller-automaton* A over an alphabet Σ is a quadruple $(Q, \delta, \mu, \mathcal{F})$, where Q, δ, μ are defined as in the finite case and $\mathcal{F} \subseteq \mathcal{P}(Q)$ is a Muller acceptance condition (cf. [1]).

For an infinite run $\rho \in Q^\omega$ let $\inf(\rho)$ denote the set of states which occur infinitely often in ρ. We say the run ρ is successful, if $\inf(\rho) \in \mathcal{F}$. For each word $w = w_1 w_2 \ldots \in \Sigma^\omega$ we define a probability measure $\mathbb{P}_A^w$ on the σ-field $\mathcal{A}_Q$ by

$$\mathbb{P}_A^w(q_0 \ldots q_k Q^\omega) := \mu(q_0) \cdot \prod_{i=1}^{k} \delta(q_{i-1}, w_i, q_i).$$

By standard measure theory, there is exactly one such probability measure $\mathbb{P}_A^w$.

The *behavior* $\|A\|\colon \Sigma^\omega \to [0,1]$ *of* A is then given by

$$\|A\|(w) := \mathbb{P}_A^w\big(\rho \in Q^\omega \ ; \ \inf(\rho) \in \mathcal{F}\big),$$

i.e. $\|A\|(w)$ is the measure of the set of all successful runs.

We call a function $S\colon \Sigma^\omega \to [0,1]$ *probabilistically ω-recognizable* if there is a probabilistic Muller-automaton A such that $S = \|A\|$.

3 Syntax and Semantics of PMSO

This section first introduces assignments and encodings. Using these definitions we will define the syntax and semantics of probabilistic MSO (PMSO) logic. Afterwards, we will give first semantic equivalences and consider possible syntax extensions.

3.1 Assignments and Encodings

For a uniform treatment of semantics we introduce assignments and their encodings. Let $\mathcal{V}_1$ be a finite set of first order variable symbols and $\mathcal{V}_2$ a disjoint finite set of second order variable symbols. We write $\mathcal{V}$ for $\mathcal{V}_1 \dot\cup \mathcal{V}_2$. Let $w \in \Sigma^\infty$ be a word. A mapping $\alpha\colon \mathcal{V} \to \mathrm{dom}(w) \cup \mathcal{P}(\mathrm{dom}(w))$ is called a $(\mathcal{V}, w)$-assignment if $\alpha(\mathcal{V}_1) \subseteq \mathrm{dom}(w)$ and $\alpha(\mathcal{V}_2) \subseteq \mathcal{P}(\mathrm{dom}(w))$. For $i \in \mathrm{dom}(w)$ and $x \in \mathcal{V}_1$ the assignment $\alpha[x \to i]$ denotes the $(\mathcal{V} \cup \{x\}, w)$-assignment which assigns x to i and agrees with α on all other variables. Likewise for $M \subseteq \mathrm{dom}(w)$ and $X \in \mathcal{V}_2$ the $(\mathcal{V} \cup \{X\}, w)$-assignment $\alpha[X \to M]$ assigns X to M and agrees with α everywhere else. We write $\alpha[L_1 \mapsto R_1, \ldots, L_n \mapsto R_n]$ for the chained assignment $\alpha[L_1 \mapsto R_1] \cdots [L_n \mapsto R_n]$.

We encode assignments as words as usual. The extended alphabet $\Sigma_\mathcal{V}$ is defined as $\Sigma \times \{0,1\}^\mathcal{V}$. Let $\overline{w} = ((w_i, \alpha_i))_{i \in \mathrm{dom}(\overline{w})} \in \Sigma_\mathcal{V}^\omega$ and $w = (w_i)_{i \in \mathrm{dom}(\overline{w})}$. We say $\overline{w}$ encodes an $(\mathcal{V}, w)$-assignment α if for every $x \in \mathcal{V}_1$ there is exactly one position j such that $\alpha_j(x) = 1$. In this case $\alpha(x)$ is then the unique position i with $\alpha_i(x) = 1$ and, for $X \in \mathcal{V}_2$, $\alpha(X)$ is the set of all positions j' with $\alpha_{j'}(X) = 1$. We denote the set of all valid encodings by $\mathcal{N}_\mathcal{V} \subseteq \Sigma_\mathcal{V}^\infty$.

Likewise every pair of a word $w \in \Sigma^\infty$ and a $(\mathcal{V}, w)$-assignment α can be encoded as a word in $N_\mathcal{V}$ in the obvious way. We will use (w, α) to describe both the pair and its encoding as word depending on the context.

Table 1. PMSO semantics

$$[\![\mathrm{P}_a(x)]\!](w,\alpha) = \begin{cases} 1, & \text{if } w_{\alpha(x)} = a \\ 0, & \text{otherwise} \end{cases} \qquad [\![x \le y]\!](w,\alpha) = \begin{cases} 1, & \text{if } \alpha(x) \le \alpha(y) \\ 0, & \text{otherwise} \end{cases}$$

$$[\![x \in X]\!](w,\alpha) = \begin{cases} 1, & \text{if } \alpha(x) \in \alpha(X) \\ 0, & \text{otherwise} \end{cases} \qquad \begin{aligned} [\![\neg\varphi]\!](w,\alpha) &= 1 - [\![\varphi]\!](w,\alpha) \\ [\![\varphi_1 \wedge \varphi_2]\!](w,\alpha) &= [\![\varphi_1]\!](w,\alpha) \cdot [\![\varphi_2]\!](w,\alpha) \end{aligned}$$

$$[\![\forall x.\varphi]\!](w,\alpha) = \begin{cases} 1, & \text{if } [\![\varphi]\!](w,\alpha[x \to i]) = 1 \text{ for all } i \in \mathrm{dom}(w) \\ 0, & \text{otherwise} \end{cases}$$

$$[\![\forall X.\varphi]\!](w,\alpha) = \begin{cases} 1, & \text{if } [\![\varphi]\!](w,\alpha[X \to M]) = 1 \text{ for all } M \subseteq \mathrm{dom}(w) \\ 0, & \text{otherwise} \end{cases}$$

- -

$$[\![\mathbb{E}_p X.\varphi]\!](w,\alpha) = \int_{\mathcal{P}(\mathrm{dom}(w))} [\![\varphi]\!](w,\alpha[X \mapsto M])\ \mathrm{B}_p^{|w|}(\mathrm{d}M)$$

3.2 Boolean PMSO

Following an idea from [3], we define the syntax of *Boolean PMSO* (bPMSO) by

$$\psi ::= \mathrm{P}_a(x) \mid x \in X \mid x \le y \mid \psi \wedge \psi \mid \neg\psi \mid \forall x.\psi \mid \forall X.\psi,$$

for $x, y \in \mathcal{V}_1$, $X \in V_2$ and $a \in \Sigma$.

The set Free(ψ) of *free variables* in ψ is defined as usual.

The semantics $[\![\psi]\!]$ of a Boolean PMSO formula ψ maps a pair (w, α) of a word $w \in \Sigma^\infty$ and a $(\mathcal{V}, w)$-assignment α with $\mathrm{Free}(\psi) \subseteq \mathcal{V}$ to a value in $[0, 1]$. The inductive definition of the semantics is given in the upper part of Table 1. It easily follows by structural induction that $[\![\psi]\!](w, \alpha) \in [0, 1]$.

Boolean PMSO corresponds essentially to the classical MSO. Disjunction and existential quantification can be obtained from the defined operators as usual.

3.3 Full PMSO

We will now extend Boolean PMSO to full PMSO. The syntax of a PMSO formula φ is given in BNF by

$$\varphi ::= \psi \mid \varphi \wedge \varphi \mid \neg\varphi \mid \mathbb{E}_p X.\varphi,$$

where ψ is a Boolean PMSO formula, $X \in \mathcal{V}_2$, and $p \in [0, 1]$ is a real number. In other words, we have added an "expected value" operator $\mathbb{E}_p$ to Boolean PMSO and permit conjunction, negation and expected value as logical operations. The set of free variables of the expected value operator is

$$\mathrm{Free}(\mathbb{E}_p X.\varphi) := \mathrm{Free}(\varphi) \setminus \{X\}\,.$$

The semantics of a PMSO formula is given in the full Table 1.

In order for $[\![\mathbb{E}_p X.\varphi]\!]$ to be well-defined, one must and can show, that the function $M \mapsto [\![\varphi]\!](w, \alpha[X \mapsto M])$ is $\mathcal{A}_\omega$-measurable and integrable for all (w, α). This is a consequence of the measurability of all ω-recognizable sets (cf. [3,23]) and of Fubini's theorem.

In case of finite words, the semantics of $\mathbb{E}_p X.\varphi$ can be rewritten to

$$[\![\mathbb{E}_p X.\varphi]\!](w, \alpha) = \sum_{M \subseteq \mathrm{dom}(w)} [\![\varphi]\!](w, \alpha[X \mapsto M]) \cdot p^{|M|}(1-p)^{|w|-|M|}.$$

Next we give an intuitive argument for the semantics of $\mathbb{E}_p X.\varphi$. The classical existential quantifier states the existence of a set which satisfies the quantified formula. Here, for the expected value operator sets are chosen using a stochastic process: For every position k we make a probabilistic choice whether k should be included in the set or not, where the probability of inclusion is p. This choice is independent from the other positions. Such a process can be considered as tossing an unfair coin for every position k to decide whether $k \in X$ holds. The semantics of the expected value operator is then the expected value of $[\![\varphi]\!]$ under this distribution. If φ is Boolean, this value can be considered as the probability that $(w, \alpha[X \mapsto M])$ satisfies φ for an arbitrary set M.

The semantics of a PMSO formula φ transfers to the extended alphabet $\Sigma_{\mathcal{V}}$ with $\mathrm{Free}(\varphi) \subseteq \mathcal{V}$ as follows. We define $[\![\varphi]\!]_{\mathcal{V}} \colon \Sigma_{\mathcal{V}}^{\infty} \to \mathbb{R}$ by

$$[\![\varphi]\!]_{\mathcal{V}}(\overline{w}) := \begin{cases} [\![\varphi]\!](w, \alpha), & \text{if } \overline{w} \in \mathcal{N}_{\mathcal{V}} \text{ and } \overline{w} = (w, \alpha) \\ 0, & \text{otherwise.} \end{cases}$$

We will use some common abbreviations:

$$(\varphi \vee \psi) := \neg(\neg\varphi \wedge \neg\psi), \qquad (\varphi \to \psi) := \neg(\varphi \wedge \neg\psi),$$
$$(\exists x.\eta) := \neg\forall x.\neg\eta, \qquad (\exists X.\eta) := \neg\forall X.\neg\eta,$$

for formulas $\varphi, \psi \in$ PMSO and $\eta \in$ bPMSO. Note that if φ and ψ are Boolean PMSO formulas, then the abbreviated formulas are again Boolean.

From the definition the semantics of $\varphi \vee \psi$ is $[\![\varphi \vee \psi]\!] = [\![\varphi]\!] + [\![\psi]\!] - [\![\varphi]\!][\![\psi]\!]$. This is analogous to the fact that that the probability of the union of two independent events A and B is $\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B) - \mathbb{P}(A)\mathbb{P}(B)$.

The following example demonstrates the use of PMSO logic using a model of a communication device.

Example 1. We consider a communication device for sending messages. At every point of time either a new input message becomes available or the device is waiting for a new message. When a new message is available the device tries to send this message. Sending a message may fail with probability $1/3$. In this case the message is stored in an internal buffer. The next time the device is waiting for a message, sending the stored message is retried. Intuitively, as sending a buffered message has already failed once, it seems to be harder to send this message. So sending a buffered message is only successful with probability $1/2$. The buffer can hold one message.

The PMSO sentence below defines for every sequence of message input (i) and wait (w) cycles the probability that this sequence will not overflow the device's buffer. In this sentence the set variable I contains all positions (i.e. points of time) where sending an input message is successful, B all positions where sending a buffered message is successful, and F all positions where the buffer is full.

$$\mathbb{E}_{2/3} I.\mathbb{E}_{1/2} B.\exists F.1 \notin F \wedge \forall x.\forall y.y = x + 1 \rightarrow$$
$$\big((\mathrm{P}_\mathrm{w}(x) \wedge x \notin B) \rightarrow (x \in F \leftrightarrow y \in F)\big) \wedge \big((\mathrm{P}_\mathrm{w}(x) \wedge x \in B) \rightarrow y \notin F\big) \wedge$$
$$\big((P_\mathrm{i}(x) \wedge x \in I) \rightarrow (x \in F \leftrightarrow y \in F)\big) \wedge \big((\mathrm{P}_\mathrm{i}(x) \wedge x \notin I) \rightarrow (x \notin F \wedge y \in F)\big)$$

3.4 Basic Properties of PMSO Semantics

The following consistency lemma is a fundamental property.

Lemma 1. *Let φ be a PMSO formula, $w \in \Sigma^\infty$, $\mathcal{V}$ a finite set of variables such that* $\mathrm{Free}(\varphi) \subseteq \mathcal{V}$*, and α a $(\mathcal{V}, w)$-assignment. Then* $[\![\varphi]\!](w, \alpha) = [\![\varphi]\!](w, \alpha|_{\mathrm{Free}(\varphi)})$.

As usual we call PMSO formula φ a PMSO sentence if $\mathrm{Free}(\varphi) = \emptyset$. As a consequence of Lemma 1, if φ is a PMSO sentence, we define $[\![\varphi]\!](w)$ as $[\![\varphi]\!](w, \alpha)$ where α is an arbitrary $(\mathcal{V}, w)$-assignment.

For two PMSO formulas φ and ψ we write $\varphi \equiv \psi$, if $[\![\varphi]\!] = [\![\psi]\!]$ holds. It follows from the semantics definition that the usual associativity, commutativity, and distributivity laws also hold for PMSO logic. For distributivity the outer formula has to be a Boolean one, i.e. $\varphi \wedge (\psi_1 \vee \psi_2) \equiv (\varphi \wedge \psi_1) \vee (\varphi \wedge \psi_2)$ only if $\varphi \in$ bPMSO. For formulas containing the expected value operator, we obtain new equivalences:

$$\neg\mathbb{E}_p X.\varphi \equiv \mathbb{E}_p X.\neg\varphi, \qquad \mathbb{E}_p X.(\varphi \wedge \psi) \equiv \varphi \wedge \mathbb{E}_p X.\psi \text{ if } X \notin \mathrm{Free}(\varphi),$$
$$\mathbb{E}_p X.\mathbb{E}_q Y.\varphi \equiv \mathbb{E}_q Y.\mathbb{E}_p X.\varphi, \qquad \mathbb{E}_p X.\varphi \equiv \varphi \text{ if } X \notin \mathrm{Free}(\varphi).$$

Note that contrary to classical quantifiers, pulling negation out of the expected value operator does not change the operator at all.

These equivalences allow us to transform PMSO formulas to a simpler form. We say a formula φ is in *prenex normal form* if it of the form

$$\varphi = \mathbb{E}_{p_1} X_1 \ldots .\mathbb{E}_{p_k} X_k.\varphi_0$$

for a bPMSO formula φ_0, real values $p_1, \ldots, p_k \in [0, 1]$, and distinct second order variables $X_1, \ldots, X_k$.

Lemma 2. *Let φ be a PMSO formula, then there is an equivalent PMSO formula φ' in prenex normal form.*

3.5 Syntax Extensions

We discuss three possible syntax extensions in this section. extensions do not alter the expressiveness of PMSO.

Probability Constants. For a real number $p \in [0,1]$, we add the formula "p" to PMSO and define the semantics by $[\![p]\!](w,\alpha) = p$ for all $w \in \Sigma^\infty$ and all $(\mathcal{V}, w)$-assignments α. Then p can be expressed in PMSO by $\mathbb{E}_p X.1 \in X$.

An Extended First Order Universal Quantifier. As for weighted MSO (see Section 5), it is possible to extend the syntax and semantics of the universal first order quantifier in PMSO to PMSO formulas φ by

$$[\![\forall x.\varphi]\!](w,\alpha) := \prod_{i \in \mathrm{dom}(w)} [\![\varphi]\!](w, \alpha[x \mapsto i]).$$

Unfortunately it follows using a shrinkage argument that this form of the universal quantifier does not preserve recognizability. Therefore we restrict φ to particular formulas, which we define next.

A *step formula* is a PMSO formula φ such that there are bPMSO formulas $\varphi_1, \ldots, \varphi_n$ and real numbers $p_1, \ldots, p_n \in [0,1]$ such that φ is equivalent to $\bigwedge_{i=1}^n (\varphi_i \to p_i)$. When this condition is satisfied $\forall x.\varphi$ can be rewritten as

$$\mathbb{E}_{p_1} X_1 \ldots . \mathbb{E}_{p_n} X_n . \forall x. \bigwedge_{i=1}^{n} (\varphi_i \to x \in X_i),$$

for new second order variables $X_1, \ldots, X_n$.

A First Order Expected Value Operator. In the case of infinite words, it is possible to define a first order expected value operator with reasonable semantics.

Let φ be a PMSO formula, $p \in (0,1)$, and x a first order variable. We define

$$\mathbb{E}_p x.\varphi := \mathbb{E}_p X.\tilde{\varphi},$$

where $\tilde{\varphi}$ is obtained from φ by replacing every occurrence of x with $\min X$. Note that, though "$\min X$" is not valid PMSO syntax, it is a well-known MSO property and thus expressible in PMSO. Also $\{\emptyset\}$ is a B_p^ω-null set.

To express the semantics of the just defined operator in a natural way, we introduce the geometric distribution on $\mathbb{N}$. For $p \in (0,1)$ let $\mathrm{G}_p\big(\{n\}\big) = (1-p)^{n-1}p$. Intuitively, $\mathrm{G}_p(\{n\})$ is the probability to get one success after n experiments in an infinitely running Bernoulli experiment. It follows that $\mathrm{G}_p = \mathrm{B}_p^\omega \circ \min^{-1}$. We apply this equality to $\mathbb{E}_p x.\varphi$ and obtain

$$[\![\mathbb{E}_p x.\varphi]\!](w,\alpha) = \int_{\mathbb{N}} [\![\varphi]\!](w, \alpha[x \mapsto i]) \; \mathrm{G}_p(\mathrm{d}i).$$

4 Equivalence of PMSO and Probabilistic Automata

Our main theorem establishes the desired expressive equivalence of PMSO sentences and probabilistic automata for both cases of finite and infinite words.

Theorem 1. *Let Σ be an alphabet.*

1. *A function $S\colon \Sigma^+ \to [0,1]$ is probabilistically recognizable iff there is a PMSO sentence φ such that $[\![\varphi]\!] = S$.*
2. *A function $S\colon \Sigma^\omega \to [0,1]$ is probabilistically ω-recognizable iff there is a PMSO sentence φ such that $[\![\varphi]\!] = S$.*

We will sketch the proof of Theorem 1 for infinite words in the rest of this section. The finite case is analogous.

For the rest of the section we write simply B_p for the Bernoulli measure if the base set is understood.

4.1 Characterization by Bernoulli Measures

We give a characterization of probabilistically recognizable functions using Bernoulli measures. This characterization could be of independent interest.

Theorem 2. *A function $S\colon \Sigma^\omega \to [0,1]$ is probabilistically ω-recognizable iff there is an alphabet Γ, a distribution p on Γ and an ω-recognizable language $L \subseteq (\Sigma \times \Gamma)^\omega$ such that*

$$S(w) = \mathrm{B}_p\big(\{u \in \Gamma^\omega \mid (w,u) \in L\}\big), \tag{1}$$

for all $w \in \Sigma^\omega$.

Equation 1 means that S is an image measure. Indeed if $\theta_w(u) = (w,u)$, then (1) can be written as $S(w) = \mathrm{B}_p \circ \theta_w^{-1}(L)$.

Proof. Given a probabilistic ω-automaton $A = (Q, \delta, \mathbb{1}_{\{\iota\}}, \mathcal{F})$ we use an enumeration $0 = d_0 < \ldots < d_n = 1$ of the set $\{\sum_{i=1}^q \delta(p,a,q) \mid p,q \in Q,\ a \in \Sigma\} \cup \{0\}$ to define $\Gamma := \{1,\ldots,n\}$ and p by $p_k := d_k - d_{k-1}$ for all $k \in \Gamma$. We construct a Muller-automaton $B = (Q, T, \iota, \mathcal{F})$ from A such that $\sum_{u \in \Gamma,\,(p,(a,u),q) \in T} = \delta(p,a,q)$ and define L as the language accepted by B.

Conversely, given a Muller-automaton $B = (Q, T, \iota, \mathcal{F})$, we construct a probabilistic Muller-automaton A which recognizes S. We define $A := (Q, \delta, \mathbb{1}_{\{\iota\}}, \mathcal{F})$ where $\delta(p,a,q) := \sum_{u \in \Gamma,\,(p,(a,u),q) \in T} p_u$. □

The last theorem used a Bernoulli measure on a finite, but arbitrary large, set Γ. In PMSO only Bernoulli measures on the two element set $\{0,1\}$ are available.

Corollary 1. *A function $S\colon \Sigma^\omega \to [0,1]$ is probabilistically ω-recognizable iff there are a natural number $n \in \mathbb{N}$, real numbers $r_1,\ldots,r_n \in [0,1]$ and an ω-recognizable language $L \subseteq (\Sigma \times \{0,1\}^n)^\omega$ such that*

$$S(w) = \left(\bigotimes_{i=1}^{n} \mathrm{B}_{r_i}\right)\Big(\big\{(M_1,\ldots,M_n) \in \mathcal{P}(\mathbb{N})^n \;\big|\; (w, \mathbb{1}_{M_1},\ldots,\mathbb{1}_{M_n}) \in L\big\}\Big),$$

for all $w \in \Sigma^\omega$.

Proof. We show that a Bernoulli measure on an arbitrary finite set can be written as an image measure under a suitable mapping h of a finite product of binary Bernoulli measures. Next, we apply Theorem 2 and show that h^{-1} retains the recognizability of L in Theorem 2. □

4.2 Proof of Theorem 1

Let $S\colon \Sigma^\omega \to [0,1]$ be probabilistically ω-recognizable. By Corollary 1 there are $n \in \mathbb{N}$, real numbers $r_1, \ldots, r_n \in [0,1]$ and an ω-recognizable language $L \subseteq (\Sigma \times \{0,1\}^n)^\omega$ such that

$$S(w) = \left(\bigotimes_{i=1}^{n} \mathrm{B}_{r_i}\right)\left(\left\{(M_1,\ldots,M_n) \in \mathcal{P}(\mathbb{N})^n \;\middle|\; (w, \mathbb{1}_{M_1}, \ldots, \mathbb{1}_{M_n}) \in L\right\}\right).$$

Let $\mathcal{V} = \{X_1, \ldots, X_n\}$. By Büchi's theorem there is a bPMSO formula φ_0 over Σ with $\mathrm{Free}(\varphi_0) = \mathcal{V}$ which defines L, i.e. $L = \mathrm{supp}(\llbracket\varphi_0\rrbracket_{\mathcal{V}})$. Let φ be the PMSO sentence given by

$$\varphi = \mathbb{E}_{r_1} X_1 \ldots \mathbb{E}_{r_n} X_n . \varphi_0 .$$

It follows by Fubini's theorem that $S = \llbracket\varphi\rrbracket$.

Conversely, let φ be a PMSO sentence with $S = \llbracket\varphi\rrbracket$. By Lemma 2 we may assume that φ is in prenex normal form, i.e. $\varphi = \mathbb{E}_{r_1} X_1 \ldots \mathbb{E}_{r_n} X_n . \varphi_0$ for a Boolean PMSO formula φ_0, real numbers $r_1, \ldots, r_n \in [0,1]$, and distinct set variables $X_1, \ldots, X_n$. Let $\mathcal{V} = \{X_1, \ldots, X_n\}$ and $L = \mathrm{supp}(\llbracket\varphi_0\rrbracket_{\mathcal{V}})$. Then $\llbracket\varphi_0\rrbracket_{\mathcal{V}} = \mathbb{1}_L$ and L is ω-recognizable by Büchi's theorem as φ_0 is Boolean. By Fubini's theorem we obtain

$$S(w) = \left(\bigotimes_{i=1}^{n} \mathrm{B}_{r_i}\right)\left(\left\{(M_1,\ldots,M_n) \in \mathcal{P}(\mathbb{N})^n \;\middle|\; (w, \mathbb{1}_{M_1}, \ldots, \mathbb{1}_{M_n}) \in L\right\}\right).$$

Therefore S is probabilistically ω-recognizable by Corollary 1. □

Remark 1. When translating a PMSO formula to a probabilistic Muller-automaton the acceptance condition of the automaton can be chosen to be a Rabin, Streett, or parity condition. This is because for all of these acceptance conditions classical ω-automata can be determinized. The latter is not true for the Büchi, reachability or safety acceptance conditions.

5 Relation to Weighted MSO

In [12] a weighted MSO (wMSO) logic was introduced. It was shown that a certain fragment of weighted MSO logic is expressively equivalent to weighted automata. This expressive equivalence holds for finite and infinite words and also for arbitrary semirings. Whereas probabilistic automata on infinite words represent a different model than weighted automata on infinite words, probabilistic automata on finite words are a special case of weighted automata over the semiring of the non-negative real numbers $\mathbb{R}^+$.

For the exact definitions of weighted automata, weighted MSO, and syntactically restricted weighted MSO (srMSO) see [12,13].

As shown in [12], a function $S\colon \Sigma^+ \to \mathbb{R}^+$ is recognizable by a weighted automaton iff S is definable in srMSO. Hence every PMSO formula can be translated to a probabilistic automaton, which then can be translated to a srMSO formula. We give a direct mapping to embed PMSO into srMSO using a syntactic transformation.

Theorem 3. *Let φ be a PMSO formula. Then there is a srMSO formula φ' over the semiring of the non-negative real numbers such that $[\![\varphi]\!] = [\![\varphi']\!]$. Moreover φ' can be obtained from φ by a effective syntactic transformation.*

Intuitively, φ' is obtained from φ by replacing every occurrence of $\mathbb{E}_p X.\varphi_0$ with $\exists X.\varphi_0 \wedge \forall x.\big((r \wedge x \in X) \vee ((1-r) \wedge x \notin X)\big)$ where x is a new variable.

6 Conclusion and Future Work

We introduced a probabilistic extension of classical MSO logic by the addition of an expected value operator. We could show that, similarly to the fundamental Büchi-Elgot-Theorem, this probabilistic MSO logic is expressively equivalent to probabilistic automata on finite and infinite words. We also gave several syntax extensions and an effective embedding into weighted MSO logic.

As our transformations between PMSO sentences and probabilistic automata are effective, all decidability results for probabilistic automata also apply to PMSO sentences. For example, it is decidable for a PMSO sentence φ if there is a finite word $w \in \Sigma^+$ such that $[\![\varphi]\!](w) > 0$ $(= 1)$ [18], or if two given PMSO formulas are equivalent on finite words [21]. On the other hand, interesting problems are undecidable. For instance, for a given formula φ it is undecidable if there is an infinite word w such that $[\![\varphi]\!](w) > 0$ $(= 1)$ [2]. Another undecidable problem is to decide for a formula φ and some $\lambda \in (0,1)$ if there is a finite or infinite word w such that $[\![\varphi]\!](w) > \lambda$ [18]. This problem remains undecidable even for $\lambda = 1/2$ and $\varphi = \mathbb{E}_{1/2} X.\varphi_0$ where φ_0 is Boolean [15].

Many concepts of probabilistic ω-automata like safety, reachability or Büchi acceptance conditions, hierarchical probabilistic automata [6], #-acyclic automata [15], or probabilistic automata which induce a simple process [10] have better decidability properties. It is an open problem to derive any of these concepts for PMSO logic.

In current work, we wish to find similar probabilistic extensions for temporal logics. For example a suitable probabilistic LTL should be expressively equivalent to the first order fragment of probabilistic MSO logic. We also hope to obtain better decidability properties using this approach.

Acknowledgements. This work is part of ongoing PhD work supervised by Prof. Manfred Droste at Leipzig University.

References

1. Baier, C., Grösser, M.: Recognizing ω-regular languages with probabilistic automata. In: Proc. LICS, pp. 137–146. IEEE (2005)
2. Baier, C., Bertrand, N., Größer, M.: On Decision Problems for Probabilistic Büchi Automata. In: Amadio, R.M. (ed.) FOSSACS 2008. LNCS, vol. 4962, pp. 287–301. Springer, Heidelberg (2008)
3. Bollig, B., Gastin, P.: Weighted versus Probabilistic Logics. In: Diekert, V., Nowotka, D. (eds.) DLT 2009. LNCS, vol. 5583, pp. 18–38. Springer, Heidelberg (2009)

4. Büchi, J.R.: Weak second-order arithmetic and finite automata. Zeitschrift für Mathematische Logik und Grundlagen der Mathematik 6, 66–92 (1960)
5. Bukharaev, R.G.: Theorie der stochastischen Automaten. Teubner (1995)
6. Chadha, R., Sistla, A.P., Viswanathan, M.: Power of Randomization in Automata on Infinite Strings. In: Bravetti, M., Zavattaro, G. (eds.) CONCUR 2009. LNCS, vol. 5710, pp. 229–243. Springer, Heidelberg (2009)
7. Chadha, R., Sistla, A.P., Viswanathan, M.: Probabilistic Büchi Automata with Non-extremal Acceptance Thresholds. In: Jhala, R., Schmidt, D. (eds.) VMCAI 2011. LNCS, vol. 6538, pp. 103–117. Springer, Heidelberg (2011)
8. Chatterjee, K., Doyen, L., Henzinger, T.A.: Probabilistic Weighted Automata. In: Bravetti, M., Zavattaro, G. (eds.) CONCUR 2009. LNCS, vol. 5710, pp. 244–258. Springer, Heidelberg (2009)
9. Chatterjee, K., Henzinger, T.: Probabilistic Automata on Infinite Words: Decidability and Undecidability Results. In: Bouajjani, A., Chin, W.-N. (eds.) ATVA 2010. LNCS, vol. 6252, pp. 1–16. Springer, Heidelberg (2010)
10. Chatterjee, K., Tracol, M.: Decidable problems for probabilistic automata on infinite words. CoRR abs/1107.2091 (2011)
11. Cheung, L., Lynch, N., Segala, R., Vaandrager, F.: Switched PIOA: Parallel composition via distributed scheduling. TCS 365(1-2), 83–108 (2006)
12. Droste, M., Gastin, P.: Weighted automata and weighted logics. Theoretical Computer Science 380(1-2), 69–86 (2007)
13. Droste, M., Kuich, W., Vogler, H., (eds.): Handbook of Weighted Automata. EATCS Monographs in Theoretical Computer Science. Springer (2009)
14. Elgot, C.C.: Decision problems of finite automata design and related arithmetics. Trans. Amer. Math. Soc. 98, 21–51 (1961)
15. Gimbert, H., Oualhadj, Y.: Probabilistic Automata on Finite Words: Decidable and Undecidable Problems. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010, Part II. LNCS, vol. 6199, pp. 527–538. Springer, Heidelberg (2010)
16. Klenke, A.: Probability Theory: A Comprehensive Course, 1st edn. Universitext. Springer (December 2007)
17. Mora-López, L., Morales, R., Sidrach de Cardona, M., Triguero, F.: Probabilistic finite automata and randomness in nature: a new approach in the modelling and prediction of climatic parameters. In: Proc. International Environmental Modelling and Software Congress, pp. 78–83 (2002)
18. Paz, A.: Introduction to Probabilistic Automata. Computer Science and Applied Mathematics. Academic Press, Inc. (1971)
19. Rabin, M.O.: Probabilistic automata. Information and Control 6(3), 230–245 (1963)
20. Ron, D., Singer, Y., Tishby, N.: The power of amnesia: Learning probabilistic automata with variable memory length. Machine Learning 25, 117–149 (1996)
21. Sakarovitch, J.: Rational and recognisable power series. In: Droste, et al. (eds.) [13] Handbook, pp. 105–174
22. Tracol, M., Baier, C., Größer, M.: Recurrence and transience for probabilistic automata. In: FSTTCS, vol. 4, pp. 395–406. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2009)
23. Vardi, M.Y.: Automatic verification of probabilistic concurrent finite state programs. In: Foundations of Computer Science, pp. 327–338 (1985)