Runtime Analysis of Probabilistic Programs with Unbounded Recursion

Tomáš Brázdil[1]*, Stefan Kiefer[2]†, Antonín Kučera[1]*, and Ivana Hutařová Vařeková[1]‡

[1] Faculty of Informatics, Masaryk University, Czech Republic.
{brazdil,kucera}@fi.muni.cz,ivarekova@centrum.cz
[2] Department of Computer Science, University of Oxford, United Kingdom.
stefan.kiefer@cs.ox.ac.uk

Abstract. We study the runtime in probabilistic programs with unbounded recursion. As underlying formal model for such programs we use *probabilistic pushdown automata (pPDA)* which exactly correspond to recursive Markov chains. We show that every pPDA can be transformed into a stateless pPDA (called "pBPA") whose runtime and further properties are closely related to those of the original pPDA. This result substantially simplifies the analysis of runtime and other pPDA properties. We prove that for every pPDA the probability of performing a long run decreases *exponentially* in the length of the run, if and only if the expected runtime in the pPDA is *finite*. If the expectation is infinite, then the probability decreases "polynomially". We show that these bounds are asymptotically tight. Our tail bounds on the runtime are *generic*, i.e., applicable to any probabilistic program with unbounded recursion. An intuitive interpretation is that in pPDA the runtime is exponentially unlikely to deviate from its expected value.

1 Introduction

We study the termination time in programs with unbounded recursion, which are either randomized or operate on statistically quantified inputs. As underlying formal model for such programs we use *probabilistic pushdown automata (pPDA)* [13, 14, 6, 3] which are equivalent to recursive Markov chains [18, 16, 17]. Since pushdown automata are a standard and well-established model for programs with recursive procedure calls, our abstract results imply *generic* and *tight* tail bounds for termination time, the main performance characteristic of probabilistic recursive programs.

A pPDA consists of a finite set of *control states*, a finite *stack alphabet*, and a finite set of *rules* of the form $pX \xhookrightarrow{x} q\alpha$, where p, q are control states, X is a stack symbol, α is a finite sequence of stack symbols (possibly empty), and $x \in (0, 1]$ is the (rational) probability of the rule. We require that for each pX, the sum of the probabilities of all rules of the form $pX \xhookrightarrow{x} q\alpha$ is equal to 1. Each pPDA Δ induces an infinite-state Markov chain M_Δ, where the states are configurations of the form $p\alpha$ (p is the current control state and α is the current stack content), and $pX\beta \xrightarrow{x} q\alpha\beta$ is a transition of M_Δ iff

* Tomáš Brázdil and Antonín Kučera are supported by the Institute for Theoretical Computer Science (ITI), project No. 1M0545, and by the Czech Science Foundation, grant No. P202/10/1469.

† Stefan Kiefer is supported by a postdoctoral fellowship of the German Academic Exchange Service (DAAD)

‡ Ivana Hutařová Vařeková is supported by by the Czech Science Foundation, grant No. 102/09/H042.

```
function And(node)
  if node.leaf then
    return node.value
  else
    v := Or(node.left)
    if v = 0 then
      return 0
    else
      return Or(node.right)
```

```
function Or(node)
  if node.leaf then
    return node.value
  else
    v := And(node.left)
    if v = 1 then
      return 1
    else
      return And(node.right)
```

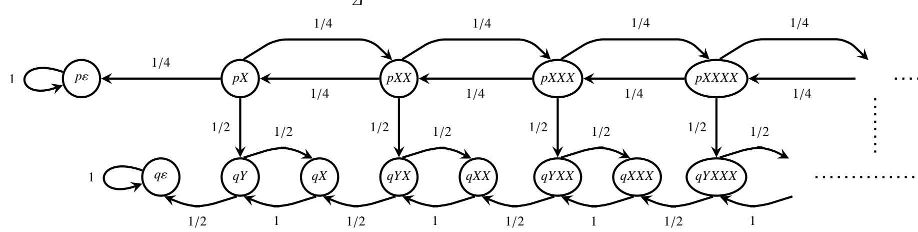
$$\begin{array}{ll} qA \xhookrightarrow{1/4} r_1\varepsilon & qO \xhookrightarrow{1/4} r_1\varepsilon \\ qA \xhookrightarrow{1/4} r_0\varepsilon & qO \xhookrightarrow{1/4} r_0\varepsilon \\ qA \xhookrightarrow{1/2} qOA & qO \xhookrightarrow{1/2} qAO \\ r_0A \xhookrightarrow{1} r_0\varepsilon & r_1O \xhookrightarrow{1} r_1\varepsilon \\ r_1A \xhookrightarrow{1} qO & r_0O \xhookrightarrow{1} qA \end{array}$$

Fig. 1. The program *Tree* and its pPDA model Δ_{Tree}.

$pX \xhookrightarrow{x} q\alpha$ is a rule of Δ. We also stipulate that $p\varepsilon \xrightarrow{1} p\varepsilon$ for every control state p, where ε denotes the empty stack. For example, consider the pPDA $\hat{\Delta}$ with two control states p, q, two stack symbols X, Y, and the rules

$$pX \xhookrightarrow{1/4} p\varepsilon,\ pX \xhookrightarrow{1/4} pXX,\ pX \xhookrightarrow{1/2} qY,\ pY \xhookrightarrow{1} pY,\ qY \xhookrightarrow{1/2} qX,\ qY \xhookrightarrow{1/2} q\varepsilon,\ qX \xhookrightarrow{1} qY\,.$$

The structure of Markov chain $M_{\hat{\Delta}}$ is indicated below.



pPDA can model programs that use unbounded "stack-like" data structures such as stacks, counters, or even queues (in some cases, the exact ordering of items stored in a queue is irrelevant and the queue can be safely replaced with a stack). Transition probabilities may reflect the random choices of the program (such as "coin flips" in randomized algorithms) or some statistical assumptions about the input data. In particular, pPDA model *recursive* programs. The global data of such a program are stored in the finite control, and the individual procedures and functions together with their local data correspond to the stack symbols (a function call/return is modeled by pushing/popping the associated stack symbol onto/from the stack). As a simple example, consider the recursive program *Tree* of Figure 1, which computes the value of an *And/Or-tree*, i.e., a tree such that (i) every node has either zero or two children, (ii) every inner node is either an And-node or an Or-node, and (iii) on any path from the root to a leaf And- and Or-nodes alternate. We further assume that the root is either a leaf or an And-node. *Tree* starts by invoking the function `And` on the root of a given And/Or-tree. Observe that the program evaluates subtrees only if necessary. Now assume that the input are random And/Or trees following the Galton-Watson distribution: a node of the tree has two children with probability $1/2$, and no children with probability $1/2$. Furthermore, the conditional probabilities that a childless node evaluates to 0 and 1 are also both equal to $1/2$. On inputs with this distribution, the algorithm corresponds to a pPDA Δ_{Tree} of Figure 1 (the control states r_0 and r_1 model the return values 0 and 1).

We study the *termination time* of runs in a given pPDA Δ. For every pair of control states p, q and every stack symbol X of Δ, let $Run(pXq)$ be the set of all runs (infinite paths) in M_Δ initiated in pX which visit $q\varepsilon$. The termination time is modeled by the random variable $\mathbf{T}_{pX}$, which to every run w assigns either the number of steps needed to reach a configuration with empty stack, or ∞ if there is no such configuration. The conditional expected value $\mathbb{E}[\mathbf{T}_{pX} \mid Run(pXq)]$, denoted just by $E[pXq]$ for short, then corresponds to the average number of steps needed to reach $q\varepsilon$ from pX, computed only for those runs initiated in pX which terminate in $q\varepsilon$. For example, using the results of [13, 14, 18], one can show that the functions `And` and `Or` of the program *Tree* terminate with probability one, and the expected termination times can be computed by solving a system of linear equations. Thus, we obtain the following:

$$\begin{array}{llll} E[qAr_0] = 7.155113 & E[qOr_0] = 7.172218 & E[r_0Ar_0] = 1.000000 & E[r_1Or_1] = 1.000000 \\ E[qAr_1] = 7.172218 & E[qOr_1] = 7.155113 & E[r_1Ar_0] = 8.172218 & E[r_0Or_1] = 8.172218 \\ & & E[r_1Ar_1] = 8.155113 & E[r_0Or_0] = 8.155113 \end{array}$$

However, the mere expectation of the termination time does not provide much information about its distribution until we analyze the associated *tail bound*, i.e., the probability that the termination time deviates from its expected value by a given amount. That is, we are interested in bounds for the conditional probability $\mathcal{P}(\mathbf{T}_{pX} \geq n \mid Run(pXq))$. (Note this probability makes sense regardless of whether $E[pXq]$ is finite or infinite.) Assuming that the (conditional) expectation and variance of $\mathbf{T}_{pX}$ are finite, one can apply Markov's and Chebyshev's inequalities and thus yield bounds of the form $\mathcal{P}(\mathbf{T}_{pX} \geq n \mid Run(pXq)) \leq c/n$ and $\mathcal{P}(\mathbf{T}_{pX} \geq n \mid Run(pXq)) \leq c/n^2$, respectively, where c is a constant depending only on the underlying pPDA. However, these bounds are asymptotically always worse than our exponential bound (see below). If $E[pXq]$ is infinite, these inequalities cannot be used at all.

Our contribution. The main contributions of this paper are the following:

- We show that every pPDA can be effectively transformed into a *stateless* pPDA (called "pBPA") so that all important quantitative characteristics of runs are preserved. This simple (but fundamental) observation was overlooked in previous works on pPDA and related models [13, 14, 6, 3, 18, 16, 17], although it simplifies virtually all of these results. Hence, we can w.l.o.g. concentrate just on the study of pBPA. Moreover, for the runtime analysis, the transformation yields a pBPA all of whose symbols terminate with probability one, which further simplifies the analysis.
- We provide tail bounds for $\mathbf{T}_{pX}$ which are *asymptotically optimal for every pPDA* and are applicable also in the case when $E[pXq]$ is infinite. More precisely, we show that for every pair of control states p, q and every stack symbol X, there are essentially three possibilities:
  - There is a "small" k such that $\mathcal{P}(\mathbf{T}_{pX} \geq n \mid Run(pXq)) = 0$ for all $n \geq k$.
  - $E[pXq]$ is finite and $\mathcal{P}(\mathbf{T}_{pX} \geq n \mid Run(pXq))$ decreases exponentially in n.
  - $E[pXq]$ is infinite and $\mathcal{P}(\mathbf{T}_{pX} \geq n \mid Run(pXq))$ decreases "polynomially" in n.

  The exact formulation of this result, including the explanation of what is meant by a "polynomial" decrease, is given in Theorem 7 (technically, Theorem 7 is formulated for pBPA which terminate with probability one, which is no restriction as explained above). Observe that a direct consequence of the above theorem is that *all* conditional moments $\mathbb{E}[\mathbf{T}_{pX}^k \mid Run(pXq)]$ are simultaneously either finite or infinite (in particular, if $E[pXq]$ is finite, then so is the conditional variance of $\mathbf{T}_{pX}$).

The characterization given in Theorem 7 is effective. In particular, it is decidable in polynomial space whether $E[pXq]$ is finite or infinite by using the results of [13, 14, 18], and if $E[pXq]$ is finite, we can compute concrete bounds on the probabilities. Our results vastly improve on what was previously known on the termination time $\mathbf{T}_{pX}$. Previous work, in particular [14, 2], has focused on computing expectations and variances for a class of random variables on pPDA runs, a class that includes $\mathbf{T}_{pX}$ as prime example. Note that our exponential bound given in Theorem 7 depends, like Markov's inequality, only on expectations, which can be efficiently approximated by the methods of [14, 12].

An intuitive interpretation of our results is that pPDA with finite (conditional) expected termination time are well-behaved in the sense that the termination time is exponentially unlikely to deviate from its expectation. Of course, a detailed analysis of a concrete pPDA may lead to better bounds, but these bounds will be *asymptotically equivalent* to our generic bounds. Also note that the conditional expected termination time can be finite even for pPDA that do not terminate with probability one. Hence, for every $\varepsilon > 0$ we can compute a tight threshold k such that if a given pPDA terminates at all, it terminates after at most k steps with probability $1 - \varepsilon$ (this is useful for interrupting programs that are supposed but not guaranteed to terminate).

Proof techniques. The main mathematical tool for establishing our results on runtime is (basic) martingale theory and its tools such as the optional stopping theorem and Azuma's inequality (see Section 3.2). More precisely, we construct two different martingales corresponding to the cases when the expected termination time is finite resp. infinite. In combination with our reduction to pBPA this establishes a powerful link between pBPA, pPDA, and martingale theory.

Our analysis of termination time in the case when the expected termination time is infinite builds on Perron-Frobenius theory for nonnegative matrices as well as on recent results from [18, 12]. We also use some of the observations presented in [13, 14, 6].

Related work. The application of Azuma's inequality in the analysis of particular randomized algorithms is also known as the *method of bounded differences*; see, e.g., [24, 10] and the references therein. In contrast, we apply martingale methods not to particular algorithms, but to the pPDA model as a whole.

Analyzing the distribution of termination time is closely related to the analysis of multitype branching processes (MT-BPs) [19]. A MT-BP is very much like a pBPA (see above). The stack symbols in pBPA correspond to species in MT-BPs. An ε-rule corresponds to the death of an individual, whereas a rule with two or more symbols on the right hand side corresponds to reproduction. Since in MT-BPs the symbols on the right hand side of rules evolve concurrently, termination time in pBPA does *not* correspond to extinction time in MT-BPs, but to the size of the *total progeny* of an individual, i.e., the number of direct or indirect descendants of an individual. The distribution of the total progeny of a MT-BP has been studied mainly for the case of a single species, see, e.g., [19, 25, 26] and the references therein, but to the best of our knowledge, no tail bounds for MT-BPs have been given. Hence, Theorem 7 can also be seen as a contribution to MT-BP theory.

Stochastic context-free grammars (SCFGs) [23] are also closely related to pBPA. The termination time in pBPA corresponds to the number of nodes in a derivation tree of a SCFG, so our analysis of pBPA immediately applies to SCFGs. Quasi-Birth-Death processes (QBDs) can also be seen as a special case of pPDA. A QBD is a generalization of a birth-death process studied in queueing theory and applied probability (see, e.g., [22, 1, 15]). Intuitively, a QBD describes an unbounded queue, using a counter to count the number of jobs in the queue, where the queue can be in one of finitely many distinct "modes". Hence, a (discrete-time) QBD can be equivalently defined by a pPDA with one stack symbol used to emulate the counter. These special pPDA are also known as *probabilistic one-counter automata (pOC)* [15, 5, 4]. Recently, it has been shown in [7] that every pOC induces a martingale apt for studying the properties of both terminating and nonterminating runs in pOC. The construction is based on ideas specific to pOC that are completely unrelated to the ones presented in this paper.

Previous work on pPDA and the equivalent model of recursive Markov chains includes [13, 14, 6, 3, 18, 16, 17]. In this paper we use many of the results presented in these papers, which is explicitly acknowledged at appropriate places. Missing proofs can be found in [8].

2 Preliminaries

In the rest of this paper, $\mathbb{N}$, $\mathbb{N}_0$, and $\mathbb{R}$ denote the set of positive integers, non-negative integers, and real numbers, respectively. The tuples of $A_1 \times A_2 \cdots \times A_n$ are often written simply as $a_1 a_2 \ldots a_n$. The set of all finite words over a given alphabet Σ is denoted by Σ^*, and the set of all infinite words over Σ is denoted by Σ^ω. We write ε for the empty word. The length of a given $w \in \Sigma^* \cup \Sigma^\omega$ is denoted by $|w|$, where the length of an infinite word is ∞. Given a word (finite or infinite) over Σ, the individual letters of w are denoted by $w(0), w(1), \ldots$

Definition 1 (Markov Chains). *A* Markov chain *is a triple $M = (S, \rightarrow, Prob)$ where S is a finite or countably infinite set of* states*, $\rightarrow \subseteq S \times S$ is a* transition relation*, and $Prob$ is a function which to each transition $s \rightarrow t$ of M assigns its probability $Prob(s \rightarrow t) > 0$ so that for every $s \in S$ we have $\sum_{s \rightarrow t} Prob(s \rightarrow t) = 1$ (as usual, we write $s \xrightarrow{x} t$ instead of $Prob(s \rightarrow t) = x$).*

A *path* in M is a finite or infinite word $w \in S^+ \cup S^\omega$ such that $w(i-1) \rightarrow w(i)$ for every $1 \leq i < |w|$. A *run* in M is an infinite path in M. We denote by $Run[M]$ the set of all runs in M. The set of all runs that start with a given finite path w is denoted by $Run[M](w)$. When M is understood, we write just Run and $Run(w)$ instead of $Run[M]$ and $Run[M](w)$, respectively. Given $s \in S$ and $A \subseteq S$, we say *A is reachable from s* if there is a run w such that $w(0) = s$ and $w(i) \in A$ for some $i \geq 0$.

To every $s \in S$ we associate the probability space $(Run(s), \mathcal{F}, \mathcal{P})$ where $\mathcal{F}$ is the σ-field generated by all *basic cylinders* $Run(w)$ where w is a finite path starting with s, and $\mathcal{P} : \mathcal{F} \rightarrow [0, 1]$ is the unique probability measure such that $\mathcal{P}(Run(w)) = \Pi_{i=1}^{|w|-1} x_i$ where $w(i-1) \xrightarrow{x_i} w(i)$ for every $1 \leq i < |w|$. If $|w| = 1$, we put $\mathcal{P}(Run(w)) = 1$. Note that only certain subsets of $Run(s)$ are $\mathcal{P}$-measurable, but in this paper we only deal with "safe" subsets that are guaranteed to be in $\mathcal{F}$.

Definition 2 (probabilistic PDA). *A* probabilistic pushdown automaton (pPDA) *is a tuple $\Delta = (Q, \Gamma, \hookrightarrow, Prob)$ where Q is a finite set of* control states*, Γ is a finite* stack alphabet*, $\hookrightarrow \subseteq (Q \times \Gamma) \times (Q \times \Gamma^{\leq 2})$ is a* transition relation *(where $\Gamma^{\leq 2} = \{\alpha \in \Gamma^*, |\alpha| \leq 2\}$), and $Prob$ is a function which to each transition $pX \hookrightarrow q\alpha$ assigns its probability $Prob(pX \hookrightarrow q\alpha) > 0$ so that for all $p \in Q$ and $X \in \Gamma$ we have that $\sum_{pX \hookrightarrow q\alpha} Prob(pX \hookrightarrow q\alpha) = 1$. As usual, we write $pX \xhookrightarrow{x} q\alpha$ instead of $Prob(pX \hookrightarrow q\alpha) = x$.*

Elements of $Q \times \Gamma^*$ are called *configurations* of Δ. A pPDA with just one control state is called pBPA.[6] In what follows, configurations of pBPA are usually written without the (only) control state p (i.e., we write just α instead of $p\alpha$). We define the *size* of a pPDA Δ as $|\Delta| = |Q| + |\Gamma| + |\hookrightarrow| + |Prob|$, where $|Prob|$ is the sum of sizes of binary representations of values taken by $Prob$. To Δ we associate the Markov chain M_Δ with $Q \times \Gamma^*$ as the set of states and transitions defined as follows:

- $p\varepsilon \xrightarrow{1} p\varepsilon$ for each $p \in Q$;
- $pX\beta \xrightarrow{x} q\alpha\beta$ is a transition of M_Δ iff $pX \overset{x}{\hookrightarrow} q\alpha$ is a transition of Δ.

For all $pXq \in Q \times \Gamma \times Q$ and $rY \in Q \times \Gamma$, we define

- $Run(pXq) = \{w \in Run(pX) \mid w(i) = q\varepsilon \text{ for some } i \in \mathbb{N}\}$
- $Run(rY\uparrow) = Run(rY) \setminus \bigcup_{s \in Q} Run(rYs)$.

Further, we put $[pXq] = \mathcal{P}(Run(pXq))$ and $[pX\uparrow] = \mathcal{P}(Run(pX\uparrow))$. If Δ is a pBPA, we write $[X]$ and $[X\uparrow]$ instead of $[pXp]$ and $[pX\uparrow]$, where p is the only control state of Δ.

Let $p\alpha \in Q \times \Gamma^*$. We denote by $\mathbf{T}_{p\alpha}$ a random variable over $Run(p\alpha)$ where $\mathbf{T}_{p\alpha}(w)$ is either the least $n \in \mathbb{N}_0$ such that $w(n) = q\varepsilon$ for some $q \in Q$, or ∞ if there is no such n. Intuitively, $\mathbf{T}_{p\alpha}(w)$ is the number of steps ("the time") in which the run w initiated in $p\alpha$ terminates.

3 The Results

In this section we present the results outlined in Section 1. More precisely, in Section 3.1 we show how to transform a given pPDA into an equivalent pBPA, and in Section 3.2 we design the promised martingales and derive our tight tail bounds for the termination probability.

3.1 Transforming pPDA into pBPA

Let $\Delta = (Q, \Gamma, \hookrightarrow, Prob)$ be a pPDA. We show how to construct a pBPA $\Delta_\bullet$ which is "equivalent" to Δ in a well-defined sense. This construction is a relatively straightforward modification of the standard method for transforming a PDA into an equivalent context-free grammar (see, e.g., [20]), but has so far been overlooked in the existing literature on probabilistic PDA. The idea behind this method is to construct a BPA with stack symbols of the form $\langle pXq \rangle$ for all $p, q \in Q$ and $X \in \Gamma$. Roughly speaking, such a triple corresponds to terminating paths from pX to $q\varepsilon$. Subsequently, transitions of the BPA are induced by transitions of the PDA in a way corresponding to this intuition. For example, a transition of the form $pX \hookrightarrow rYZ$ induces transitions of the form $\langle pXq \rangle \hookrightarrow \langle rYs \rangle \langle sZq \rangle$ for all $s \in Q$. Then each path from pX to $q\varepsilon$ maps naturally to a path from $\langle pXq \rangle$ to ε. This construction can also be applied in the probabilistic setting by assigning probabilities to transitions so that the probability of the corresponding paths is preserved. We also deal with nonterminating runs by introducing new stack symbols of the form $\langle pX\uparrow \rangle$.

Formally, the stack alphabet of $\Delta_\bullet$ is defined as follows: For every $pX \in Q \times \Gamma$ such that $[pX\uparrow] > 0$ we add a stack symbol $\langle pX\uparrow \rangle$, and for every $pXq \in Q \times \Gamma \times Q$ such that

[6] The "BPA" acronym stands for "Basic Process Algebra" and it is used mainly for historical reasons. pBPA are closely related to stochastic context-free grammars and are also called *1-exit recursive Markov chains* (see, e.g., [18]).

$[pXq] > 0$ we add a stack symbol $\langle pXq \rangle$. Note that the stack alphabet of $\Delta_\bullet$ is effectively constructible in polynomial space by applying the results of [13, 18].

Now we construct the rules $\hookrightarrow_\bullet$ of $\Delta_\bullet$. For all $\langle pXq \rangle$ we have the following rules:

- if $pX \stackrel{x}{\hookrightarrow} rYZ$ in Δ, then for all $s \in Q$ such that $y = x \cdot [rYs] \cdot [sZq] > 0$ we put $\langle pXq \rangle \xhookrightarrow{y/[pXq]}_\bullet \langle rYs \rangle \langle sZq \rangle$;
- if $pX \stackrel{x}{\hookrightarrow} rY$ in Δ, where $y = x \cdot [rYq] > 0$, we put $\langle pXq \rangle \xhookrightarrow{y/[pXq]}_\bullet \langle rYq \rangle$;
- if $pX \stackrel{x}{\hookrightarrow} q\varepsilon$ in Δ, we put $\langle pXq \rangle \xhookrightarrow{x/[pXq]}_\bullet \varepsilon$.

For all $\langle pX{\uparrow} \rangle$ we have the following rules:

- if $pX \stackrel{x}{\hookrightarrow} rYZ$ in Δ, then for every $s \in Q$ where $y = x \cdot [rYs] \cdot [sZ{\uparrow}] > 0$ we add $\langle pX{\uparrow} \rangle \xhookrightarrow{y/[pX\uparrow]}_\bullet \langle rYs \rangle \langle sZ{\uparrow} \rangle$;
- for all $qY \in Q \times \Gamma$ where $x = [qY{\uparrow}] \cdot \sum_{pX \hookrightarrow qY\beta} Prob(pX \hookrightarrow qY\beta) > 0$, we add $\langle pX{\uparrow} \rangle \xhookrightarrow{x/[pX\uparrow]}_\bullet \langle qY{\uparrow} \rangle$.

Note that the transition probabilities of $\Delta_\bullet$ may take irrational values. Still, the construction of $\Delta_\bullet$ is to some extent "effective" due to the following proposition:

Proposition 3 ([13, 18]). *Let $\Delta = (Q, \Gamma, \hookrightarrow, Prob)$ be a pPDA. Let $pXq \in Q \times \Gamma \times Q$. There is a formula $\Phi(x)$ of $ExTh(\mathbb{R})$ (the existential theory of the reals) with one free variable x such that the length of $\Phi(x)$ is polynomial in $|\Delta|$ and $\Phi(x/r)$ is valid iff $r = [pXq]$.*

Using Proposition 3, one can compute formulae of $ExTh(\mathbb{R})$ that "encode" transition probabilities of $\Delta_\bullet$. Moreover, these probabilities can be effectively approximated up to an arbitrarily small error by employing either the decision procedure for $ExTh(\mathbb{R})$ [9] or by using Newton's method [11, 21, 12].

Example 4. Consider a pPDA Δ with two control states, p, q, one stack symbol, X, and the following transition rules:

$$pX \stackrel{a}{\hookrightarrow} qXX,\ pX \xhookrightarrow{1-a} q\varepsilon,\ qX \stackrel{b}{\hookrightarrow} pXX,\ qX \xhookrightarrow{1-b} p\varepsilon,$$

where both a, b are greater than $1/2$. Apparently, $[pXp] = [qXq] = 0$. Using results of [13] one can easily verify that $[pXq] = (1-a)/b$ and $[qXp] = (1-b)/a$. Thus $[pX{\uparrow}] = (a+b-1)/b$ and $[qX{\uparrow}] = (a+b-1)/a$. Thus the stack symbols of $\Delta_\bullet$ are $\langle pXq \rangle, \langle qXp \rangle, \langle pX{\uparrow} \rangle, \langle qX{\uparrow} \rangle$. The transition rules of $\Delta_\bullet$ are:

$$\begin{array}{llll}
\langle pXq \rangle \xhookrightarrow{1-b}_\bullet \langle qXp \rangle \langle pXq \rangle & \langle pXq \rangle \stackrel{b}{\hookrightarrow}_\bullet \varepsilon & \langle qXp \rangle \xhookrightarrow{1-a}_\bullet \langle pXq \rangle \langle qXp \rangle & \langle qXp \rangle \stackrel{a}{\hookrightarrow}_\bullet \varepsilon \\
\langle pX{\uparrow} \rangle \xhookrightarrow{1-b}_\bullet \langle qXp \rangle \langle pX{\uparrow} \rangle & \langle pX{\uparrow} \rangle \stackrel{b}{\hookrightarrow}_\bullet \langle qX{\uparrow} \rangle & \langle qX{\uparrow} \rangle \xhookrightarrow{1-a}_\bullet \langle pXq \rangle \langle qX{\uparrow} \rangle & \langle qX{\uparrow} \rangle \stackrel{a}{\hookrightarrow}_\bullet \langle pX{\uparrow} \rangle
\end{array}$$

As both a, b are greater than $1/2$, the resulting pBPA has a tendency to remove symbols rather than add symbols. Thus both $\langle pXq \rangle$ and $\langle qXp \rangle$ terminate with probability 1.

When studying long-run properties of pPDA (such as ω-regular properties or limit-average properties), one usually assumes that the runs are initiated in a configuration p_0X_0 which cannot terminate, i.e., $[p_0X_0{\uparrow}] = 1$. Under this assumption, the probability spaces over $Run[M_\Delta](p_0X_0)$ and $Run[M_{\Delta_\bullet}](\langle p_0X_0{\uparrow} \rangle)$ are "isomorphic" w.r.t. all properties that depend only on the control states and the top-of-the-stack symbols of the configurations visited along a run. This is formalized in our next proposition.

Proposition 5. *Let $p_0X_0 \in Q\times\Gamma$ such that $[p_0X_0{\uparrow}] = 1$. Then there is a partial function $\Upsilon : Run[M_\Delta](p_0X_0) \to Run[M_{\Delta_\bullet}](\langle p_0X_0{\uparrow}\rangle)$ such that for every $w \in Run[M_\Delta](p_0X_0)$, where $\Upsilon(w)$ is defined, and every $n \in \mathbb{N}$ we have the following: if $w(n) = qY\beta$, then $\Upsilon(w)(n) = \langle qY\dagger\rangle\gamma$, where $\dagger$ is either an element of Q or $\uparrow$. Further, for every measurable set of runs $R \subseteq Run[M_{\Delta_\bullet}](\langle p_0X_0{\uparrow}\rangle)$ we have that $\Upsilon^{-1}(R)$ is measurable and $\mathcal{P}(R) = \mathcal{P}(\Upsilon^{-1}(R))$.*

As for terminating runs, observe that the "terminating" symbols of the form $\langle pXq\rangle$ do not depend on the "nonterminating" symbols of the form $\langle pX{\uparrow}\rangle$, i.e., if we restrict $\Delta_\bullet$ just to terminating symbols, we again obtain a pBPA. A straightforward computation reveals the following proposition about terminating runs that is crucial for our results presented in the next section.

Proposition 6. *Let $pXq \in Q \times \Gamma \times Q$ and $[pXq] > 0$. Then almost all runs of $M_{\Delta_\bullet}$ initiated in $\langle pXq\rangle$ terminate, i.e., reach ε. Further, for all $n \in \mathbb{N}$ we have that*

$$\mathcal{P}(\mathbf{T}_{pX} = n \mid Run(pXq)) \quad = \quad \mathcal{P}(\mathbf{T}_{\langle pXq\rangle} = n \mid Run(\langle pXq\rangle))$$

Observe that this proposition, together with a very special form of rules in $\Delta_\bullet$, implies that all configurations reachable from a nonterminating configuration p_0X_0 have the form $\alpha\langle qY{\uparrow}\rangle$, where α terminates almost surely and $\langle qY{\uparrow}\rangle$ never terminates. It follows that such a pBPA can be transformed into a finite-state Markov chain (whose states are the nonterminating symbols) which is allowed to make recursive calls that almost surely terminate (using rules of the form $\langle pX{\uparrow}\rangle \hookrightarrow \langle rZq\rangle\langle qY{\uparrow}\rangle$). This observation is very useful when investigating the properties of nonterminating runs, and many of the existing results about pPDA can be substantially simplified using this result.

3.2 Analysis of pBPA

In this section we establish the promised tight tail bounds for termination probability. By virtue of Proposition 6, it suffices to analyze pBPA where each stack symbol terminates with probability 1. In what follows we assume that Δ is such a pBPA, and we also fix an initial stack symbol X_0. For $X, Y \in \Gamma$, we say that X *depends directly on* Y, if there is a rule $X \hookrightarrow \alpha$ such that Y occurs in α. Further, we say that X *depends on* Y, if either X depends directly on Y, or X depends directly on a symbol $Z \in \Gamma$ which depends on Y. One can compute, in linear time, the directed acyclic graph (DAG) of strongly connected components (SCCs) of the dependence relation. The *height* of this DAG, denoted by h, is defined as the longest distance between a top SCC and a bottom SCC plus 1 (i.e., $h = 1$ if there is only one SCC). We can safely assume that all symbols on which X_0 does not depend were removed from Δ. We abbreviate $\mathcal{P}(\mathbf{T}_{X_0} \geq n \mid Run(X_0))$ to $\mathcal{P}(\mathbf{T}_{X_0}{\geq}n)$, and we use p_{min} to denote $\min\{p \mid X \overset{p}{\hookrightarrow} \alpha \text{ in } \Delta\}$. Here is our main result:

Theorem 7. *Let Δ be a pBPA with stack alphabet Γ where every stack symbol terminates with probability one. Assume that $X_0 \in \Gamma$ depends on all $X \in \Gamma \setminus \{X_0\}$, and let $p_{min} = \min\{p \mid X \overset{p}{\hookrightarrow} \alpha \text{ in } \Delta\}$. Then one of the following is true:*

(1) $\mathcal{P}(\mathbf{T}_{X_0}{\geq}2^{|\Gamma|}) = 0$.
(2) $\mathbb{E}[\mathbf{T}_{X_0}]$ *is* **finite** *and for all $n \in \mathbb{N}$ with $n \geq 2\mathbb{E}[\mathbf{T}_{X_0}]$ we have that*

$$p_{min}^n \quad \leq \quad \mathcal{P}(\mathbf{T}_{X_0}{\geq}n) \quad \leq \quad \exp\left(1 - \frac{n}{8E_{max}^2}\right)$$

where $E_{max} = \max_{X\in\Gamma} \mathbb{E}[\mathbf{T}_X]$.

(3) $\mathbb{E}[\mathbf{T}_{X_0}]$ *is* **infinite** *and there is* $n_0 \in \mathbb{N}$ *such that for all* $n \geq n_0$ *we have that*

$$c/n \quad \leq \quad \mathcal{P}(\mathbf{T}_{X_0} \geq n) \quad \leq \quad d_1/n^{d_2}$$

where $d_1 = 18h|\Gamma|/p_{min}^{3|\Gamma|}$, *and* $d_2 = 1/(2^{h+1}-2)$. *Here, h is the height of the DAG of SCCs of the dependence relation, and c is a suitable positive constant depending on* Δ.

More colloquially, Theorem 7 states that Δ satisfies either (1) or (2) or (3), where (1) is when Δ does not have any long terminating runs; and (2) resp. (3) is when the expected termination time is finite (resp. infinite) and the probability of performing a terminating run of length n decreases exponentially (resp. polynomially) in n.

One can effectively distinguish between the three cases set out in Theorem 7. More precisely, case (1) can be recognized in polynomial time by looking only at the structure of the pBPA, i.e., disregarding the probabilities. Determining whether $\mathbb{E}[\mathbf{T}_{X_0}]$ is finite or infinite can be done in polynomial space by employing the decision procedure for $\mathbb{E}[\mathbf{T}_{X_0}]$ and the results of [14, 2]. This holds even if the transition probabilities of Δ are represented just symbolically by formulae of *ExTh*($\mathbb{R}$) (see Proposition 3).

The proof of Theorem 7 is based on designing suitable martingales that are used to analyze the concentration of the termination probability. Recall that a *martingale* is an infinite sequence of random variables $m^{(0)}, m^{(1)}, \ldots$ such that, for all $i \in \mathbb{N}$, $\mathbb{E}[|m^{(i)}|] < \infty$, and $\mathbb{E}[m^{(i+1)} \mid m^{(1)}, \ldots, m^{(i)}] = m^{(i)}$ almost surely. If $|m^{(i)} - m^{(i-1)}| < c_i$ for all $i \in \mathbb{N}$, then we have the following *Azuma's inequality* (see, e.g., [27]):

$$\mathcal{P}(m^{(n)} - m^{(0)} \geq t) \quad \leq \quad \exp\left(\frac{-t^2}{2\sum_{k=1}^{n} c_k^2}\right)$$

Due to space restrictions we can only sketch the proof of Theorem 7 (see [8] for details).

Proof sketch for the upper bound of Theorem 7(2). Observe that if $\mathbb{E}[\mathbf{T}_{X_0}]$ is finite, then $\mathbb{E}[\mathbf{T}_Y]$ is finite for every $Y \in \Gamma$ (here we use the assumption that X_0 depends on Y). Further, for every configuration $\beta\gamma$ reachable from X_0 we have that $\mathbb{E}\left[\mathbf{T}_{\beta\gamma}\right] = \mathbb{E}\left[\mathbf{T}_{\beta}\right] + \mathbb{E}\left[\mathbf{T}_{\gamma}\right]$. Hence, $\mathbb{E}[\mathbf{T}_\alpha] < \infty$ for every $\alpha \in \Gamma^*$. Now observe that, for every $\alpha \in \Gamma^*$ such that $\alpha \neq \varepsilon$, performing one transition from α decreases the expected termination time by one on average (here we need that $\mathbb{E}[\mathbf{T}_\alpha] < \infty$ and α terminates with probability one). Let $w \in Run(X_0)$. We denote by $I(w)$ the maximal number $j \geq 0$ such that $w(j-1) \neq \varepsilon$. For every $i \geq 0$, we put

$$m^{(i)}(w) \quad = \quad \mathbb{E}[\mathbf{T}_{w(i)}] + \min\{i, I(w)\}$$

It is easy to see that $\mathbb{E}\left[m^{(i+1)} \mid m^{(i)}\right] = m^{(i)}$, i.e., $m^{(0)}, m^{(1)}, \ldots$ is a martingale. A full proof of this claim is given in [8].

Let $E_{max} = \max_{X \in \Gamma} \mathbb{E}[\mathbf{T}_X]$, and let $n \geq 2\mathbb{E}[\mathbf{T}_{X_0}]$. By applying Azuma's inequality we obtain

$$\mathcal{P}(m^{(n)} - \mathbb{E}[\mathbf{T}_{X_0}] \geq n - \mathbb{E}[\mathbf{T}_{X_0}]) \quad \leq \quad \exp\left(\frac{-(n-\mathbb{E}[\mathbf{T}_{X_0}])^2}{2\sum_{k=1}^{n}(2E_{max})^2}\right) \quad \leq \quad \exp\left(\frac{2\mathbb{E}[\mathbf{T}_{X_0}] - n}{8E_{max}^2}\right).$$

For every $w \in Run(X_0)$ we have that $w(n) \neq \varepsilon$ implies $m^{(n)} \geq n$. It follows:

$$\mathcal{P}(\mathbf{T}_{X_0} \geq n) \quad \leq \quad \mathcal{P}(m^{(n)} \geq n) \quad \leq \quad \exp\left(\frac{2\mathbb{E}[\mathbf{T}_{X_0}] - n}{8E_{max}^2}\right) \quad \leq \quad \exp\left(1 - \frac{n}{8E_{max}^2}\right).$$

Proof sketch for the upper bound of Theorem 7(3). Assume that $\mathbb{E}[\mathbf{T}_{X_0}]$ is infinite. To give some idea of the (quite involved) proof, let us first consider a simple pBPA Δ with $\Gamma = \{X\}$ and the rules $X \overset{1/2}{\hookrightarrow} XX$ and $X \overset{1/2}{\hookrightarrow} \varepsilon$. In fact, Δ is closely related to a simple random walk starting at 1, for which the time until it hits 0 can be exactly analyzed (see, e.g., [27]). Clearly, we have $h = |\Gamma| = 1$ and $p_{min} = 1/2$. Theorem 7(3) implies $\mathcal{P}(\mathbf{T}_X \geq n) \in O(1/\sqrt{n})$. Let us sketch why this upper bound holds.

Let $\theta > 0$, define $g(\theta) := \frac{1}{2} \cdot \exp(-\theta \cdot (-1)) + \frac{1}{2} \cdot \exp(-\theta \cdot (+1))$, and define for a run $w \in Run(X)$ the sequence

$$m_\theta^{(i)}(w) = \begin{cases} \exp(-\theta \cdot |w(i)|)/g(\theta)^i & \text{if } i = 0 \text{ or } w(i-1) \neq \varepsilon \\ m_\theta^{(i-1)}(w) & \text{otherwise.} \end{cases}$$

One can show (cf. [27]) that $m_\theta^{(0)}, m_\theta^{(1)}, \ldots$ is a martingale, i.e., $\mathbb{E}\left[m_\theta^{(i)} \mid m_\theta^{(i-1)}\right] = m_\theta^{(i-1)}$ for all $\theta > 0$. Our proof crucially depends on some analytic properties of the function $g : \mathbb{R} \to \mathbb{R}$: It is easy to verify that $1 = g(0) < g(\theta)$ for all $\theta > 0$, and $0 = g'(0)$, and $1 = g''(0)$. One can show that Doob's Optional-Stopping Theorem (see Theorem 10.10 (ii) of [27]) applies, which implies $m_\theta^{(0)} = \mathbb{E}\left[m_\theta^{(\mathbf{T}_X)}\right]$. It follows that for all $n \in \mathbb{N}$ and $\theta > 0$ we have that

$$\begin{aligned} \exp(-\theta) &= m_\theta^{(0)} = \mathbb{E}\left[m_\theta^{(\mathbf{T}_X)}\right] = \mathbb{E}\left[g(\theta)^{-\mathbf{T}_X}\right] = \sum_{i=0}^{\infty} \mathcal{P}(\mathbf{T}_X = i) \cdot g(\theta)^{-i} \\ &\leq \sum_{i=0}^{n-1} \mathcal{P}(\mathbf{T}_X = i) \cdot 1 + \sum_{i=n}^{\infty} \mathcal{P}(\mathbf{T}_X = i) \cdot g(\theta)^{-n} = 1 - \mathcal{P}(\mathbf{T}_X \geq n) + \mathcal{P}(\mathbf{T}_X \geq n) \cdot g(\theta)^{-n} \end{aligned}$$

Rearranging this inequality yields $\mathcal{P}(\mathbf{T}_X \geq n) \leq \frac{1-\exp(-\theta)}{1-g(\theta)^{-n}}$, from which one obtains, setting $\theta := 1/\sqrt{n}$, and using the mentioned properties of g and several applications of l'Hopital's rule, that $\mathcal{P}(\mathbf{T}_X \geq n) \in O(1/\sqrt{n})$.

Next we sketch how we generalize this proof to pBPA that consist of only one SCC, but have more than one stack symbol. In this case, the term $|w(i)|$ in the definition of $m_\theta^{(i)}(w)$ needs to be replaced by the sum of *weights* of the symbols in $w(i)$. Each $Y \in \Gamma$ has a weight which is drawn from the dominant eigenvector of a certain matrix, which is characteristic for Δ. Perron-Frobenius theory guarantees the existence of a suitable weight vector $\boldsymbol{u} \in \mathbb{R}_+^\Gamma$. The function g consequently needs to be replaced by a function g_Y for each $Y \in \Gamma$. We need to keep the property that $g_Y''(0) > 0$. Intuitively, this means that Δ must have, for each $Y \in \Gamma$, a rule $Y \hookrightarrow \alpha$ such that Y and α have different weights. This can be accomplished by transforming Δ into a certain normal form.

Finally, we sketch how the proof is generalized to pBPA with more than one SCC. For simplicity, assume that Δ has only two stack symbols, say X and Y, where X depends on Y, but Y does not depend on X. Let us change the execution order of pBPA as follows: whenever a rule with $\alpha \in \Gamma^*$ on the right hand side fires, then all X-symbols in α are added on top of the stack, but all Y-symbols are added at the *bottom* of the stack. This change does not influence the termination time of pBPA, but it allows to decompose runs into two phases: an X-phase where X-rules are executed which may produce Y-symbols or further X-symbols; and a Y-phase where Y-rules are executed which may produce further Y-symbols but no X-symbols, because Y does not depend on X. Arguing only qualitatively, assume that $\mathbf{T}_X$ is "large". Then either (a) the X-phase is "long" or (b) the X-phase is "short", but the Y-phase is "long". For the probability of event (a) one can give an upper bound using the bound for one SCC, because the produced Y-symbols can be ignored. For event (b), observe that if the X-phase is short, then only few Y-symbols can be created during the X-phase. For a bound on the probability of event (b) we need a bound on the probability that a pBPA with one SCC and a "short" initial configuration takes a "long" time to terminate. The previously sketched proof for an initial configuration with a single stack symbol can be suitably generalized to handle other "short" configurations. The details can be found in [8].

Finally, the following proposition shows that the upper bound in Theorem 7 (3) cannot be substantially tightened.

Proposition 8. *Let Δ_h be the pBPA with $\Gamma_h = \{X_1, \dots, X_h\}$ and the following rules:*

$$X_h \xhookrightarrow{1/2} X_h X_h\,,\ X_h \xhookrightarrow{1/2} X_{h-1}\,,\ \dots\,,\ X_2 \xhookrightarrow{1/2} X_2 X_2\,,\ X_2 \xhookrightarrow{1/2} X_1\,,\ X_1 \xhookrightarrow{1/2} X_1 X_1\,,\ X_1 \xhookrightarrow{1/2} \varepsilon$$

Then $[X_h] = 1$, $\mathbb{E}\left[\mathbf{T}_{X_h}\right] = \infty$, and there is $c_h > 0$ with $\mathcal{P}(\mathbf{T}_{X_h} \ge n) \ge c_h \cdot n^{-1/2^h}$ for all $n \ge 1$.

4 Conclusions and Future work

We have provided a reduction from stateful to stateless pPDA which gives new insights into the theory of pPDA and at the same time simplifies it substantially. We have used this reduction and martingale theory to exhibit a dichotomy result that precisely characterizes the distribution of the termination time in terms of its expected value.

Although the bounds presented in this paper are asymptotically optimal, there is still space for improvements. We conjecture that the lower bound of Theorem 7 (3) can be strengthened to $\Omega(1/\sqrt{n})$. We also conjecture that our results can be extended to more general reward-based models, where each configuration is assigned a nonnegative reward and the total reward accumulated in a given service is considered instead of its length. This is particularly challenging if the rewards are unbounded (for example, the reward assigned to a given configuration may correspond to the total memory allocated by the procedures in the current call stack). Full answers to these questions would generalize some of the existing deep results about simpler models, and probably reveal an even richer underlying theory of pPDA which is still undiscovered.

Acknowledgment. The authors thank Javier Esparza for useful suggestions.

References

1. D. Bini, G. Latouche, and B. Meini. *Numerical methods for Structured Markov Chains*. Oxford University Press, 2005.
2. T. Brázdil. *Verification of Probabilistic Recursive Sequential Programs*. PhD thesis, Masaryk University, Faculty of Informatics, 2007.
3. T. Brázdil, V. Brožek, J. Holeček, and A. Kučera. Discounted properties of probabilistic pushdown automata. In *Proceedings of LPAR 2008*, volume 5330 of *Lecture Notes in Computer Science*, pages 230–242. Springer, 2008.
4. T. Brázdil, V. Brožek, and K. Etessami. One-counter stochastic games. In *Proceedings of FST&TCS 2010*, volume 8 of *Leibniz International Proceedings in Informatics*, pages 108–119. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2010.

5. T. Brázdil, V. Brožek, K. Etessami, A. Kučera, and D. Wojtczak. One-counter Markov decision processes. In *Proceedings of SODA 2010*, pages 863–874. SIAM, 2010.
6. T. Brázdil, J. Esparza, and A. Kučera. Analysis and prediction of the long-run behavior of probabilistic sequential programs with recursion. In *Proceedings of FOCS 2005*, pages 521–530. IEEE Computer Society Press, 2005.
7. T. Brázdil, S. Kiefer, and A. Kučera. Efficient analysis of probabilistic programs with an unbounded counter. *CoRR*, abs/1102.2529, 2011.
8. T. Brázdil, S. Kiefer, A. Kučera, and I. Hutařová Vařeková. Runtime analysis of probabilistic programs with unbounded recursion. *CoRR*, abs/1007.1710, 2010.
9. J. Canny. Some algebraic and geometric computations in PSPACE. In *Proceedings of STOC'88*, pages 460–467. ACM Press, 1988.
10. D.P. Dubhashi and A. Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009.
11. J. Esparza, S. Kiefer, and M. Luttenberger. Convergence thresholds of Newton's method for monotone polynomial equations. In *STACS 2008*, pages 289–300, 2008.
12. J. Esparza, S. Kiefer, and M. Luttenberger. Computing the least fixed point of positive polynomial systems. *SIAM Journal on Computing*, 39(6):2282–2335, 2010.
13. J. Esparza, A. Kučera, and R. Mayr. Model-checking probabilistic pushdown automata. In *Proceedings of LICS 2004*, pages 12–21. IEEE Computer Society Press, 2004.
14. J. Esparza, A. Kučera, and R. Mayr. Quantitative analysis of probabilistic pushdown automata: Expectations and variances. In *Proceedings of LICS 2005*, pages 117–126. IEEE Computer Society Press, 2005.
15. K. Etessami, D. Wojtczak, and M. Yannakakis. Quasi-birth-death processes, tree-like QBDs, probabilistic 1-counter automata, and pushdown systems. In *Proceedings of 5th Int. Conf. on Quantitative Evaluation of Systems (QEST'08)*. IEEE Computer Society Press, 2008.
16. K. Etessami and M. Yannakakis. Algorithmic verification of recursive probabilistic systems. In *Proceedings of TACAS 2005*, volume 3440 of *Lecture Notes in Computer Science*, pages 253–270. Springer, 2005.
17. K. Etessami and M. Yannakakis. Checking LTL properties of recursive Markov chains. In *Proceedings of 2nd Int. Conf. on Quantitative Evaluation of Systems (QEST'05)*, pages 155–165. IEEE Computer Society Press, 2005.
18. K. Etessami and M. Yannakakis. Recursive Markov chains, stochastic grammars, and monotone systems of nonlinear equations. *Journal of the Association for Computing Machinery*, 56, 2009.
19. T.E. Harris. *The Theory of Branching Processes*. Springer, 1963.
20. J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979.
21. S. Kiefer, M. Luttenberger, and J. Esparza. On the convergence of Newton's method for monotone systems of polynomial equations. In *STOC 2007*, pages 217–226, 2007.
22. G. Latouche and V. Ramaswami. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. ASA-SIAM series on statistics and applied probability, 1999.
23. C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999.
24. R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 2006.
25. A.G. Pakes. Some limit theorems for the total progeny of a branching process. *Advances in Applied Probability*, 3(1):176–192, 1971.
26. M. P. Quine and W. Szczotka. Generalisations of the Bienayme-Galton-Watson branching process via its representation as an embedded random walk. *The Annals of Applied Probability*, 4(4):1206–1222, 1994.
27. D. Williams. *Probability with Martingales*. Cambridge University Press, 1991.