Computing Expected Runtimes for Constant Probability Programs*

Jürgen Giesl[1], Peter Giesl[2], and Marcel Hark[1]

[1] LuFG Informatik 2, RWTH Aachen University, Germany
{giesl,marcel.hark}@cs.rwth-aachen.de
[2] Department of Mathematics, University of Sussex, UK
p.a.giesl@sussex.ac.uk

Abstract. We introduce the class of *constant probability (CP) programs* and show that classical results from probability theory directly yield a simple decision procedure for (positive) almost sure termination of programs in this class. Moreover, asymptotically tight bounds on their expected runtime can always be computed easily. Based on this, we present an algorithm to infer the *exact* expected runtime of any CP program.

Keywords: Probabilistic Programs · Expected Runtimes · (Positive) Almost Sure Termination · Complexity · Decidability

1 Introduction

Probabilistic programs are used to describe randomized algorithms and probability distributions, with applications in many areas. As an example, consider the well-known program which models the race between a tortoise and a hare (see, e.g., [11,24,30]). As long as the tortoise (variable t) is not behind the hare (variable h), it does one step in each iteration. With probability $\frac{1}{2}$, the hare stays at its position and with probability $\frac{1}{2}$ it does a random number of steps uniformly chosen between 0 and 10. The race ends when the hare is in front of the tortoise. Here, the hare wins with probability one and the technique of [30] infers the upper bound $\frac{2}{3} \cdot \max(t-h+9, 0)$ on the expected number of loop iterations. Thus, the program is positively almost surely terminating.

```
while (h ≤ t) {
  t = t + 1;
  {h = h + Unif(0, 10)} ⊕_{1/2} {h = h};
}
```

Sect. 2 recapitulates preliminaries on probabilistic programs and on the connection between their expected runtime and their corresponding recurrence equation. Then we show in Sect. 3 and 4 that classical results on random walk theory directly yield a very simple decision procedure for (positive) almost sure termination of *CP programs* like the tortoise and hare example. In this way, we also obtain asymptotically tight bounds on the expected runtime of any CP program. Based on these bounds, in Sect. 5 we develop the first algorithm to

* Supported by the DFG Research Training Group 2236 UnRAVeL and the London Mathematical Society (Grant 41662, Research in Pairs).

compute closed forms for the *exact* expected runtime of such programs. In Sect. 6, we present its implementation in our tool KoAT [10] and discuss related and future work. We refer to the appendix for a collection of examples to illustrate the application of our algorithm and for all proofs.

2 Expected Runtimes of Probabilistic Programs

Example 1 (Tortoise and Hare). *The program $\mathcal{P}_{race}$ on the right formulates the race of the tortoise and the hare as a* CP *program. In the loop guard, we use the scalar product $(1,-1) \bullet (t,h)$ which stands for $t-h$. Exactly one of the instructions with numbers in brackets $[\dots]$ is executed in each loop iteration and the number indicates the probability that the corresponding instruction is chosen.*

```
while ((1,−1) • (t,h) > −1) {
  (t,h) = (t,h) + (1,0)     [6/11];
  (t,h) = (t,h) + (1,1)     [1/22];
  (t,h) = (t,h) + (1,2)     [1/22];
  (t,h) = (t,h) + (1,3)     [1/22];
    ⋮
  (t,h) = (t,h) + (1,10)    [1/22];
}
```

We now define the kind of probabilistic programs considered in this paper.

Definition 2 (Probabilistic Program). *A program has the form on the right, where $\boldsymbol{x} = (x_1, \dots, x_r)$ for some $r \geq 1$ is a tuple of pairwise different program variables, $\boldsymbol{a}, \boldsymbol{c}_1, \dots, \boldsymbol{c}_n \in \mathbb{Z}^r$ are tuples of integers, the $\boldsymbol{c}_j$ are pairwise distinct, $b \in \mathbb{Z}$, $\bullet$ is the scalar product (i.e., $(a_1, \dots, a_r) \bullet (x_1, \dots, x_r) = a_1 \cdot x_1 + \dots + a_r \cdot x_r$), and $\boldsymbol{d} \in \mathbb{Z}^r$ with $\boldsymbol{a} \bullet \boldsymbol{d} \leq b$. We require $p_{\boldsymbol{c}_1}(\boldsymbol{x}), \dots, p_{\boldsymbol{c}_n}(\boldsymbol{x}), p'(\boldsymbol{x}) \in \mathbb{R}_{\geq 0} = \{r \in \mathbb{R} \mid r \geq 0\}$ and $\sum_{1 \leq j \leq n} p_{\boldsymbol{c}_j}(\boldsymbol{x}) + p'(\boldsymbol{x}) = 1$ for all $\boldsymbol{x} \in \mathbb{Z}^r$. It is a program with* direct termination *if there is an $\boldsymbol{x} \in \mathbb{Z}^r$ with $\boldsymbol{a} \bullet \boldsymbol{x} > b$ and $p'(\boldsymbol{x}) > 0$. If all probabilities are constant, i.e., if there are $p_{\boldsymbol{c}_1}, \dots, p_{\boldsymbol{c}_n}, p' \in \mathbb{R}_{\geq 0}$ such that $p_{\boldsymbol{c}_j}(\boldsymbol{x}) = p_{\boldsymbol{c}_j}$ and $p'(\boldsymbol{x}) = p'$ for all $1 \leq j \leq n$ and all $\boldsymbol{x} \in \mathbb{Z}^r$, we call it a* constant probability (CP) program*.*

```
while (a • x > b) {
  x = x + c_1    [p_c1(x)];
    ⋮
  x = x + c_n    [p_cn(x)];
  x = d          [p'(x)];
}
```

Such a program means that the integer variables $\boldsymbol{x}$ are changed to $\boldsymbol{x} + \boldsymbol{c}_j$ with probability $p_{\boldsymbol{c}_j}(\boldsymbol{x})$. For inputs $\boldsymbol{x}$ with $\boldsymbol{a} \bullet \boldsymbol{x} \leq b$ the program terminates immediately. Note that the program in Ex. 1 has *no* direct termination (i.e., $p'(\boldsymbol{x}) = 0$ for all $\boldsymbol{x} \in \mathbb{Z}^r$). Since the values of the program variables only depend on their values in the previous loop iteration, our programs correspond to *Markov Chains* [32] and they are related to *random walks* [17,21,33], cf. the appendix for details.

Clearly, in general termination is undecidable and closed forms for the runtimes of programs are not computable. Thus, decidability results can only be obtained for suitably restricted forms of programs. Our class nevertheless includes many examples that are often regarded in the literature on probabilistic programs. So while other approaches are concerned with *incomplete* techniques to analyze termination and complexity, we investigate classes of probabilistic programs where one can *decide* the termination behavior, *always* find complexity bounds, and even compute the expected runtime *exactly*. Our decision procedure could be integrated into general tools for termination and complexity analysis of probabilistic programs: As soon as one has to investigate a sub-program that falls into our class, one can use the decision procedure to compute its exact runtime. Our contributions provide a starting point for such results and the considered class of programs can be extended further in future work.

In probability theory (see, e.g., [2]), given a set Ω of possible events, the goal is to measure the probability that events are in certain subsets of Ω. To this end, one regards a set $\mathfrak{F}$ of subsets of Ω, such that $\mathfrak{F}$ contains the full set Ω and is closed under complement and countable unions. Such a set $\mathfrak{F}$ is called a *σ-field*, and a pair of Ω and a corresponding σ-field $\mathfrak{F}$ is called a *measurable space*.

A *probability space* $(\Omega, \mathfrak{F}, \mathbb{P})$ extends a measurable space $(\Omega, \mathfrak{F})$ by a *probability measure* $\mathbb{P}$ which maps every set from $\mathfrak{F}$ to a number between 0 and 1, where $\mathbb{P}(\Omega) = 1$, $\mathbb{P}(\varnothing) = 0$, and $\mathbb{P}(\biguplus_{j \geq 0} A_j) = \sum_{j \geq 0} \mathbb{P}(A_j)$ for any pairwise disjoint sets $A_0, A_1, \ldots \in \mathfrak{F}$. So $\mathbb{P}(A)$ is the probability that an event from Ω is in the subset A. In our setting, we use the probability space $((\mathbb{Z}^r)^\omega, \mathfrak{F}^{\mathbb{Z}^r}, \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0})$ arising from the standard cylinder-set construction of MDP theory, cf. App. B. Here, $(\mathbb{Z}^r)^\omega$ corresponds to all infinite sequences of program states and $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}$ is the probability measure induced by the program $\mathcal{P}$ when starting in the state $\boldsymbol{x}_0 \in \mathbb{Z}^r$. For example, if $A \subseteq (\mathbb{Z}^2)^\omega$ consists of all infinite sequences starting with $(5,1)$, $(6,1)$, $(7,6)$, then $\mathbb{P}^{\mathcal{P}_{race}}_{(5,1)}(A) = \frac{6}{11} \cdot \frac{1}{22} = \frac{3}{121}$. So, if one starts with $(5,1)$, then $\frac{3}{121}$ is the probability that the next two states are $(6,1)$ and $(7,6)$. Once a state is reached that violates the loop guard, then the probability to remain in this state is 1. Hence, if B contains all infinite sequences starting with $(7,8)$, $(7,8)$, then $\mathbb{P}^{\mathcal{P}_{race}}_{(7,8)}(B) = 1$. In the following, for any set of numbers M let $\overline{M} = M \cup \{\infty\}$.

Definition 3 (Termination Time). *For a program $\mathcal{P}$ as in Def. 2, its* termination time *is the random variable $T^{\mathcal{P}} : (\mathbb{Z}^r)^\omega \to \overline{\mathbb{N}}$ that maps every infinite sequence $\langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots \rangle$ to the first index j where $\boldsymbol{z}_j$ violates $\mathcal{P}$'s loop guard.*

Thus, $T^{\mathcal{P}_{race}}(\langle (5,1), (6,1), (7,8), (7,8), \ldots \rangle) = 2$ and $T^{\mathcal{P}_{race}}(\langle (5,1), (6,1), (5,6), (8,6), (9,6), \ldots \rangle) = \infty$ (i.e., this sequence always satisfies $\mathcal{P}_{race}$'s loop guard as the jth entry is $(5+j, 6)$ for $j \geq 3$). Now we can define the different notions of termination and the expected runtime of a probabilistic program. As usual, for any random variable X on a probability space $(\Omega, \mathfrak{F}, \mathbb{P})$, $\mathbb{P}(X = j)$ stands for $\mathbb{P}(X^{-1}(\{j\}))$. So $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} = j)$ is the probability that a sequence has termination time j. Similarly, $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} < \infty) = \sum_{j \in \mathbb{N}} \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} = j)$. The *expected value* $\mathbb{E}(X)$ of a random variable $X : \Omega \to \overline{\mathbb{N}}$ for a probability space $(\Omega, \mathfrak{F}, \mathbb{P})$ is the weighted average under the probability measure $\mathbb{P}$, i.e., $\mathbb{E}(X) = \sum_{j \in \overline{\mathbb{N}}} j \cdot \mathbb{P}(X = j)$, where $\infty \cdot 0 = 0$ and $\infty \cdot u = \infty$ for all $u \in \mathbb{N}_{>0}$.

Definition 4 (Termination and Expected Runtime). *A program $\mathcal{P}$ as in Def. 2 is* almost surely terminating (AST) *if $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} < \infty) = 1$ for any initial value $\boldsymbol{x}_0 \in \mathbb{Z}^r$. For any $\boldsymbol{x}_0 \in \mathbb{Z}^r$, its* expected runtime *$rt^{\mathcal{P}}_{\boldsymbol{x}_0}$ (i.e., the expected number of loop iterations) is defined as the expected value of the random variable $T^{\mathcal{P}}$ under the probability measure $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}$, i.e., $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{E}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(T^{\mathcal{P}}\right) =$*

$\sum_{j\in\mathbb{N}} j \cdot \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}}{=}j)$ *if* $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}}{<}\infty) = 1$, *and* $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{E}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(T^{\mathcal{P}}\right) = \infty$ *otherwise. The program* $\mathcal{P}$ *is* positively almost surely terminating (PAST) *if for any initial value* $\boldsymbol{x}_0 \in \mathbb{Z}^r$*, the expected runtime of* $\mathcal{P}$ *is finite, i.e., if* $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{E}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(T^{\mathcal{P}}\right) < \infty$*.*

Example 5 (Expected Runtime for $\mathcal{P}_{race}$*). By the observations in Sect. 4 we will infer that* $\frac{2}{3}\cdot(t-h+1) \le rt^{\mathcal{P}_{race}}_{(t,h)} \le \frac{2}{3}\cdot(t-h+1)+\frac{16}{3}$ *holds whenever* $t-h > -1$*, cf. Ex. 22. So the expected number of steps until termination is finite (and linear in the input variables) and thus,* $\mathcal{P}_{race}$ *is PAST. The algorithm in Sect. 5 will even be able to compute* $rt^{\mathcal{P}_{race}}_{(t,h)}$ *exactly, cf. Ex. 34.*

If the initial values $\boldsymbol{x}_0$ violate the loop guard, then the runtime is trivially 0.

Corollary 6 (Expected Runtime for Violating Initial Values). *For any program* $\mathcal{P}$ *as in Def. 2 and any* $\boldsymbol{x}_0 \in \mathbb{Z}^r$ *with* $\boldsymbol{a} \bullet \boldsymbol{x}_0 \le b$*, we have* $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = 0$*.*

To obtain our results, we use an alternative, well-known characterization of the expected runtime, cf. e.g., [4, 9, 16, 24–27, 32, 34]. To this end, we search for the *smallest* (or "*least*") solution of the recurrence equation that describes the runtime of the program as 1 plus the sum of the runtimes in the next loop iteration, multiplied with the corresponding probabilities. Here, functions are compared pointwise, i.e., for $f, g : \mathbb{Z}^r \to \overline{\mathbb{R}_{\ge 0}}$ we have $f \le g$ if $f(\boldsymbol{x}) \le g(\boldsymbol{x})$ holds for all $\boldsymbol{x} \in \mathbb{Z}^r$. So we search for the smallest function $f : \mathbb{Z}^r \to \overline{\mathbb{R}_{\ge 0}}$ that satisfies

$$f(\boldsymbol{x}) = \sum\nolimits_{1\le j\le n} p_{\boldsymbol{c}_j}(\boldsymbol{x}){\cdot}f(\boldsymbol{x}{+}\boldsymbol{c}_j){+}p'(\boldsymbol{x}){\cdot}f(\boldsymbol{d}){+}1 \quad \text{for all } \boldsymbol{x} \text{ with } \boldsymbol{a}\bullet\boldsymbol{x} > b. \tag{1}$$

Equivalently, we can search for the least fixpoint of the "expected runtime transformer" $\mathcal{L}^{\mathcal{P}}$ which transforms the left-hand side of (1) into its right-hand side.

Definition 7 **(**$\mathcal{L}^{\mathcal{P}}$, cf. [32]**).** *For* $\mathcal{P}$ *as in Def. 2, we define the* expected runtime transformer $\mathcal{L}^{\mathcal{P}} : (\mathbb{Z}^r \to \overline{\mathbb{R}_{\ge 0}}) \to (\mathbb{Z}^r \to \overline{\mathbb{R}_{\ge 0}})$*, where for any* $f : \mathbb{Z}^r \to \overline{\mathbb{R}_{\ge 0}}$*:*

$$\mathcal{L}^{\mathcal{P}}(f)(\boldsymbol{x}) = \begin{cases} \sum_{1\le j\le n} p_{\boldsymbol{c}_j}(\boldsymbol{x})\cdot f(\boldsymbol{x}+\boldsymbol{c}_j) + p'(\boldsymbol{x})\cdot f(\boldsymbol{d}) + 1, & \text{if } \boldsymbol{a}\bullet\boldsymbol{x} > b \\ f(\boldsymbol{x}), & \text{if } \boldsymbol{a}\bullet\boldsymbol{x} \le b \end{cases}$$

Example 8 (Expected Runtime Transformer for $\mathcal{P}_{race}$*). For* $\mathcal{P}_{race}$ *from Ex. 1,* $\mathcal{L}^{\mathcal{P}_{race}}$ *maps any function* $f : \mathbb{Z}^2 \to \overline{\mathbb{R}_{\ge 0}}$ *to* $\mathcal{L}^{\mathcal{P}_{race}}(f)$*, where* $\mathcal{L}^{\mathcal{P}_{race}}(f)(t,h) =$

$$\begin{cases} \frac{6}{11}\cdot f(t+1,h) + \frac{1}{22}\cdot \sum_{1\le j\le 10} f(t+1,h+j)+1, & \text{if } t-h > -1 \\ f(t,h), & \text{if } t-h \le -1 \end{cases} \tag{2}$$

Thm. 9 recapitulates that the least fixpoint of $\mathcal{L}^{\mathcal{P}}$ indeed yields an equivalent characterization of the expected runtime. In the following, let $\mathfrak{o} : \mathbb{Z}^r \to \overline{\mathbb{R}_{\ge 0}}$ be the function with $\mathfrak{o}(\boldsymbol{x}) = 0$ for all $\boldsymbol{x} \in \mathbb{Z}^r$.

Theorem 9 (Connection Between Expected Runtime and Least Fixpoint of $\mathcal{L}^{\mathcal{P}}$, cf. [32]**).** *For any* $\mathcal{P}$ *as in Def. 2, the expected runtime transformer* $\mathcal{L}^{\mathcal{P}}$ *is continuous. Thus, it has a least fixpoint* $\mathrm{lfp}(\mathcal{L}^{P}) : \mathbb{Z}^r \to \overline{\mathbb{R}_{\ge 0}}$ *with* $\mathrm{lfp}(\mathcal{L}^{\mathcal{P}}) = \sup\{\mathfrak{o}, \mathcal{L}^{\mathcal{P}}(\mathfrak{o}), (\mathcal{L}^{\mathcal{P}})^2(\mathfrak{o}), \ldots\}$*. Moreover, the least fixpoint of* $\mathcal{L}^{\mathcal{P}}$ *is the expected runtime of* $\mathcal{P}$*, i.e., for any* $\boldsymbol{x}_0 \in \mathbb{Z}^r$*, we have* $\mathrm{lfp}(\mathcal{L}^{\mathcal{P}})(\boldsymbol{x}_0) = rt^{\mathcal{P}}_{\boldsymbol{x}_0}$*.*

So the expected runtime $rt^{\mathcal{P}_{race}}_{(t,h)}$ can also be characterized as the smallest function $f : \mathbb{Z}^2 \to \overline{\mathbb{R}_{\geq 0}}$ satisfying $f(t,h) = (2)$, i.e., as the least fixpoint of $\mathcal{L}^{\mathcal{P}_{race}}$.

3 Expected Runtime of Programs with Direct Termination

We start with stating a decidability result for the case where for all $\boldsymbol{x}$ with $\boldsymbol{a} \bullet \boldsymbol{x} > b$, the probability $p'(\boldsymbol{x})$ for direct termination is at least p' for some $p' > 0$. Intuitively, these programs have a termination time whose distribution is closely related to the geometric distribution with parameter p' (which has expected value $\frac{1}{p'}$). By using the alternative characterization of $rt^{\mathcal{P}}_{\boldsymbol{x}_0}$ from Thm. 9, one obtains that such programs are *always PAST* and their expected runtime is indeed *bounded by the constant* $\frac{1}{p'}$. This result will be used in Sect. 5 when computing the *exact* expected runtime of such programs. The more involved case where $p'(\boldsymbol{x}) = 0$ is considered in Sect. 4.

Theorem 10 (PAST and Expected Runtime for Programs With Direct Termination). *Let $\mathcal{P}$ be a program as in Def. 2 where there is a $p' > 0$ such that $p'(\boldsymbol{x}) \geq p'$ for all $\boldsymbol{x} \in \mathbb{Z}^r$ with $\boldsymbol{a} \bullet \boldsymbol{x} > b$. Then $\mathcal{P}$ is PAST and its expected runtime is at most $\frac{1}{p'}$, i.e., $rt^{\mathcal{P}}_{\boldsymbol{x}_0} \leq \frac{1}{p'}$ if $\boldsymbol{a} \bullet \boldsymbol{x}_0 > b$, and $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = 0$ if $\boldsymbol{a} \bullet \boldsymbol{x}_0 \leq b$.*

$$
\begin{array}{ll}
\texttt{while } ((1,-1) \bullet (t,h) > -1) \ \{ & \\
\quad (t,h) = (t,h) + (1,0) & [\frac{9}{10}]; \\
\quad (t,h) = (7,8) & [\frac{1}{10}]; \\
\} &
\end{array}
$$

Example 11 (Ex. 1 with Direct Termination). Consider the variant $\mathcal{P}_{direct}$ of $\mathcal{P}_{race}$ on the right, where in each iteration, the hare either does nothing with probability $\frac{9}{10}$ or one directly reaches a configuration where the hare is ahead of the tortoise. By Thm. 10 the program is PAST and its expected runtime is at most $\frac{1}{\frac{1}{10}} = 10$, i.e., independent of the initial state it takes at most 10 loop iterations on average. In Sect. 5 it will turn out that 10 is indeed the exact expected runtime, cf. Ex. 32.

4 Expected Runtimes of Constant Probability Programs

Now we present a very simple decision procedure for termination of CP programs (Sect. 4.2) and show how to infer their asymptotic expected runtimes (Sect. 4.3). This will be needed for the computation of exact expected runtimes in Sect. 5.

4.1 Reduction to Random Walk Programs

As a first step, we show that we can restrict ourselves to *random walk programs*, i.e., programs with a single program variable x and the loop condition $x > 0$.

$$
\begin{array}{ll}
\texttt{while } (x > 0) \ \{ & \\
\quad x = x + m & [p_m]; \\
\quad \vdots & \\
\quad x = x + 1 & [p_1]; \\
\quad x = x & [p_0]; \\
\quad x = x - 1 & [p_{-1}]; \\
\quad \vdots & \\
\quad x = x - k & [p_{-k}]; \\
\quad x = d & [p']; \\
\} &
\end{array}
$$

Definition 12 (Random Walk Program). *A CP program $\mathcal{P}$ is called a* random walk program *if there exist $m, k \in \mathbb{N}$ and $d \in \mathbb{Z}$ with $d \leq 0$ such that $\mathcal{P}$ has the form on the right. Here, we require that $m > 0$ implies $p_m > 0$ and that $k > 0$ implies $p_{-k} > 0$.*

Def. 13 shows how to transform any CP program as in Def. 2 into a random walk program. The idea is to replace the tuple $\boldsymbol{x}$ by a single variable x that stands for $\boldsymbol{a} \bullet \boldsymbol{x} - b$. Thus, the loop condition $\boldsymbol{a} \bullet \boldsymbol{x} > b$ now becomes $x > 0$. Moreover, a change from $\boldsymbol{x}$ to $\boldsymbol{x} + \boldsymbol{c}_j$ now becomes a change from x to $x + \boldsymbol{a} \bullet \boldsymbol{c}_j$.

```
while (a • x > b) {
  x = x + c_1    [p_{c_1}];
    ⋮
  x = x + c_n    [p_{c_n}];
  x = d          [p'];
}
```

Definition 13 (Transforming CP Programs to Random Walk Programs). *Let $\mathcal{P}$ be the CP program on the left with $\boldsymbol{x} = (x_1, \ldots, x_r)$ and $\boldsymbol{a} \bullet \boldsymbol{d} \leq b$. Let $rdw_{\mathcal{P}}$ denote the affine map $rdw_{\mathcal{P}} : \mathbb{Z}^r \to \mathbb{Z}$ with $rdw_{\mathcal{P}}(\boldsymbol{z}) = \boldsymbol{a} \bullet \boldsymbol{z} - b$ for all $\boldsymbol{z} \in \mathbb{Z}^r$. Thus, $rdw_{\mathcal{P}}(\boldsymbol{d}) \leq 0$. Let $k_{\mathcal{P}}, m_{\mathcal{P}} \in \mathbb{N}$ be minimal such that $-k_{\mathcal{P}} \leq \boldsymbol{a} \bullet \boldsymbol{c}_j \leq m_{\mathcal{P}}$ holds for all $1 \leq j \leq n$. For all $-k_{\mathcal{P}} \leq j \leq m_{\mathcal{P}}$, we define $p_j^{rdw} = \sum_{1 \leq u \leq n,\ \boldsymbol{a} \bullet \boldsymbol{c}_u = j} p_{\boldsymbol{c}_u}$. This results in the random walk program $\mathcal{P}^{rdw}$ on the right.*

```
while (x > 0) {
  x = x + m_P       [p^{rdw}_{m_P}];
    ⋮
  x = x - k_P       [p^{rdw}_{-k_P}];
  x = rdw_P(d)      [p'];
}
```

Example 14 (Transforming $\mathcal{P}_{race}$). For the program $\mathcal{P}_{race}$ of Ex. 1, the mapping $rdw_{\mathcal{P}_{race}} : \mathbb{Z}^2 \to \mathbb{Z}$ is $rdw_{\mathcal{P}_{race}}(t, h) = (1, -1) \bullet (t, h) + 1 = t - h + 1$. Hence we obtain the random walk program $\mathcal{P}_{race}^{rdw}$ on the right, where $x = rdw_{\mathcal{P}_{race}}(t, h)$ represents the distance between the tortoise and the hare.

```
while (x > 0) {
  x = x + 1    [6/11];
  x = x        [1/22];
  x = x - 1    [1/22];
  x = x - 2    [1/22];
    ⋮
  x = x - 9    [1/22];
}
```

Approaches based on supermartingales (e.g., [1,5,11,13,14,18]) use mappings similar to $rdw_{\mathcal{P}}$ in order to infer a real-valued term which over-approximates the expected runtime. However, in the following (non-trivial) theorem we show that our transformation is not only an over- or under-approximation, but the termination behavior and the expected runtime of $\mathcal{P}$ and $\mathcal{P}^{rdw}$ are *identical*.

Theorem 15 (Transformation Preserves Termination & Expected Runtime). *Let $\mathcal{P}$ be a CP program as in Def. 2. Then the termination times $T^{\mathcal{P}}$ and $T^{\mathcal{P}^{rdw}}$ are identically distributed w.r.t. $rdw_{\mathcal{P}}$, i.e., for all $\boldsymbol{x}_0 \in \mathbb{Z}^r$ with $x_0 = rdw_{\mathcal{P}}(\boldsymbol{x}_0)$ and all $j \in \overline{\mathbb{N}}$ we have $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} = j) = \mathbb{P}^{\mathcal{P}^{rdw}}_{x_0}(T^{\mathcal{P}^{rdw}} = j)$. So in particular, $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} < \infty) = \mathbb{P}^{\mathcal{P}^{rdw}}_{x_0}(T^{\mathcal{P}^{rdw}} < \infty)$ and $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{E}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}}) = \mathbb{E}^{\mathcal{P}^{rdw}}_{x_0}(T^{\mathcal{P}^{rdw}}) = rt^{\mathcal{P}^{rdw}}_{x_0}$. Thus, the expected runtimes of $\mathcal{P}$ on the input $\boldsymbol{x}_0$ and of $\mathcal{P}^{rdw}$ on x_0 coincide.*

The following definition identifies pathological programs that can be disregarded.

Definition 16 (Trivial Program). *Let $\mathcal{P}$ be a CP program as in Def. 2. We call $\mathcal{P}$* trivial *if $\boldsymbol{a} = \boldsymbol{0} = (0, 0, \ldots, 0)$ or if $\mathcal{P}^{rdw}$ is the program on the right.*

```
while (x > 0) {
  x = x    [1];
}
```

Note that a random walk program $\mathcal{P}$ is trivial iff it has the form $\texttt{while}(x > 0)\{x = x\ \ [1];\}$, since $\mathcal{P} = \mathcal{P}^{rdw}$ holds for random walk programs $\mathcal{P}$. From now on, we will exclude trivial programs $\mathcal{P}$ as their termination behavior is obvious: for inputs $\boldsymbol{x}_0$ that satisfy the loop condition $\boldsymbol{a} \bullet \boldsymbol{x}_0 > b$, the program never terminates (i.e., $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \infty$) and for inputs $\boldsymbol{x}_0$ with $\boldsymbol{a} \bullet \boldsymbol{x}_0 \leq b$ we have $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = 0$. Note that if $\boldsymbol{a} = \boldsymbol{0}$, then the termination behavior just depends on b: if $b < 0$, then $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \infty$ for all $\boldsymbol{x}_0$ and if $b \geq 0$, then $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = 0$ for all $\boldsymbol{x}_0$.

4.2 Deciding Termination

We now present a simple decision procedure for (P)AST of random walk programs $\mathcal{P}$. By the results of Sect. 4.1, this also yields a decision procedure for arbitrary CP programs. If $p' > 0$, then Thm. 10 already shows that $\mathcal{P}$ is PAST and its expected runtime is bounded by the constant $\frac{1}{p'}$. Thus, in the rest of Sect. 4 we regard *random walk programs without direct termination*, i.e., $p' = 0$.

Def. 17 introduces the *drift* of a random walk program, i.e., the expected value of the change of the program variable in one loop iteration, cf. [5].

Definition 17 (Drift). *Let $\mathcal{P}$ be a random walk program $\mathcal{P}$ as in Def. 12. Then its* drift *is $\mu_{\mathcal{P}} = \sum_{-k \leq j \leq m} j \cdot p_j$.*

Thm. 18 shows that to decide (P)AST, one just has to compute the drift.

Theorem 18 (Decision Procedure for (P)AST of Random Walk Programs). *Let $\mathcal{P}$ be a non-trivial random walk program without direct termination.*

- *If $\mu_{\mathcal{P}} > 0$, then the program is* not *AST.*
- *If $\mu_{\mathcal{P}} = 0$, then the program is AST but* not *PAST.*
- *If $\mu_{\mathcal{P}} < 0$, then the program is PAST.*

Example 19 ($\mathcal{P}_{race}$ is PAST). The drift of $\mathcal{P}^{rdw}_{race}$ in Ex. 14 is $\mu_{\mathcal{P}^{rdw}_{race}} = 1 \cdot \frac{6}{11} + \frac{1}{22} \cdot \sum_{-9 \leq j \leq 0} j = -\frac{3}{2} < 0$. So on average the distance x between the tortoise and the hare decreases in each loop iteration. Hence by Thm. 18, $\mathcal{P}^{rdw}_{race}$ is PAST and the following Cor. 20 implies that $\mathcal{P}_{race}$ is PAST as well.

Corollary 20 (Decision Procedure for (P)AST of CP programs). *For a non-trivial CP program $\mathcal{P}$, $\mathcal{P}$ is (P)AST iff $\mathcal{P}^{rdw}$ is (P)AST. Hence, Thm. 15 and 18 yield a decision procedure for AST and PAST of CP programs.*

In the appendix, we show that Thm. 18 follows from classical results on random walks [33]. Alternatively, Thm. 18 could also be proved by combining several recent results on probabilistic programs: The approach of [28] could be used to show that $\mu_{\mathcal{P}} = 0$ implies AST. Moreover, one could prove that $\mu_{\mathcal{P}} < 0$ implies PAST by showing that x is a *ranking supermartingale* of the program [5,11,14,18]. That the program is not PAST if $\mu_{\mathcal{P}} \geq 0$ and not AST if $\mu_{\mathcal{P}} > 0$ could be proved by showing that $-x$ is a $\mu_{\mathcal{P}}$*-repulsing supermartingale* [13].

While the proof of Thm. 18 is based on known results, the formulation of Thm. 18 shows that there is an extremely *simple* decision procedure for (P)AST of CP programs, i.e., checking the sign of the drift is much simpler than applying existing (general) techniques for termination analysis of probabilistic programs.

4.3 Computing Asymptotic Expected Runtimes

It turns out that for random walk programs (and thus by Thm. 15, also for CP programs), one can not only decide termination, but one can also infer tight bounds on the expected runtime. Thm. 21 shows that the computation of the bounds is again very *simple*.

Theorem 21 (Bounds on the Expected Runtime of CP Programs). *Let $\mathcal{P}$ be a non-trivial CP program as in Def. 2 without direct termination which is PAST (i.e., $\mu_{\mathcal{P}^{rdw}} < 0$). Moreover, let $k_{\mathcal{P}}$ be obtained according to the transformation from Def. 13. If $rdw_{\mathcal{P}}(\boldsymbol{x}_0) \leq 0$, then $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = 0$. If $rdw_{\mathcal{P}}(\boldsymbol{x}_0) > 0$, then $\mathcal{P}$'s expected runtime is asymptotically linear and we have*

$$-\tfrac{1}{\mu_{\mathcal{P}^{rdw}}} \cdot rdw_{\mathcal{P}}(\boldsymbol{x}_0) \quad \leq \quad rt^{\mathcal{P}}_{\boldsymbol{x}_0} \quad \leq \quad -\tfrac{1}{\mu_{\mathcal{P}^{rdw}}} \cdot rdw_{\mathcal{P}}(\boldsymbol{x}_0) + \tfrac{1-k_{\mathcal{P}}}{\mu_{\mathcal{P}^{rdw}}}.$$

Example 22 (Bounds on the Runtime of $\mathcal{P}_{race}$). In Ex. 19 we saw that the program $\mathcal{P}^{rdw}_{race}$ from Ex. 14 is PAST as it has the drift $\mu_{\mathcal{P}^{rdw}_{race}} = -\frac{3}{2} < 0$. Note that here $k = 9$. Hence by Thm. 21 we get that whenever $rdw_{\mathcal{P}_{race}}(t,h) = t-h+1$ is positive, the expected runtime $rt^{\mathcal{P}_{race}}_{(t,h)}$ is between $-\frac{1}{\mu_{\mathcal{P}^{rdw}_{race}}} \cdot rdw_{\mathcal{P}_{race}}(t,h) = \frac{2}{3} \cdot (t-h+1)$ and $-\frac{1}{\mu_{\mathcal{P}^{rdw}_{race}}} \cdot rdw_{\mathcal{P}_{race}}(t,h) + \frac{1-k}{\mu_{\mathcal{P}^{rdw}_{race}}} = \frac{2}{3} \cdot (t-h+1) + \frac{16}{3}$. The same upper bound $\frac{2}{3} \cdot (t-h+1) + \frac{16}{3}$ was inferred in [30] by an incomplete technique based on several inference rules and linear programming solvers. In contrast, Thm. 21 allows us to read off such bounds directly from the program.

Our proof of Thm. 21 in the appendix again uses the connection to random walks and shows that the classical Lemma of Wald [21, Lemma 10.2(9)] directly yields both the upper and the lower bound for the expected runtime. Alternatively, the upper bound in Thm. 21 could also be proved by considering that $rdw_{\mathcal{P}}(\boldsymbol{x}_0) + (1-k_{\mathcal{P}})$ is a *ranking supermartingale* [1,5,11,14,18] whose expected decrease in each loop iteration is $\mu_{\mathcal{P}}$. The lower bound could also be inferred by considering the *difference-bounded submartingale* $-rdw_{\mathcal{P}}(\boldsymbol{x}_0)$ [8,20].

5 Computing Exact Expected Runtimes

While Thm. 10 and 21 state how to deduce the *asymptotic* expected runtime, we now show that based on these results one can compute the runtime of CP programs *exactly*. In general, whenever it is possible, then inferring the exact runtimes of programs is preferable to asymptotic runtimes which ignore the "coefficients" of the runtime.

Again, we first consider *random walk* programs and generalize our technique to CP programs using Thm. 15 afterwards. Throughout Sect. 5, for any random walk program $\mathcal{P}$ as in Def. 12, we require that $\mathcal{P}$ is PAST, i.e., that $p' > 0$ (cf. Thm. 10) or that the drift $\mu_{\mathcal{P}}$ is negative if $p' = 0$ (cf. Thm. 18). Note that whenever $k = 0$ and $\mathcal{P}$ is PAST, then $p' > 0$.[3]

To compute $\mathcal{P}$'s expected runtime exactly, we use its characterization as the least fixpoint of the expected runtime transformer $\mathcal{L}^{\mathcal{P}}$ (cf. Thm. 9), i.e., $rt^{\mathcal{P}}_x$ is

[3] If $p' = 0$ and $k = 0$ then $\mu_{\mathcal{P}} \geq 0$.

the smallest function $f : \mathbb{Z} \to \overline{\mathbb{R}_{\geq 0}}$ satisfying the constraint

$$f(x) \;=\; \sum\nolimits_{-k\leq j\leq m} p_j \cdot f(x+j) + p' \cdot f(d) + 1 \quad \text{for all } x > 0, \tag{3}$$

cf. (1). Since $\mathcal{P}$ is PAST, f never returns ∞, i.e., $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$. Note that the smallest function $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ that satisfies (3) also satisfies

$$f(x) \;=\; 0 \quad \text{for all } x \leq 0. \tag{4}$$

Therefore, as $d \leq 0$, the constraint (3) can be simplified to

$$f(x) \;=\; \sum\nolimits_{-k\leq j\leq m} p_j \cdot f(x+j) + 1 \quad \text{for all } x > 0. \tag{5}$$

In Sect. 5.1 we recapitulate how to compute all solutions of such inhomogeneous recurrence equations (cf., e.g., [15, Ch. 2]). However, to compute $rt_x^{\mathcal{P}}$, the challenge is to find the *smallest* solution $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ of the recurrence equation (5). Therefore, in Sect. 5.2 we will exploit the knowledge gained in Thm. 10 and 21 to show that there is only a *single* function f that satisfies both (4) and (5) *and* is bounded by a constant (if $p' > 0$, cf. Thm. 10) resp. by a linear function (if $p' = 0$, cf. Thm. 21). This observation then allows us to compute $rt_x^{\mathcal{P}}$ exactly. So the crucial prerequisites for this result are Thm. 9 (which characterizes the expected runtime as the smallest solution of the recurrence equation (5)), Thm. 18 (which allows the restriction to negative drift if $p' = 0$), and in particular Thm. 10 and 21 (since Sect. 5.2 will show that the results of Thm. 10 and 21 on the asymptotic runtime can be translated into suitable conditions on the solutions of (5)).

5.1 Finding All Solutions of the Recurrence Equation

Example 23 (Modification of $\mathcal{P}_{race}^{rdw}$). To illustrate our approach, we use a modified version of $\mathcal{P}_{race}^{rdw}$ from Ex. 14 to ease readability. In Sect. 6, we will consider the original program $\mathcal{P}_{race}^{rdw}$ resp. $\mathcal{P}_{race}$ from Ex. 14 resp. Ex. 1 again and show its exact expected runtime inferred by the implementation of our approach. In the modified program $\mathcal{P}_{race}^{mod}$ on the right, the distance between the tortoise and the hare still increases with probability $\frac{6}{11}$, but the probability of decreasing by more than two is distributed to the cases where it stays the same and where it decreases by two. We have $p' = 0$ and the drift is $\mu_{\mathcal{P}_{race}^{mod}} = 1 \cdot \frac{6}{11} + 0 \cdot \frac{1}{11} - 1 \cdot \frac{1}{22} - 2 \cdot \frac{7}{22} = -\frac{3}{22} < 0$. So by Thm. 18, $\mathcal{P}_{race}^{mod}$ is PAST. By Thm. 9, $rt_x^{\mathcal{P}_{race}^{mod}}$ is the smallest function $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ satisfying

```
while (x > 0) {
  x = x + 1   [6/11];
  x = x       [1/11];
  x = x − 1   [1/22];
  x = x − 2   [7/22];
}
```

$$f(x) \;=\; \tfrac{6}{11} \cdot f(x+1) + \tfrac{1}{11} \cdot f(x) + \tfrac{1}{22} \cdot f(x-1) + \tfrac{7}{22} \cdot f(x-2) + 1 \;\; \textit{for all } x > 0. \tag{6}$$

Instead of searching for the *smallest* $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ satisfying (5), we first calculate the set of *all* functions $f : \mathbb{Z} \to \mathbb{C}$ that satisfy (5), i.e., we also consider functions returning negative or complex numbers. Clearly, (5) is equivalent to

$$\begin{aligned} 0 \;=\; & p_m \cdot f(x+m) + \ldots + p_1 \cdot f(x+1) + (p_0 - 1) \cdot f(x) + \\ & p_{-1} \cdot f(x-1) + \ldots + p_{-k} \cdot f(x-k) + 1 \qquad\qquad \text{for all } x > 0. \end{aligned} \tag{7}$$

The set of solutions on $\mathbb{Z} \to \mathbb{C}$ of this linear, inhomogeneous recurrence equation is an affine space which can be written as an arbitrary particular solution of the inhomogeneous equation plus any linear combination of $k+m$ linearly independent solutions of the corresponding homogeneous recurrence equation.

We start with computing a solution to the inhomogeneous equation (7). To this end, we use the bounds for $rt_x^{\mathcal{P}}$ from Thm. 10 and 21 (where we take the upper bound $\frac{1}{p'}$ if $p' > 0$ and the lower bound $-\frac{1}{\mu_{\mathcal{P}}} \cdot x$ if $p' = 0$). So we define

$$C_{const} = \tfrac{1}{p'}, \text{ if } p' > 0 \qquad \text{and} \qquad C_{lin} = -\tfrac{1}{\mu_{\mathcal{P}}}, \text{ if } p' = 0.$$

One easily shows that if $p' > 0$, then $f(x) = C_{const}$ is a solution of the inhomogeneous recurrence equation (7) and if $p' = 0$, then $f(x) = C_{lin} \cdot x$ solves (7).

Example 24 (Ex. 23 cont.). In the program $\mathcal{P}_{race}^{mod}$ of Ex. 23, we have $p' = 0$ and $\mu_{\mathcal{P}_{race}^{mod}} = -\frac{3}{22}$. Hence $C_{lin} = \frac{22}{3}$ and $C_{lin} \cdot x$ is a solution of (6).

After having determined one particular solution of the inhomogeneous recurrence equation (7), now we compute the solutions of the *homogeneous* recurrence equation which results from (7) by replacing the add-on "+ 1" with 0. To this end, we consider the corresponding *characteristic polynomial* $\chi_{\mathcal{P}}$:[4]

$$\chi_{\mathcal{P}}(\lambda) = p_m \cdot \lambda^{k+m} + \ldots + p_1 \cdot \lambda^{k+1} + (p_0 - 1) \cdot \lambda^k + p_{-1} \cdot \lambda^{k-1} + \ldots + p_{-k} \quad (8)$$

Let $\lambda_1, \ldots, \lambda_c$ denote the pairwise different (possibly complex) roots of the characteristic polynomial $\chi_{\mathcal{P}}$. For all $1 \leq j \leq c$, let $v_j \in \mathbb{N} \setminus \{0\}$ be the multiplicity of the root λ_j. Thus, we have $v_1 + \ldots + v_c = k + m$.

Then we obtain the following $k + m$ linearly independent solutions of the homogeneous recurrence equation resulting from (7):

$$\lambda_j^x \cdot x^u \qquad \text{for all } 1 \leq j \leq c \text{ and all } 0 \leq u \leq v_j - 1$$

So $f : \mathbb{Z} \to \mathbb{C}$ is a solution of (5) (resp. (7)) iff there exist coefficients $a_{j,u} \in \mathbb{C}$ with

$$f(x) \;=\; C(x) + \sum\nolimits_{1 \leq j \leq c} \; \sum\nolimits_{0 \leq u \leq v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u \quad \text{for all } x > -k, \qquad (9)$$

where $C(x) = C_{const} = \frac{1}{p'}$ if $p' > 0$ and $C(x) = C_{lin} \cdot x = -\frac{1}{\mu_{\mathcal{P}}} \cdot x$ if $p' = 0$. The reason for requiring (9) for all $x > -k$ is that $-k + 1$ is the smallest argument where f's value is taken into account in (5).

Example 25 (Ex. 24 cont.). The characteristic polynomial for the program $\mathcal{P}_{race}^{mod}$ of Ex. 23 has the degree $k + m = 2 + 1 = 3$ and is given by

$$\chi_{\mathcal{P}_{race}^{mod}}(\lambda) \;=\; \tfrac{6}{11} \cdot \lambda^3 - \tfrac{10}{11} \cdot \lambda^2 + \tfrac{1}{22} \cdot \lambda + \tfrac{7}{22}.$$

[4] If $m = 0$ then $\chi_{\mathcal{P}}(\lambda) = (p_0 - 1) \cdot \lambda^k + p_{-1} \cdot \lambda^{k-1} + \ldots + p_{-k}$, and if $k = 0$ then $\chi_{\mathcal{P}}(\lambda) = p_m \cdot \lambda^m + \ldots + p_1 \cdot \lambda + (p_0 - 1)$. Note that $p_0 \neq 1$ since $\mathcal{P}$ is PAST and in Def. 12 we required that $m > 0$ implies $p_m > 0$ and $k > 0$ implies $p_{-k} > 0$. Hence, the characteristic polynomial has exactly the degree $k + m$, even if $m = 0$ or $k = 0$.

Its roots are $\lambda_1 = 1$, $\lambda_2 = -\frac{1}{2}$, *and* $\lambda_3 = \frac{7}{6}$. *So here, all roots are real numbers and they all have the multiplicity* 1. *Hence, three linearly independent solutions of the homogeneous part of* (6) *are the functions* $1^x = 1$, $(-\frac{1}{2})^x$, *and* $(\frac{7}{6})^x$. *Therefore, a function* $f : \mathbb{Z} \to \mathbb{C}$ *satisfies* (6) *iff there are* $a_1, a_2, a_3 \in \mathbb{C}$ *such that*

$$\begin{array}{rcll} f(x) & = & C_{lin} \cdot x + a_1 \cdot 1^x + a_2 \cdot (-\frac{1}{2})^x + a_3 \cdot (\frac{7}{6})^x & \\ & = & \frac{22}{3} \cdot x + a_1 + a_2 \cdot (-\frac{1}{2})^x + a_3 \cdot (\frac{7}{6})^x & \text{for } x > -2. \end{array} \tag{10}$$

5.2 Finding the Smallest Solution of the Recurrence Equation

In Sect. 5.1, we recapitulated the standard approach for solving inhomogeneous recurrence equations which shows that any function $f : \mathbb{Z} \to \mathbb{C}$ that satisfies the constraint (5) is of the form (9). Now we will present a novel technique to compute $rt_x^{\mathcal{P}}$, i.e., the *smallest non-negative* solution $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ of (5). By Thm. 10 and 21, this function f is bounded by a constant (if $p' > 0$) resp. linear (if $p' = 0$). So, when representing f in the form (9), we must have $a_{j,u} = 0$ whenever $|\lambda_j| > 1$. The following lemma shows how many roots with absolute value less or equal to 1 there are (i.e., these are the only roots that we have to consider). It is proved using Rouché's Theorem which allows us to infer the number of roots whose absolute value is below a certain bound. Note that 1 is a root of the characteristic polynomial iff $p' = 0$, since $\sum_{-k \leq j \leq m} p_j = 1 - p'$.

Lemma 26 (Number of Roots With Absolute Value ≤ 1). *Let* $\mathcal{P}$ *be a random walk program as in Def. 12 that is PAST. Then the characteristic polynomial* $\chi_{\mathcal{P}}$ *has* k *roots* $\lambda \in \mathbb{C}$ *(counted with multiplicity) with* $|\lambda| \leq 1$.

Example 27 (Ex. 25 cont.). *In* $\mathcal{P}_{race}^{mod}$ *of Ex. 23 we have* $k = 2$. *So by Lemma 26,* $\chi_{\mathcal{P}}$ *has exactly two roots with absolute value* ≤ 1. *Indeed, the roots of* $\chi_{\mathcal{P}}$ *are* $\lambda_1 = 1$, $\lambda_2 = -\frac{1}{2}$, *and* $\lambda_3 = \frac{7}{6}$, *cf. Ex. 25. So* $|\lambda_3| > 1$, *but* $|\lambda_1| \leq 1$ *and* $|\lambda_2| \leq 1$.

Based on Lemma 26, the following lemma shows that when imposing the restriction that $a_{j,u} = 0$ whenever $|\lambda_j| > 1$, then there is only a *single* function of the form (9) that also satisfies the constraint (4). Hence, this must be the function that we are searching for, because the desired smallest solution $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ of (5) also satisfies (4).

Lemma 28 (Unique Solution of (4) and (5) when Disregarding Roots With Absolute Value > 1). *Let* $\mathcal{P}$ *be a random walk program as in Def. 12 that is PAST. Then there is exactly one function* $f : \mathbb{Z} \to \mathbb{C}$ *which satisfies both* (4) *and* (5) *(thus, it has the form* (9)*) and has* $a_{j,u} = 0$ *whenever* $|\lambda_j| > 1$.

The main theorem of Sect. 5 now shows how to compute the expected runtime exactly. By Thm. 10 and 21 on the bounds for the expected runtime and by Lemma 28, we no longer have to search for the *smallest* function that satisfies (4) and (5), but we just search for *any* solution of (4) and (5) which has $a_{j,u} = 0$ whenever $|\lambda_j| > 1$ (because there is just a single such solution). So one only has to determine the values of the remaining k coefficients $a_{j,u}$ for $|\lambda_j| \leq 1$, which can be done by exploiting that $f(x)$ has to satisfy both (4) for all $x \leq 0$ and it has to be of the form (9) for all $x > -k$. In other words, the function in (9) must be 0 for $-k + 1 \leq x \leq 0$.

Theorem 29 (Exact Expected Runtime for Random Walk Programs). *Let $\mathcal{P}$ be a random walk program as in Def. 12 that is PAST and let $\lambda_1, \ldots, \lambda_c$ be the roots of its characteristic polynomial with multiplicities $v_1, \ldots, v_c$. Moreover, let $C(x) = C_{const} = \frac{1}{p'}$ if $p' > 0$ and $C(x) = C_{lin} \cdot x = -\frac{1}{\mu_{\mathcal{P}}} \cdot x$ if $p' = 0$. Then the expected runtime of $\mathcal{P}$ is $rt_x^{\mathcal{P}} = 0$ for $x \leq 0$ and*

$$rt_x^{\mathcal{P}} \;=\; C(x) \;+\; \sum\nolimits_{1 \leq j \leq c,\; |\lambda_j| \leq 1} \;\sum\nolimits_{0 \leq u \leq v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u \quad \text{for } x > 0,$$

where the coefficients $a_{j,u}$ are the unique solution of the k linear equations:

$$0 \;=\; C(x) + \sum\nolimits_{1 \leq j \leq c,\; |\lambda_j| \leq 1} \sum\nolimits_{0 \leq u \leq v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u \quad \text{for } -k+1 \leq x \leq 0 \tag{11}$$

So in the special case where $k = 0$, we have $rt_x^{\mathcal{P}} = C(x) = C_{const} = \frac{1}{p'}$ for $x > 0$.

Thus for $x > 0$, the expected runtime $rt_x^{\mathcal{P}}$ can be computed by summing up the bound $C(x)$ and an add-on $\sum_{1 \leq j \leq c,\; |\lambda_j| \leq 1} \sum_{0 \leq u \leq v_j - 1} \ldots$ Since $C(x)$ is an upper bound for $rt_x^{\mathcal{P}}$ if $p' > 0$ and a lower bound for $rt_x^{\mathcal{P}}$ if $p' = 0$, this add-on is non-positive if $p' > 0$ and non-negative if $p' = 0$.

Example 30 (Ex. 27 cont.). By Thm. 29, the expected runtime of the program $\mathcal{P}_{race}^{mod}$ from Ex. 23 is $rt_x^{\mathcal{P}_{race}^{mod}} = 0$ for $x \leq 0$ and

$$rt_x^{\mathcal{P}_{race}^{mod}} \;=\; \tfrac{22}{3} \cdot x + a_1 + a_2 \cdot (-\tfrac{1}{2})^x \quad \text{for } x > 0, \quad \text{cf. (10).}$$

The coefficients a_1 and a_2 are the unique solution of the $k = 2$ linear equations

$$0 = \tfrac{22}{3} \cdot 0 + a_1 + a_2 \cdot (-\tfrac{1}{2})^0 = a_1 + a_2$$
$$0 = \tfrac{22}{3} \cdot (-1) + a_1 + a_2 \cdot (-\tfrac{1}{2})^{-1} = -\tfrac{22}{3} + a_1 - 2 \cdot a_2$$

So $a_1 = \frac{22}{9}$, $a_2 = -\frac{22}{9}$, and hence $rt_x^{\mathcal{P}_{race}^{mod}} = \frac{22}{3} \cdot x + \frac{22}{9} - \frac{22}{9} \cdot (-\frac{1}{2})^x$ for $x > 0$.

By Thm. 15, we can lift Thm. 29 to arbitrary CP programs $\mathcal{P}$ immediately.

Corollary 31 (Exact Expected Runtime for CP Programs). *For any CP program, its expected runtime can be computed exactly.*

Note that irrespective of the degree of the characteristic polynomial, its roots can always be approximated numerically with any chosen precision. Thus, "exact computation" of the expected runtime in the corollary above means that a closed form for $rt_{\boldsymbol{x}}^{\mathcal{P}}$ can also be computed with any desired precision.

Example 32 (Exact Expected Runtime of $\mathcal{P}_{direct}$). Reconsider the program $\mathcal{P}_{direct}$ of Ex. 11 with the probability $p' = \frac{1}{10}$ for direct termination. $\mathcal{P}_{direct}$ is PAST and its expected runtime is at most $\frac{1}{p'} = 10$, cf. Ex. 11. The random walk program $\mathcal{P}_{direct}^{rdw}$ on the right is obtained by the transforma-

```
while (x > 0) {
  x = x + 1    [9/10];
  x = 0        [1/10];
}
```

tion of Def. 13. As $k = 0$, by Thm. 29 we obtain $rt_x^{\mathcal{P}^{rdw}_{direct}} = \frac{1}{p'} = 10$ for $x > 0$. By Thm. 15, this implies $rt_{(t,h)}^{\mathcal{P}_{direct}} = rt_{rdw_{\mathcal{P}_{direct}}(t,h)}^{\mathcal{P}^{rdw}_{direct}} = 10$ if $rdw_{\mathcal{P}_{direct}}(t,h) = t - h + 1 > 0$, i.e., 10 is indeed the exact *expected runtime of $\mathcal{P}_{direct}$.*

Note that Thm. 29 and Cor. 31 imply that for any $\boldsymbol{x}_0 \in \mathbb{Z}^r$, the expected runtime $rt_{\boldsymbol{x}_0}^{\mathcal{P}}$ of a CP program $\mathcal{P}$ that is PAST and has only *rational* probabilities $p_{\boldsymbol{c}_1}, \ldots, p_{\boldsymbol{c}_n}, p' \in \mathbb{Q}$ is always an algebraic number. Thus, one could also compute a closed form for the exact expected runtime $rt_{\boldsymbol{x}}^{\mathcal{P}}$ using a representation with algebraic numbers instead of numerical approximations.

Nevertheless, Thm. 29 may yield a representation of $rt_x^{\mathcal{P}}$ which contains complex numbers $a_{j,u}$ and λ_j, although $rt_x^{\mathcal{P}}$ is always real. However, one can easily obtain a more intuitive representation of $rt_x^{\mathcal{P}}$ without complex numbers:

Since the characteristic polynomial $\chi_{\mathcal{P}}$ only has real coefficients, whenever $\chi_{\mathcal{P}}$ has a complex root λ of multiplicity v, its conjugate $\overline{\lambda}$ is also a root of $\chi_{\mathcal{P}}$ with the same multiplicity v. So the pairwise different roots $\lambda_1, \ldots, \lambda_c$ can be distinguished into pairwise different real roots $\lambda_1, \ldots, \lambda_s$, and into pairwise different non-real complex roots $\lambda_{s+1}, \overline{\lambda_{s+1}}, \ldots, \lambda_{s+t}, \overline{\lambda_{s+t}}$, where $c = s + 2 \cdot t$.

For any coefficients $a_{j,u}, a'_{j,u} \in \mathbb{C}$ with $j \in \{s{+}1, \ldots, s{+}t\}$ and $u \in \{0, \ldots, v_j - 1\}$ let $b_{j,u} = 2 \cdot \mathrm{Re}(a_{j,u}) \in \mathbb{R}$ and $b'_{j,u} = -2 \cdot \mathrm{Im}(a_{j,u}) \in \mathbb{R}$. Then $a_{j,u} \cdot \lambda_j^x + a'_{j,u} \cdot \overline{\lambda_j}^x = b_{j,u} \cdot \mathrm{Re}(\lambda_j^x) + b'_{j,u} \cdot \mathrm{Im}(\lambda_j^x)$. Hence, by Thm. 29 we get the following representation of the expected runtime which only uses *real* numbers:

$$rt_x^{\mathcal{P}} = \begin{cases} C(x) + \sum\limits_{1 \le j \le s,\ |\lambda_j| \le 1} \ \sum\limits_{0 \le u \le v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u \\ \quad + \sum\limits_{s+1 \le j \le s+t,\ |\lambda_j| \le 1} \ \sum\limits_{0 \le u \le v_j - 1} \left(b_{j,u} \cdot \mathrm{Re}(\lambda_j^x) + b'_{j,u} \cdot \mathrm{Im}(\lambda_j^x)\right) \cdot x^u, & \text{for } x > 0 \\ 0, & \text{for } x \le 0 \end{cases} \tag{12}$$

To compute $\mathrm{Re}(\lambda_j^x)$ and $\mathrm{Im}(\lambda_j^x)$, take the polar representation of the non-real roots $\lambda_j = w_j \cdot e^{\theta_j \cdot i}$. Then $\mathrm{Re}(\lambda_j^x) = w_j^x \cdot \cos(\theta_j \cdot x)$ and $\mathrm{Im}(\lambda_j^x) = w_j^x \cdot \sin(\theta_j \cdot x)$.

Therefore, we obtain the following algorithm to deduce the exact expected runtime automatically.

Algorithm 33 (Computing the Exact Expected Runtime). *To infer the runtime of a CP program $\mathcal{P}$ as in Def. 12 that is PAST, we proceed as follows:*

1. *Transform $\mathcal{P}$ into $\mathcal{P}^{rdw}$ by the transformation of Def. 13. Thus, $\mathcal{P}^{rdw}$ is a random walk program as in Def. 12.*
2. *Compute the solution $C(x) = C_{const} = \frac{1}{p'}$ resp. $C(x) = C_{lin} \cdot x = -\frac{1}{\mu_{\mathcal{P}^{rdw}}} \cdot x$ of the inhomogeneous recurrence equation (7).*
3. *Compute the $k + m$ (possibly complex) roots of the characteristic polynomial $\chi_{\mathcal{P}^{rdw}}$ (cf. (8)) and keep the k roots λ with $|\lambda| \le 1$.*
4. *Determine the coefficients $a_{j,u}$ by solving the k linear equations in (11).*
5. *Return the solution (12) where $b_{j,u} = 2 \cdot \mathrm{Re}(a_{j,u})$, $b'_{j,u} = -2 \cdot \mathrm{Im}(a_{j,u})$, and for $\lambda_j = w_j \cdot e^{\theta_j \cdot i}$ we have $\mathrm{Re}(\lambda_j^x) = w_j^x \cdot \cos(\theta_j \cdot x)$ and $\mathrm{Im}(\lambda_j^x) = w_j^x \cdot \sin(\theta_j \cdot x)$. Moreover, x must be replaced by $rdw_{\mathcal{P}}(\boldsymbol{x})$.*

6 Conclusion, Implementation, and Related Work

We presented decision procedures for termination and complexity of classes of probabilistic programs. They are based on the connection between the expected runtime of a program and the smallest solution of its corresponding recurrence equation, cf. Sect. 2. For our notion of probabilistic programs, if the probability for leaving the loop directly is at least p' for some $p' > 0$, then the program is always PAST and its expected runtime is asymptotically constant, cf. Sect. 3. In Sect. 4 we showed that a very simple decision procedure for AST and PAST of CP programs can be obtained by classical results from random walk theory and that the expected runtime is asymptotically linear if the program is PAST. Based on these results, in Sect. 5 we presented our algorithm to automatically infer a closed form for the *exact* expected runtime of CP programs (i.e., with arbitrarily high precision). All proofs and a collection of examples to demonstrate our algorithm can be found in the appendix.

Implementation. We implemented Alg. 33 in our tool KoAT [10], which was already one of the leading tools for complexity analysis of (non-probabilistic) integer programs. The implementation is written in OCaml and uses the Python libraries MpMath [22] and SymPy [29] for solving linear equations and for finding the roots of the characteristic polynomial. In addition to the closed form for the exact expected runtime, our implementation can also compute the concrete number of expected loop iterations if the user specifies the initial values of the variables. For further details, a set of benchmarks, and to download our implementation, we refer to https://aprove-developers.github.io/recurrence/.

Example 34 (Computing the Exact Expected Runtime of $\mathcal{P}_{race}$ Automatically). For the tortoise and hare program $\mathcal{P}_{race}$ from Ex. 1, our implementation in KoAT computes the following expected runtime within 0.49 s on an Intel Core i7-6500 with 8 GB memory (when selecting a precision of 2 decimal places):

$$\begin{aligned} rt_{(t,h)}^{\mathcal{P}_{race}} = {} & 0.049 \cdot 0.65^{(t-h+1)} \cdot \sin(2.8 \cdot (t-h+1)) - 0.35 \cdot 0.65^{(t-h+1)} \cdot \cos(2.8 \cdot (t-h+1)) \\ & +0.15 \cdot 0.66^{(t-h+1)} \cdot \sin(2.2 \cdot (t-h+1)) - 0.35 \cdot 0.66^{(t-h+1)} \cdot \cos(2.2 \cdot (t-h+1)) \\ & +0.3 \cdot 0.7^{(t-h+1)} \cdot \sin(1.5 \cdot (t-h+1)) - 0.39 \cdot 0.7^{(t-h+1)} \cdot \cos(1.5\,(t-h+1)) \\ & +0.62 \cdot 0.75^{(t-h+1)} \cdot \sin(0.83 \cdot (t-h+1)) - 0.49 \cdot 0.75^{(t-h+1)} \cdot \cos(0.83 \cdot (t-h+1)) \\ & +\tfrac{2}{3} \cdot (t-h) \; + \; 2.3 \end{aligned}$$

So when starting in a state with $t = 1000$ and $h = 0$, according to our implementation the number of expected loop iterations is $rt_{(1000,0)}^{\mathcal{P}_{race}} = 670$.

Related Work. Many techniques to analyze (P)AST have been developed, which mostly rely on ranking supermartingales, e.g., [1, 5, 11, 13, 14, 18, 20, 28, 30]. Indeed, several of these works (e.g., [1, 5, 18, 20]) present complete criteria for (P)AST, although (P)AST is undecidable. However, the corresponding automation of these techniques is of course incomplete. In [14] it is shown that for affine probabilistic programs, a superclass of our CP programs, the existence of a linear ranking supermartingale is decidable. However, the existence of a linear ranking supermartingale is sufficient but not necessary for PAST or an at most linear expected runtime.

Classes of programs where termination is decidable have already been studied for deterministic programs. In [35] it was shown that for a class of linear loop programs over the reals, the halting problem is decidable. This result was transferred to the rationals [6] and under certain conditions to integer programs [6, 19, 31]. Termination analysis for probabilistic programs is substantially harder than for non-probabilistic ones [23]. Nevertheless, there is some previous work on classes of probabilistic programs where termination is decidable and asymptotic bounds on the expected runtime are computable. For instance, in [7] it was shown that AST is decidable for certain stochastic games and [12] presents an automatic approach for inferring asymptotic upper bounds on the expected runtime by considering uni- and bivariate recurrence equations.

However, our algorithm is the first which computes a general formula (i.e., a closed form) for the *exact* expected runtime of arbitrary CP programs. To our knowledge, up to now such a formula was only known for the very restricted special case of *bounded* simple random walks (cf. [17]), i.e., programs of the form on the right for some $1 \geq p \geq 0$ and some $b \in \mathbb{Z}$. Note that due to the *two* boundary conditions $x > 0$ and $b > x$, the resulting recurrence equation for the expected runtime of the program only has a *single* solution $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ that also satisfies $f(0) = 0$ and $f(b) = 0$. Hence, standard techniques for solving recurrence equations suffice to compute this solution. In contrast, we developed an algorithm to compute the exact expected runtime of *unbounded arbitrary* CP programs where the loop condition only has *one* boundary condition $x > 0$, i.e., x can grow infinitely large. For that reason, here the challenge is to find an algorithm which computes the *smallest* solution $f : \mathbb{Z} \to \mathbb{R}_{\geq 0}$ of the resulting recurrence equation. We showed that this can be done using the information on the asymptotic bounds of the expected runtime from Sect. 3 and 4.

```
while (b > x > 0) {
  x = x + 1   [p];
  x = x − 1   [1 − p];
}
```

Future Work. There are several directions for future work. In Sect. 4.1 we reduced CP programs to random walk programs. In future work, we will consider more advanced reductions in order to extend the class of probabilistic programs where termination and complexity are decidable. Moreover, we want to develop techniques to automatically *over- or under-approximate* the runtime of a program $\mathcal{P}$ by the runtimes of corresponding CP programs $\mathcal{P}_1$ and $\mathcal{P}_2$ such that $rt_{\boldsymbol{x}}^{\mathcal{P}_1} \leq rt_{\boldsymbol{x}}^{\mathcal{P}} \leq rt_{\boldsymbol{x}}^{\mathcal{P}_2}$ holds for all $\boldsymbol{x} \in \mathbb{Z}^r$. Furthermore, we will integrate the easy inference of runtime bounds for CP programs into existing techniques for analyzing more general probabilistic programs.

Acknowledgments We would like to thank Nicos Georgiou and Vladislav Vysotskiy for drawing our attention to Wald's Lemma and to the work of Frank Spitzer on random walks, and Benjamin Lucien Kaminski and Christoph Matheja for many helpful discussions. Furthermore, we thank Tom Küspert who helped with the implementation of our technique in our tool KoAT.

References

1. Agrawal, S., Chatterjee, K., Novotný, P.: Lexicographic ranking supermartingales: an efficient approach to termination of probabilistic programs. Proc. ACM Program. Lang. **2**(POPL), 34:1–34:32 (2018), https://doi.org/10.1145/3158122
2. Ash, R.B., Doleans-Dade, C.A.: Probability and Measure Theory. Elsevier/Academic Press (2000)
3. Bauer, H.: Probability Theory. Walter de Gruyter & Co. (1996)
4. Bazzi, L., Mitter, S.: The solution of linear probabilistic recurrence relations. Algorithmica **36**(1), 41–57 (2003), https://doi.org/10.1007/s00453-002-1003-4
5. Bournez, O., Garnier, F.: Proving positive almost-sure termination. In: Proc. RTA '05. pp. 323–337. LNCS 3467 (2005), https://doi.org/10.1007/978-3-540-32033-3_24
6. Braverman, M.: Termination of integer linear programs. In: Proc. CAV '06. pp. 372–385. LNCS 4144 (2006), https://doi.org/10.1007/11817963_34
7. Brázdil, T., Brozek, V., Etessami, K.: One-counter stochastic games. In: Proc. FSTTCS '10. pp. 108–119. LIPIcs 8 (2010), https://doi.org/10.4230/LIPIcs.FSTTCS.2010.108
8. Brázdil, T., Kucera, A., Novotný, P., Wojtczak, D.: Minimizing expected termination time in one-counter Markov decision processes. In: Proc. ICALP '12. pp. 141–152. LNCS 7392 (2012), https://doi.org/10.1007/978-3-642-31585-5_16
9. Brázdil, T., Esparza, J., Kiefer, S., Kucera, A.: Analyzing probabilistic pushdown automata. Formal Methods in System Design **43**(2), 124–163 (2013), https://doi.org/10.1007/s10703-012-0166-0
10. Brockschmidt, M., Emmes, F., Falke, S., Fuhs, C., Giesl, J.: Analyzing runtime and size complexity of integer programs. ACM Trans. Program. Lang. Syst. **38**(4), 13:1–13:50 (2016), https://doi.org/10.1145/2866575
11. Chakarov, A., Sankaranarayanan, S.: Probabilistic program analysis with martingales. In: Proc. CAV '13. pp. 511–526. LNCS 8044 (2013), https://doi.org/10.1007/978-3-642-39799-8_34
12. Chatterjee, K., Fu, H., Murhekar, A.: Automated recurrence analysis for almost-linear expected-runtime bounds. In: Proc. CAV '17. pp. 118–139. LNCS 10426 (2017), https://doi.org/10.1007/978-3-319-63387-9_6
13. Chatterjee, K., Novotný, P., Zikelic, D.: Stochastic invariants for probabilistic termination. In: Proc. POPL '17. pp. 145–160 (2017), https://doi.org/10.1145/3093333.3009873
14. Chatterjee, K., Fu, H., Novotný, P., Hasheminezhad, R.: Algorithmic analysis of qualitative and quantitative termination problems for affine probabilistic programs. ACM Trans. Program. Lang. Syst. **40**(2), 7:1–7:45 (2018), https://doi.org/10.1145/3174800
15. Elaydi, S.: An Introduction to Difference Equations. Springer (2005), https://doi.org/10.1007/0-387-27602-5
16. Esparza, J., Kucera, A., Mayr, R.: Quantitative analysis of probabilistic pushdown automata: Expectations and variances. In: Proc. LICS '05. pp. 117–126 (2005), https://doi.org/10.1109/LICS.2005.39
17. Feller, W.: An Introduction to Probability Theory and Its Applications, Probability and Mathematical Statistics, vol. 1. John Wiley & Sons (1950)
18. Fioriti, L.M.F., Hermanns, H.: Probabilistic termination: Soundness, completeness, and compositionality. In: Proc. POPL '15. pp. 489–501 (2015), https://doi.org/10.1145/2676726.2677001

19. Frohn, F., Giesl, J.: Termination of triangular integer loops is decidable. In: Proc. CAV '19. pp. 426–444. LNCS 11562 (2019), https://doi.org/10.1007/978-3-030-25543-5_24
20. Fu, H., Chatterjee, K.: Termination of nondeterministic probabilistic programs. In: Proc. VMCAI 2019. pp. 468–490. LNCS 11388 (2019), https://doi.org/10.1007/978-3-030-11245-5_22
21. Grimmett, G., Stirzaker, D.: Probability and Random Processes. Oxford University Press (2001)
22. Johansson, F., et al.: MpMath: a Python library for arbitrary-precision floating-point arithmetic, http://mpmath.org/
23. Kaminski, B.L., Katoen, J.: On the hardness of almost-sure termination. In: Proc. MFCS '15. pp. 307–318 (2015), https://doi.org/10.1007/978-3-662-48057-1_24
24. Kaminski, B.L., Katoen, J., Matheja, C., Olmedo, F.: Weakest precondition reasoning for expected run-times of probabilistic programs. In: Proc. ESOP '16. pp. 364–389. LNCS 9632 (2016), https://doi.org/10.1007/978-3-662-49498-1_15
25. Karp, R.M.: Probabilistic recurrence relations. J. ACM **41**(6), 1136–1150 (1994), https://doi.org/10.1145/195613.195632
26. Kozen, D.: Semantics of probabilistic programs. In: Proc. FOCS '79. pp. 101–114 (1979), https://doi.org/10.1109/SFCS.1979.38
27. McIver, A., Morgan, C.: Abstraction, Refinement and Proof for Probabilistic Systems. Springer (2005), https://doi.org/10.1007/b138392
28. McIver, A., Morgan, C., Kaminski, B.L., Katoen, J.: A new proof rule for almost-sure termination. Proc. ACM Program. Lang. **2**(POPL), 33:1–33:28 (2018), https://doi.org/10.1145/3158121
29. Meurer, A., Smith, C.P., Paprocki, M., Certík, O., Kirpichev, S.B., Rocklin, M., Kumar, A., Ivanov, S., Moore, J.K., Singh, S., Rathnayake, T., Vig, S., Granger, B.E., Muller, R.P., Bonazzi, F., Gupta, H., Vats, S., Johansson, F., Pedregosa, F., Curry, M.J., Terrel, A.R., Roucka, S., Saboo, A., Fernando, I., Kulal, S., Cimrman, R., Scopatz, A.M.: SymPy: symbolic computing in Python. PeerJ Computer Science **3**(e103) (2017), https://doi.org/10.7717/peerj-cs.103
30. Ngo, V.C., Carbonneaux, Q., Hoffmann, J.: Bounded expectations: resource analysis for probabilistic programs. In: Proc. PLDI '18. pp. 496–512 (2018), https://doi.org/10.1145/3192366.3192394, Extended Version available at https://arxiv.org/abs/1711.08847.
31. Ouaknine, J., Pinto, J.S., Worrell, J.: On termination of integer linear loops. In: Proc. SODA '15. pp. 957–969 (2015), https://doi.org/10.1137/1.9781611973730.65
32. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons (1994)
33. Spitzer, F.: Principles of Random Walk. Springer (1964), https://doi.org/10.1007/978-1-4757-4229-9
34. Tassarotti, J., Harper, R.: Verified tail bounds for randomized programs. In: Proc. ITP '18. pp. 560–578. LNCS 10895 (2018), https://doi.org/10.1007/978-3-319-94821-8_33
35. Tiwari, A.: Termination of linear programs. In: Proc. CAV '04. pp. 70–82. LNCS 3114 (2004), https://doi.org/10.1007/978-3-540-27813-9_6

Appendix

This appendix contains a collection of examples to demonstrate the application of our algorithm in App. A and all proofs (in App. B-E).

A Case Studies

In this section, we demonstrate our approach for the computation of the exact expected runtime on further examples.

Example 35 (Example with Direct Termination and Non-Constant Exact Runtime). *As an example with $p' > 0$ where the exact expected runtime is not constant, consider the following program $\mathcal{P}$.*

$$
\begin{array}{l}
\texttt{while}\ (x > 0)\ \{ \\
\quad x = x + 1 \quad [\frac{1}{8}]; \\
\quad x = x \qquad\quad [\frac{1}{2}]; \\
\quad x = x - 1 \quad [\frac{1}{4}]; \\
\quad x = 0 \qquad\quad [\frac{1}{8}]; \\
\}
\end{array}
$$

The characteristic polynomial is $\chi_{\mathcal{P}}(\lambda) = \frac{1}{8} \cdot \lambda^2 - \frac{1}{2} \cdot \lambda + \frac{1}{4}$. It has the $k + m = 1 + 1 = 2$ roots $2 \pm \sqrt{2}$. So the only root with absolute value ≤ 1 is $2 - \sqrt{2}$. By Thm. 29 we obtain $rt_x^{\mathcal{P}} = 0$ for $x \leq 0$ and

$$rt_x^{\mathcal{P}} = 8 + a_1 \cdot (2 - \sqrt{2})^x \quad \text{for } x > 0.$$

Here, a_1 is the unique solution of the linear equation $0 = 8 + a_1 \cdot (2 - \sqrt{2})^0 \cdot 0^0 = 8 + a_1$, i.e., $a_1 = -8$. So for $x > 0$ we have

$$rt_x^{\mathcal{P}} = 8 - 8 \cdot (2 - \sqrt{2})^x,$$

i.e., here the negative add-on $-8 \cdot (2 - \sqrt{2})^x$ is added to the upper bound 8.

Example 36 (Example with Complex Roots). *To show that complex roots are indeed possible, we apply Alg. 33 to the following program $\mathcal{P}$, where $p' = 0$ and $\mu_{\mathcal{P}} = -\frac{13}{30}$. Thus, $C_{lin} = \frac{30}{13}$ and $C(x) = \frac{30}{13} \cdot x$.*

$$
\begin{array}{l}
\texttt{while}\ (x > 0)\ \{ \\
\quad x = x + 1 \quad [\frac{5}{36}]; \\
\quad x = x \qquad\quad [\frac{1}{2}]; \\
\quad x = x - 1 \quad [\frac{13}{60}]; \\
\quad x = x - 2 \quad [\frac{7}{90}]; \\
\quad x = x - 3 \quad [\frac{1}{15}]; \\
\}
\end{array}
$$

The characteristic polynomial $\chi_{\mathcal{P}}(\lambda)$ has the roots 1, 3, and the two complex roots $\frac{-1\pm\sqrt{3}\,i}{5}$. Hence, the $k=3$ roots with absolute value ≤ 1 are 1 and $\frac{-1\pm\sqrt{3}\,i}{5}$. By Thm. 29 we obtain the following general solution:

$$f(x) = \tfrac{30}{13} \cdot x + a_1 + a_2 \cdot (\tfrac{-1+\sqrt{3}\,i}{5})^x + a_3 \cdot (\tfrac{-1-\sqrt{3}\,i}{5})^x \text{ for } x > -3$$

The coefficients a_1, a_2, a_3 are determined by the three linear equations $0 = f(x)$ for $-2 \leq x \leq 0$, cf. (11). They have the unique solution $a_1 = \frac{180}{169}$, $a_2 = -\frac{90}{169} - \frac{2}{169} \cdot \sqrt{3}\,i$, and $a_3 = -\frac{90}{169} + \frac{2}{169} \cdot \sqrt{3}\,i$. Thus, $b_2 = 2 \cdot \mathrm{Re}(a_2) = -\frac{180}{169}$, and $b_2' = -2 \cdot \mathrm{Im}(a_2) = \frac{4}{169} \cdot \sqrt{3}$. The polar representation of $\lambda = \frac{-1+\sqrt{3}\,i}{5}$ is $\frac{2}{5} \cdot e^{\frac{2\pi}{3} \cdot i}$. Hence, $\mathrm{Re}(\lambda^x) = (\frac{2}{5})^x \cdot \cos(\frac{2\pi}{3} \cdot x)$ and $\mathrm{Im}(\lambda^x) = (\frac{2}{5})^x \cdot \sin(\frac{2\pi}{3} \cdot x)$. Thus, we get $rt_x^{\mathcal{P}} = 0$ for $x \leq 0$ and for $x > 0$ we have

$$rt_x^{\mathcal{P}} = \tfrac{30}{13} \cdot x + \tfrac{180}{169} - \tfrac{180}{169} \cdot \left(\tfrac{2}{5}\right)^x \cdot \cos\left(\tfrac{2\pi}{3} \cdot x\right) + \tfrac{4}{169} \cdot \sqrt{3} \cdot \left(\tfrac{2}{5}\right)^x \cdot \sin\left(\tfrac{2\pi}{3} \cdot x\right).$$

Example 37 (Example with Root of Higher Multiplicity). As an example where the characteristic polynomial has a root with multiplicity greater than 1, consider the following program $\mathcal{P}$.

$$
\begin{array}{ll}
\texttt{while}\ (x > 0)\ \{ & \\
\quad x = x + 1 & [\tfrac{5}{21}]; \\
\quad x = x & [\tfrac{4}{7}]; \\
\quad x = x - 1 & [\tfrac{3}{35}]; \\
\quad x = x - 2 & [\tfrac{7}{75}]; \\
\quad x = x - 3 & [\tfrac{2}{175}]; \\
\} &
\end{array}
$$

We use the approach of Alg. 33 to infer the exact expected runtime. Step 1 is not necessary, since we already have a random walk program.

2. *We have $p' = 0$, $\mu_{\mathcal{P}} = -\frac{12}{175}$, and thus, $C_{lin} = \frac{175}{12}$.*
3. *The characteristic polynomial has the degree $k + m = 3 + 1 = 4$ and is given by $\chi_{\mathcal{P}}(\lambda) = \frac{5}{21} \cdot \lambda^4 - \frac{3}{7} \cdot \lambda^3 + \frac{3}{35} \cdot \lambda^2 + \frac{7}{75} \cdot \lambda + \frac{2}{175}$. It has the roots $\lambda_1 = 1$ with multiplicity 1, $\lambda_2 = \frac{6}{5}$ with multiplicity 1, and $\lambda_3 = -\frac{1}{5}$ with multiplicity 2. Hence, the three roots with absolute value ≤ 1 are 1 and $-\frac{1}{5}$ with multiplicity 2. As proved in Lemma 26 we have $1 + 2 = 3 = k$ such roots counted with multiplicity.*
4. *By Thm. 29, the general solution is*

$$f(x) = \tfrac{175}{12} \cdot x + a_{1,0} + a_{2,0} \cdot (-\tfrac{1}{5})^x + a_{2,1} \cdot x \cdot (-\tfrac{1}{5})^x \text{ for } x > -3.$$

The coefficients $a_{1,0}$, $a_{2,0}$, and $a_{2,1}$ are determined by the following linear equations, cf. (11):

$$
\begin{aligned}
0 &= f(0) = a_{1,0} + a_{2,0} \\
0 &= f(-1) = -\tfrac{175}{12} + a_{1,0} - 5 \cdot a_{2,0} + 5 \cdot a_{2,1} \\
0 &= f(-2) = -\tfrac{175}{6} + a_{1,0} + 25 \cdot a_{2,0} - 50 \cdot a_{2,1}
\end{aligned}
$$

They have the unique solution $a_{1,0} = \frac{175}{36}$, $a_{2,0} = -\frac{175}{36}$, *and* $a_{2,1} = -\frac{35}{12}$. *Hence,* $rt_x^{\mathcal{P}} = 0$ *for* $x \leq 0$ *and for* $x > 0$ *we have*

$$rt_x^{\mathcal{P}} = \tfrac{175}{12} x + \tfrac{175}{36} - \tfrac{175}{36} \cdot (-\tfrac{1}{5})^x - \tfrac{35}{12} \cdot x \cdot (-\tfrac{1}{5})^x.$$

Example 38 (Negative Binomial Loop from [28, Sect. 5.1]). Consider the following program $\mathcal{P}$ *from [28, Sect. 5.1].*

$$\begin{array}{ll} \texttt{while}\ (x > 0)\ \{ & \\ \quad x = x & [\frac{1}{2}]; \\ \quad x = x - 1 & [\frac{1}{2}]; \\ \} & \end{array}$$

The drift of this program is $\mu_{\mathcal{P}} = -\frac{1}{2} < 0$ *and by Thm. 18 we can conclude that the negative binomial loop is positive almost surely terminating. Furthermore, as* $k = 1$ *and* $m = 0$ *we obtain that the expected runtime* $rt_x^{\mathcal{P}}$ *of this program satisfies* $2 \cdot x \leq rt_x^{\mathcal{P}} \leq 2 \cdot x$ *for all* $x > 0$ *by Thm. 21, i.e.,*

$$rt_x^{\mathcal{P}} = \begin{cases} 2 \cdot x, & \text{if } x > 0 \\ 0, & \text{if } x \leq 0 \end{cases}$$

So with our approach, the expected runtime of this example can be determined with clearly less effort than with the technique presented in [28]. On the other hand, the reasoning of [28] can be applied to arbitrary probabilistic programs which may even include non-determinism.

Example 39 (Symmetric Random Walk). Consider the following program $\mathcal{P}$.

$$\begin{array}{ll} \texttt{while}\ (x > 0)\ \{ & \\ \quad x = x + 1 & [\frac{1}{2}]; \\ \quad x = x - 1 & [\frac{1}{2}]; \\ \} & \end{array}$$

One easily calculates the drift $\mu_{\mathcal{P}} = \frac{1}{2} - \frac{1}{2} = 0$. *So by Thm. 18 we immediately obtain the well-known result that this program is almost surely terminating but not positive almost surely terminating, i.e., the expected runtime is infinite.*

Example 40 (Example with Irrational Runtime from [14, Ex. 5.1]). Consider the following program $\mathcal{P}$ *which was presented in [14, Ex. 5.1] to show that expected runtimes can be irrational.*

$$\begin{array}{ll} \texttt{while}\ (x > 0)\ \{ & \\ \quad x = x + 1 & [\frac{1}{2}]; \\ \quad x = x - 2 & [\frac{1}{2}]; \\ \} & \end{array}$$

Its drift is $\mu_{\mathcal{P}} = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot (-2) = -\frac{1}{2} < 0$, *so by Thm. 18 this program is indeed PAST. As* $k = 2$, *we obtain the following bounds on the expected runtime by Thm. 21 for any positive initial value* $x > 0$*:*

$$2 \cdot x \leq rt_x^{\mathcal{P}} \leq 2 \cdot x + 2$$

The characteristic polynomial of this program is $\chi_{\mathcal{P}}(x) = \frac{1}{2} \cdot x^3 - x^2 + \frac{1}{2}$. *It has the three roots* 1, $\frac{1+\sqrt{5}}{2}$, *and* $\frac{1-\sqrt{5}}{2}$. *So the* $k = 2$ *roots of absolute value* ≤ 1 *are* 1 *and* $\frac{1-\sqrt{5}}{2}$. *By Thm. 29, the general solution is*

$$f(x) = 2 \cdot x + a_1 + a_2 \cdot (\tfrac{1-\sqrt{5}}{2})^x \text{ for } x > -2.$$

The coefficients a_1 *and* a_2 *are determined by the following equations:*

$$\begin{aligned} 0 &= f(0) = a_1 + a_2 \\ 0 &= f(-1) = -2 + a_1 + a_2 \cdot \tfrac{2}{1-\sqrt{5}} \end{aligned}$$

They have the unique solution $a_1 = 3 - \sqrt{5}$ *and* $a_2 = \sqrt{5} - 3$. *Hence, we infer the following exact expected runtime for* $x > 0$*:*

$$rt_x^{\mathcal{P}} = 2 \cdot x + 3 - \sqrt{5} + (\sqrt{5} - 3) \cdot (\tfrac{1-\sqrt{5}}{2})^x$$

So in particular, $rt_1^{\mathcal{P}} = 1 + \sqrt{5}$. *The expected runtime obtained in [14, Ex. 5.1] is slightly different (they obtain* $2 \cdot (5 + \sqrt{5})$*), because [14] counts the number of executed statements whereas we count loop iterations.*

Example 41 (Example from [30, Sect. 3.1]). Consider the following program $\mathcal{P}$. *It was used in [30, Sect. 3.1] to show how one can infer the expected runtime of a probabilistic program by solving a recurrence equation. However, the authors of [30] conclude that recurrence equations are not well suited for runtime analyses, while our paper shows that for CP programs, an automated runtime analysis based on recurrence equations is feasible.*

$$\begin{array}{lr} \texttt{while } (x > 0)\ \{ & \\ \quad x = x + 1 & [\frac{1}{4}]; \\ \quad x = x - 1 & [\frac{3}{4}]; \\ \} & \end{array}$$

Its drift is $\mu_{\mathcal{P}} = \frac{1}{4} \cdot 1 + \frac{3}{4} \cdot (-1) = -\frac{1}{2} < 0$, *so by Thm. 18 this program is indeed PAST. By Thm. 21, we can infer the following bounds on the expected runtime for any positive initial value* $x > 0$*:*

$$2 \cdot x \leq rt_x^{\mathcal{P}} \leq 2 \cdot x.$$

Hence, in this example we can directly conclude that for any $x > 0$ *the expected runtime is* $rt_x^{\mathcal{P}} = 2 \cdot x$, *without having to solve the corresponding recurrence equation with Thm. 29 resp. Alg. 33.*

B Proofs for Sect. 2

We begin with introducing some auxiliary definitions that will be needed in the proofs. To define the *run* of a program, we use the "Kronecker-Delta" where for any $\boldsymbol{y}, \boldsymbol{z} \in \mathbb{Z}^r$ with $\boldsymbol{y} \neq \boldsymbol{z}$ we have $\delta_{\boldsymbol{y},\boldsymbol{z}} = 0$ and $\delta_{\boldsymbol{y},\boldsymbol{y}} = 1$.

Definition 42 (Run of a Program). *For any program $\mathcal{P}$ as in Def. 2, a* run *is an infinite sequence $\langle \boldsymbol{z}_0, \boldsymbol{z}_1, \boldsymbol{z}_2, \ldots\rangle \in (\mathbb{Z}^r)^\omega$ and a* prefix run *is a finite sequence $\langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_j\rangle \in (\mathbb{Z}^r)^{j+1}$ for some $j \in \mathbb{N}$. For a prefix run π, its* cylinder set *$Cyl^{\mathbb{Z}^r}(\pi) \subseteq (\mathbb{Z}^r)^\omega$ consists of all runs with prefix π.*
For any initial value $\boldsymbol{x}_0 \in \mathbb{Z}^r$ of the program variables, we define a function $pr^{\mathcal{P}}_{\boldsymbol{x}_0}$ that maps any prefix run π to its probability (i.e., $0 \leq pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\pi) \leq 1$). Thus, for any prefix run $\langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_j\rangle$, let $pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\langle \boldsymbol{z}_0\rangle) = \delta_{\boldsymbol{x}_0,\boldsymbol{z}_0}$ and if $j \geq 1$, we define:

$$pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\langle \boldsymbol{z}_0, \ldots, \boldsymbol{z}_j\rangle) = \begin{cases} pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\langle \boldsymbol{z}_0, \ldots, \boldsymbol{z}_{j-1}\rangle) \cdot (p_{\boldsymbol{z}_j - \boldsymbol{z}_{j-1}}(\boldsymbol{z}_{j-1}) + \delta_{\boldsymbol{z}_j,\boldsymbol{d}} \cdot p'(\boldsymbol{z}_{j-1})), & \\ & \text{if } \boldsymbol{a} \bullet \boldsymbol{z}_{j-1} > b \\ pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\langle \boldsymbol{z}_0, \ldots, \boldsymbol{z}_{j-1}\rangle) \cdot \delta_{\boldsymbol{z}_{j-1},\boldsymbol{z}_j}, & \text{if } \boldsymbol{a} \bullet \boldsymbol{z}_{j-1} \leq b \end{cases}$$

Example 43 (Run in $\mathcal{P}_{race}$). For $\mathcal{P}_{race}$ from Ex. 1 and a start configuration where the tortoise is 10 steps ahead of the hare (e.g., $\boldsymbol{x}_0 = (11,1)$), the prefix run $\langle(11,1),(12,1),(13,6)\rangle$ has the probability $pr^{\mathcal{P}_{race}}_{(11,1)}(\langle(11,1),(12,1),(13,6)\rangle) = \delta_{(11,1),(11,1)} \cdot p_{(12,1)-(11,1)}(11,1) \cdot p_{(13,6)-(12,1)}(12,1) = p_{(1,0)}(11,1) \cdot p_{(1,5)}(12,1) = \frac{6}{11} \cdot \frac{1}{22} = \frac{3}{121}$. So we take into account whether the prefix run starts with $\boldsymbol{x}_0 = (11,1)$ and multiply the probability to get from $\boldsymbol{x} = (11,1)$ to $\boldsymbol{x} = (12,1)$ with the probability to get from $\boldsymbol{x} = (12,1)$ to $\boldsymbol{x} = (13,6)$.

In our setting, we regard a measurable space $(\Omega, \mathfrak{F})$ where Ω is the set of runs $(\mathbb{Z}^r)^\omega$ and we want to measure the probability that a run starts with a certain sequence π of numbers. So we regard the smallest σ-field $\mathfrak{F}^{\mathbb{Z}^r}$ that contains all cylinder sets $Cyl^{\mathbb{Z}^r}(\pi)$ for all prefix runs π. Moreover, we consider the probability space $((\mathbb{Z}^r)^\omega, \mathfrak{F}^{\mathbb{Z}^r}, \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0})$. Here, the probability measure $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}$ for a program $\mathcal{P}$ is defined such that the probability that a run is in $Cyl^{\mathbb{Z}^r}(\pi)$ is the probability $pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\pi)$ of the prefix run π.

Definition 44 (Probability Measure for a Program). *For any program $\mathcal{P}$ as in Def. 2 and any $\boldsymbol{x}_0 \in \mathbb{Z}^r$, let $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0} : \mathfrak{F}^{\mathbb{Z}^r} \to [0,1]$ be the unique* probability measure *such that we have $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(Cyl^{\mathbb{Z}^r}(\pi)) = pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\pi)$ for any prefix run π.*

Example 45 (Probability Measure for $\mathcal{P}_{race}$). $Cyl^{\mathbb{Z}^2}(\langle(11,1),(12,1),(13,6)\rangle)$ consists of all runs that start with $(11,1)$, $(12,1)$, $(13,6)$. If the initial value is $\boldsymbol{x}_0 = (11,1)$, then the probability that a run is in $Cyl^{\mathbb{Z}^2}(\langle(11,1),(12,1),(13,6)\rangle)$ is

$$\mathbb{P}^{\mathcal{P}_{race}}_{(11,1)}(Cyl^{\mathbb{Z}^2}(\langle(11,1),(12,1),(13,6)\rangle)) = pr^{\mathcal{P}_{race}}_{(11,1)}(\langle(11,1),(12,1),(13,6)\rangle) = \tfrac{3}{121}.$$

Now we introduce a stochastic process $\mathbf{X}^{\mathbb{Z}^r}$ (i.e., a family of random variables $X^{\mathbb{Z}^r}_j$) which corresponds to the values of the program variables during a run.

Definition 46 (Stochastic Process $\mathbf{X}^{\mathbb{Z}^r}$). *For $r \geq 1$, let $\mathbf{X}^{\mathbb{Z}^r} = (X^{\mathbb{Z}^r}_j)_{j\in\mathbb{N}}$ where $X^{\mathbb{Z}^r}_j : (\mathbb{Z}^r)^\omega \to \mathbb{Z}^r$ is defined as $X^{\mathbb{Z}^r}_j(\langle \boldsymbol{z}_0, \ldots, \boldsymbol{z}_j, \ldots\rangle) = \boldsymbol{z}_j$, i.e., when applied to a run, $X^{\mathbb{Z}^r}_j$ returns the values of the program variables after the j-th loop iteration.*

So $X_0^{\mathbb{Z}^2}(\langle(11,1),(12,1),\ldots\rangle) = (11,1)$ and $X_1^{\mathbb{Z}^2}(\langle(11,1),(12,1),\ldots\rangle) = (12,1)$.

Using $\mathbf{X}^{\mathbb{Z}^r}$, the *termination time* of a program (cf. Def. 3) can also be defined as $T^{\mathcal{P}}(\pi) = \inf\{j \in \mathbb{N} \mid \boldsymbol{a} \bullet X_j^{\mathbb{Z}^r}(\pi) \leq b\}$ for any $\pi \in (\mathbb{Z}^r)^\omega$. As shown in Def. 4, the termination time is needed to define the expected runtime of a program. We first prove that if the initial values $\boldsymbol{x}_0$ violate the loop guard, then the expected runtime is trivially 0.

Corollary 6 (Expected Runtime for Violating Initial Values). *For any program $\mathcal{P}$ as in Def. 2 and any $\boldsymbol{x}_0 \in \mathbb{Z}^r$ with $\boldsymbol{a} \bullet \boldsymbol{x}_0 \leq b$, we have $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = 0$.*

Proof. We have $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(X_0^{\mathbb{Z}^r} = \boldsymbol{x}_0) = \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}((X_0^{\mathbb{Z}^r})^{-1}(\{\boldsymbol{x}_0\})) = \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(Cyl^{\mathbb{Z}^r}(\boldsymbol{x}_0)) = pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\boldsymbol{x}_0) = \delta_{\boldsymbol{x}_0,\boldsymbol{x}_0} = 1$. Thus, for $\boldsymbol{x}_0$ with $\boldsymbol{a} \bullet \boldsymbol{x}_0 \leq b$, we obtain $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} = 0) = \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(\boldsymbol{a} \bullet X_0^{\mathbb{Z}^r} \leq b) \leq \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(X_0^{\mathbb{Z}^r} = \boldsymbol{x}_0) = 1$ and hence $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{E}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(T^{\mathcal{P}}\right) = 0$. □

To prove Thm. 9 we show how to translate any probabilistic program into a Markov Decision Process (MDP) and then reuse existing corresponding results for MDPs [32]. In this section we recapitulate the needed concepts for MDPs and after the introduction of any concept, we show how it is related to the corresponding notions for probabilistic programs.

We consider infinite time horizon MDPs, where we restrict ourselves to deterministic MDPs without final states, i.e., to *Discrete Time Markov Chains (DTMCs)*. So there is one unique action for every state of the MDP.

Definition 47 (Discrete Time Markov Chain). *A* Discrete Time Markov Chain (DTMC) *without final states $\mathcal{M} = (\mathcal{S}, P, rew)$ consists of the following components:*

- *$\mathcal{S}$ is a set of states.*
- *$P : \mathcal{S} \times \mathcal{S} \to [0,1]$ is a transition probability function such that for all states $s \in \mathcal{S}$ we have $\sum_{s' \in \mathcal{S}} P(s,s') = 1$.*
- *$rew : \mathcal{S} \to \mathbb{R}$ is the reward function.*

Def. 48 shows how to translate any probabilistic program $\mathcal{P}$ to a corresponding DTMC $\mathcal{M}_{\mathcal{P}}$. This is possible for our notion of probabilistic programs, because the values of the program variables only depend on their values in the previous loop iteration. To ease notation, let the probabilities $p_{\boldsymbol{c}}(\boldsymbol{x})$ be constant zero for all $\boldsymbol{c} \in \mathbb{Z}^r \setminus \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n\}$.

Definition 48 (Translating Probabilistic Programs to DTMCs). *Let $\mathcal{P}$ be a program as in Def. 2. Its* corresponding DTMC *$\mathcal{M}_{\mathcal{P}} = (\mathcal{S}, P, rew)$ is given by*

- *$\mathcal{S} = \mathbb{Z}^r$*
- *For states satisfying the loop guard, the probability function P is induced by the probabilities $p_{\boldsymbol{c}_j}$, and for states that do not satisfy the loop guard, the probability to remain in the state is 1:*

$$P(s,s') = \begin{cases} p_{s'-s}(s) + \delta_{s',\boldsymbol{d}} \cdot p'(s), & \text{if } \boldsymbol{a} \bullet s > b \\ \delta_{s,s'}, & \text{if } \boldsymbol{a} \bullet s \leq b \end{cases}$$

- *The reward function is given by* $rew(s) = \begin{cases} 1, & \text{if } \boldsymbol{a} \bullet s > b \\ 0, & \text{if } \boldsymbol{a} \bullet s \leq b \end{cases}$

For a DTMC $\mathcal{M} = (\mathcal{S}, P, rew)$ and each initial state $\boldsymbol{x}_0 \in \mathcal{S}$, we examine a stochastic process $\mathbf{X}^{\mathcal{S}}$ using a probability measure $\mathbb{P}^{\mathcal{M}}_{\boldsymbol{x}_0}$ for the measurable space $(\mathcal{S}^\omega, \mathfrak{F}^{\mathcal{S}})$. The definitions of $\mathfrak{F}^{\mathcal{S}}$, $\mathbb{P}^{\mathcal{M}}_{\boldsymbol{x}_0}$, and $\mathbf{X}^{\mathcal{S}}$ are generalizations of the corresponding definitions from Sect. 2 to arbitrary state spaces.
Moreover, instead of (prefix) runs we now regard *histories* resp. *sample paths* and instead of the probability $pr^{\mathcal{P}}_{\boldsymbol{x}_0}$ of a run with the initial variable assignment $\boldsymbol{x}_0$ we regard the probability $pr^{\mathcal{M}}_{x_0}$ of a sample path with the initial state x_0.

Definition 49 (Probability Measure for a DTMC). *Let* $\mathcal{M} = (\mathcal{S}, P, rew)$ *be a DTMC.*

- *A* sample path *is an infinite sequence* $\langle s_0, s_1, s_2, \ldots \rangle \in \mathcal{S}^\omega$ *and a* history *is a finite sequence* $\langle s_0, s_1, \ldots, s_j \rangle \in \mathcal{S}^{j+1}$ *for some* $j \in \mathbb{N}$*. The* cylinder set $Cyl^{\mathcal{S}}(\pi)$ *of a history* π *consists of all sample paths with prefix* π*.*
- *For any* $x_0 \in \mathcal{S}$*,* $pr^{\mathcal{M}}_{x_0} : \bigcup_{j \in \mathbb{N}} \mathcal{S}^{j+1} \to [0,1]$ *is the function that maps any history* $\langle s_0, \ldots, s_j \rangle$ *to its probability if* x_0 *is the initial state. Thus, let* $pr^{\mathcal{M}}_{x_0}(\langle s_0 \rangle) = \delta_{x_0, s_0}$ *and if* $j \geq 1$*, we define:*

$$pr^{\mathcal{M}}_{x_0}(\langle s_0, \ldots, s_j \rangle) = pr^{\mathcal{M}}_{x_0}(\langle s_0, \ldots, s_{j-1} \rangle) \cdot P(s_{j-1}, s_j)$$

- *The (canonical)* measurable space *for a DTMC is* $(\mathcal{S}^\omega, \mathfrak{F}^{\mathcal{S}})$*, where* $\mathfrak{F}^{\mathcal{S}}$ *is the smallest* σ*-field containing all cylinder sets* $Cyl^{\mathcal{S}}(\pi)$ *for all histories* π*.*
- *For any* $x_0 \in \mathcal{S}$*, the* probability measure $pr^{\mathcal{M}}_{x_0} : \mathfrak{F}^{\mathcal{S}} \to [0,1]$ for the DTMC $\mathcal{M}$ and the initial state x_0 *is the unique probability measure such that for any history* π *we have* $\mathbb{P}^{\mathcal{M}}_{x_0}(Cyl^{\mathcal{S}}(\pi)) = pr^{\mathcal{M}}_{x_0}(\pi)$*.*
- *The stochastic process* $\mathbf{X}^{\mathcal{S}} = (X^{\mathcal{S}}_j)_{j \in \mathbb{N}}$ *is defined as* $X^{\mathcal{S}}_j : \mathcal{S}^\omega \to \mathcal{S}$*, where* $X^{\mathcal{S}}_j(s_0, \ldots, s_j, \ldots) = s_j$*.*

The following corollary shows that for any probabilistic program $\mathcal{P}$, the probability spaces for $\mathcal{P}$ and for its corresponding DTMC $\mathcal{M}_{\mathcal{P}}$ are the same.

Corollary 50 ($\mathcal{P}$ and $\mathcal{M}_{\mathcal{P}}$ Have the Same Probability Measure). *For any program* $\mathcal{P}$ *as in Def. 2 and its corresponding DTMC* $\mathcal{M}_{\mathcal{P}}$*, the corresponding probability spaces are the same. So in particular, we have* $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{P}^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}$ *for any* $\boldsymbol{x}_0 \in \mathbb{Z}^r$*.*

Proof. By Def. 48 and 49, the measurable space for $\mathcal{M}_{\mathcal{P}}$ is $((\mathbb{Z}^r)^\omega, \mathfrak{F}^{\mathbb{Z}^r})$, which is also the measurable space for $\mathcal{P}$. Moreover, Def. 49 implies $pr^{\mathcal{P}}_{\boldsymbol{x}_0} = pr^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}$ and thus, $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{P}^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}$ for any $\boldsymbol{x}_0 \in \mathbb{Z}^r$. □

For DTMCs, one is interested in the *expected total reward*. For a DTMC $\mathcal{M} = (\mathcal{S}, P, rew)$ and the stochastic process $\mathbf{X}^{\mathcal{S}}$, the expected total reward maps any initial state $s_0 \in \mathcal{S}$ to the expected value of $\sum_{j \in \mathbb{N}} rew(X^{\mathcal{S}}_j)$ under the probability measure $\mathbb{P}^{\mathcal{M}}_{s_0}$ (if this expected value exists). Note that if $rew(s) \geq 0$ for all $s \in \mathcal{S}$, then the sum $\sum_{j\in\mathbb{N}} rew(X^{\mathcal{S}}_j) : \mathcal{S}^\omega \to \overline{\mathbb{R}_{\geq 0}}$ is a non-negative[5] random variable. Hence, its expected value under the probability measure $\mathbb{P}^{\mathcal{M}}_{s_0}$ is well defined. In particular, this holds for the DTMCs $\mathcal{M}_{\mathcal{P}}$ corresponding to programs $\mathcal{P}$, because for any run $\pi = \langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots\rangle \in (\mathbb{Z}^r)^\omega$, $rew(X^{\mathbb{Z}^r}_j(\pi)) = rew(\boldsymbol{z}_j)$ is 1 if the j-th tuple $\boldsymbol{z}_j$ in the run does not violate the loop condition $\boldsymbol{a} \bullet \boldsymbol{z}_j > b$ and 0, otherwise (i.e., $rew(\boldsymbol{z}) \in \{0,1\}$ for all $\boldsymbol{z} \in \mathbb{Z}^r$).

Definition 51 (Expected Total Reward). *Let $\mathcal{M} = (\mathcal{S}, P, rew)$ be a DTMC. For any $s_0 \in \mathcal{S}$, the* expected total reward $tr^{\mathcal{M}}_{s_0} \in \mathbb{R} \cup \{-\infty, \infty\}$ *of $\mathcal{M}$ is*

$$tr^{\mathcal{M}}_{s_0} \;=\; \lim_{u\to\infty} \mathbb{E}^{\mathcal{M}}_{s_0}\left(\sum_{0\leq j\leq u} rew(X^{\mathcal{S}}_j)\right)$$

whenever this limit exists in $\mathbb{R} \cup \{-\infty, \infty\}$. As argued above, the limit always exists in the special case of non-negative rewards. Therefore, in the case where $rew(s) \in \{0,1\}$ for all $s \in \mathcal{S}$, we have

$$tr^{\mathcal{M}}_{s_0} = \sum_{u\in\overline{\mathbb{N}}} u \cdot \mathbb{P}^{\mathcal{M}}_{s_0}(\sum_{j\in\mathbb{N}} rew(X^{\mathcal{S}}_j) = u).$$

The following lemma shows the connection between the termination time and the total reward of a run. In the following, we say that a run $\pi = \langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots\rangle$ is *constant on violating states* if $\boldsymbol{a} \bullet \boldsymbol{z}_j \leq b$ implies $\boldsymbol{z}_j = \boldsymbol{z}_{j+1}$ for all $j \in \mathbb{N}$.

Lemma 52 (Total Reward is Termination Time). *Let $\mathcal{P}$ be a program as in Def. 2. For every run π that is constant on violating states, we have $\sum_{j\in\mathbb{N}} rew(X^{\mathbb{Z}^r}_j(\pi)) = T^{\mathcal{P}}(\pi)$.*

Proof. First, we show that the equality holds for runs $\pi = \langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots\rangle$ where $T^{\mathcal{P}}(\pi) = u < \infty$. So $\boldsymbol{a} \bullet \boldsymbol{z}_j > b$ for all $j < u$ and since π is constant on violating states, we have $\boldsymbol{a} \bullet \boldsymbol{z}_j \leq b$ for all $j \geq u$. Here we obtain

$$\begin{aligned}
\sum_{j\in\mathbb{N}} rew(X^{\mathbb{Z}^r}_j(\pi)) &= \sum_{j\in\mathbb{N}} rew(\boldsymbol{z}_j)\\
&= \sum_{0\leq j<u} 1 + \sum_{j\geq u} 0\\
&= u\\
&= T^{\mathcal{P}}(\langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots\rangle)\\
&= T^{\mathcal{P}}(\pi).
\end{aligned}$$

[5] The non-negativity of rew ensures that the infinite sum of all $rew(X^{\mathcal{S}}_j)$ is a value in $\overline{\mathbb{R}_{\geq 0}}$. In contrast, if we have positive and negative rewards, then the infinite sum might diverge and neither converge to $-\infty$ nor to ∞.

Now we consider a run $\pi = \langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots \rangle$ such that $T^{\mathcal{P}}(\pi) = \infty$, i.e., $\boldsymbol{a} \bullet \boldsymbol{z}_j > b$ for all $j \in \mathbb{N}$. Then we have

$$\begin{aligned}\sum_{j\in\mathbb{N}} rew(X_j^{\mathbb{Z}^r}(\pi)) &= \sum_{j\in\mathbb{N}} rew(\boldsymbol{z}_j)\\ &= \sum_{j\in\mathbb{N}} 1\\ &= \infty\\ &= T^{\mathcal{P}}(\langle \boldsymbol{z}_0, \boldsymbol{z}_1, \ldots\rangle)\\ &= T^{\mathcal{P}}(\pi). \qquad \square\end{aligned}$$

With Cor. 50 and Lemma 52 we can show that the expected runtime of a program $\mathcal{P}$ is identical to the expected total reward of its corresponding DTMC $\mathcal{M}_{\mathcal{P}}$. This is the crucial theorem which allows us to apply results on DTMCs also for probabilistic programs.

Theorem 53 (Expected Total Reward is Expected Runtime). *For any program $\mathcal{P}$ as in Def. 2, the expected runtime of $\mathcal{P}$ and the expected total reward of the corresponding DTMC $\mathcal{M}_{\mathcal{P}}$ are the same, i.e., for any $\boldsymbol{x}_0 \in \mathbb{Z}^r$ we have $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = tr^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}$.*

Proof. Due to Def. 51 we have $tr^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0} = \sum_{u\in\overline{\mathbb{N}}} u \cdot \mathbb{P}^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}(A_u)$, where $A_u = \{\pi \in (\mathbb{Z}^r)^\omega \mid \sum_{j\in\mathbb{N}} rew(X_j^{\mathbb{Z}^r}(\pi)) = u\}$. Note that $pr^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}(\pi) = 0$ if π is not constant on violating states. Thus, $\mathbb{P}^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}(A_u) = \mathbb{P}^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}(A'_u)$ where

$$\begin{aligned}A'_u &= \{\pi \in (\mathbb{Z}^r)^\omega \mid \textstyle\sum_{j\in\mathbb{N}} rew(X_j^{\mathbb{Z}^r}(\pi)) = u \text{ and } \pi \text{ is constant on violating states}\}\\ &= \{\pi \in (\mathbb{Z}^r)^\omega \mid T^{\mathcal{P}}(\pi) = u \text{ and } \pi \text{ is constant on violating states}\} \text{ by Lemma 52.}\end{aligned}$$

Hence, we obtain

$$\begin{aligned}tr^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0} &= \sum_{u\in\overline{\mathbb{N}}} u \cdot \mathbb{P}^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0}(A'_u)\\ &= \sum_{u\in\overline{\mathbb{N}}} u \cdot \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(A'_u) \quad \text{by Cor. 50.}\end{aligned}$$

Note that $pr^{\mathcal{P}}_{\boldsymbol{x}_0}(\pi) = 0$ if π is not constant on violating states. Thus, $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(A'_u) = \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(A''_u)$ where $A''_u = \{\pi \in (\mathbb{Z}^r)^\omega \mid T^{\mathcal{P}}(\pi) = u\}$. So we get

$$\begin{aligned}tr^{\mathcal{M}_{\mathcal{P}}}_{\boldsymbol{x}_0} &= \sum_{u\in\overline{\mathbb{N}}} u \cdot \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(A''_u)\\ &= \sum_{u\in\overline{\mathbb{N}}} u \cdot \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} = u)\\ &= \mathbb{E}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(T^{\mathcal{P}}\right)\\ &= rt^{\mathcal{P}}_{\boldsymbol{x}_0}. \qquad \square\end{aligned}$$

Now we introduce the transformer $\mathcal{L}$ that is used for DTMCs and corresponds to the expected runtime transformer for probabilistic programs. In the following, we restrict ourselves to DTMCs with non-negative rewards to ensure that the expected total reward exists.

Definition 54 ($\mathcal{L}^{\mathcal{M}}$, cf. [32, Eq. 7.1.5]). *Let $\mathcal{M} = (\mathcal{S}, P, rew)$ be a DTMC with only non-negative rewards. We define the mapping $\mathcal{L}^{\mathcal{M}} : (\mathcal{S} \to \overline{\mathbb{R}_{\geq 0}}) \to (\mathcal{S} \to \overline{\mathbb{R}_{\geq 0}})$ such that for every function $f : \mathcal{S} \to \overline{\mathbb{R}_{\geq 0}}$ and every $s \in \mathcal{S}$, we have*

$$\mathcal{L}^{\mathcal{M}}(f)(s) = rew(s) + \sum_{s' \in \mathcal{S}} P(s, s') \cdot f(s').$$

The following corollary shows that the expected runtime transformer $\mathcal{L}^{\mathcal{P}}$ of a program $\mathcal{P}$ is the same as the transformer $\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}$ of the corresponding DTMC $\mathcal{M}_{\mathcal{P}}$.

Corollary 55 ($\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}$ is Expected Runtime Transformer $\mathcal{L}^{P}$). *For any program $\mathcal{P}$, the expected runtime transformer $\mathcal{L}^{\mathcal{P}}$ from Def. 7 is identical to the transformer $\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}$ from Def. 54.*

Proof. Let $\mathcal{P}$ be a program as in Def. 2 and let $\mathcal{M}_{\mathcal{P}} = (\mathbb{Z}^r, P, rew)$. Consider an arbitrary function $f : \mathbb{Z}^r \to \overline{\mathbb{R}_{\geq 0}}$ and an $s \in \mathbb{Z}^r$. If $\boldsymbol{a} \bullet s \leq b$ then $rew(s) = 0$, $P(s, s) = 1$, and $P(s, s') = 0$ for $s' \neq s$. Hence

$$\begin{aligned}
\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}(f)(s) &= rew(s) + \sum_{s' \in \mathcal{S}} P(s, s') \cdot f(s') \\
&= rew(s) + f(s) \\
&= 0 + f(s) \\
&= f(s) \\
&= \mathcal{L}^{\mathcal{P}}(f)(s).
\end{aligned}$$

If $\boldsymbol{a} \bullet s > b$ then

$$\begin{aligned}
\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}(f)(s) &= rew(s) + \sum_{s' \in \mathcal{S}} P(s, s') \cdot f(s') \\
&= 1 + \sum_{s' \in \mathcal{S}} (p_{s'-s}(s) + \delta_{s', \boldsymbol{d}} \cdot p'(s)) \cdot f(s') \\
&= 1 + \sum_{1 \leq j \leq n} p_{\boldsymbol{c}_j}(s) \cdot f(s + \boldsymbol{c}_j) + p'(s) \cdot f(\boldsymbol{d}) \\
&= \mathcal{L}^{\mathcal{P}}(f)(s).
\end{aligned}$$

□

Now that we know that the transformers $\mathcal{L}^{\mathcal{P}}$ and $\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}$ are the same, we can use existing results on DTMCs to obtain results for programs $\mathcal{P}$. More precisely, for any DTMC $\mathcal{M} = (\mathcal{S}, P, rew)$ with only non-negative rewards, it is known that the expected total reward function $tr^{\mathcal{M}} : \mathcal{S} \to \overline{\mathbb{R}_{\geq 0}}$ with $tr^{\mathcal{M}}(s_0) = tr^{\mathcal{M}}_{s_0}$ for any $s_0 \in \mathcal{S}$ is a fixpoint of $\mathcal{M}$'s transformer $\mathcal{L}^{\mathcal{M}}$.

Theorem 56 (Expected Total Reward is Fixpoint). *Let $\mathcal{M}$ be a DTMC with only non-negative rewards. Then $tr^{\mathcal{M}}$ is a fixpoint of $\mathcal{L}^{\mathcal{M}}$.*

Proof. The proof can be found in [32, Thm. 7.1.3]. Note that it requires the assumption that the expected total reward exists [32, Assumption 7.1.1] which is ensured by a non-negative reward function. □

Moreover, the expected total reward function is smaller or equal than any other fixpoint of $\mathcal{L}^{\mathcal{M}}$ (and than every function f which satisfies the inequality $f \geq \mathcal{L}^{\mathcal{M}}(f)$).

Theorem 57 (Expected Total Reward is Smaller Than Other Fixpoints). *Let $\mathcal{M} = (\mathcal{S}, P, rew)$ be a DTMC with only non-negative rewards and let there be a function $f : \mathcal{S} \to \overline{\mathbb{R}_{\geq 0}}$ such that $f \geq \mathcal{L}^{\mathcal{M}}(f)$. Then $f \geq tr^{\mathcal{M}}$.*

Proof. The proof of the finite case, i.e., $f(s) < \infty$ for all $s \in \mathcal{S}$, can be found in [32, Thm. 7.2.2]. Note that in our case there is a unique strategy (since we restrict ourselves to DTMCs) and we have only non-negative rewards. Therefore, the proof holds for functions f that map to infinity as well. □

Thm. 56 and 57 imply that the expected total reward function $tr^{\mathcal{M}}$ is the least fixpoint of the transformer $\mathcal{L}^{\mathcal{M}}$.

Corollary 58 (Expected Total Reward is Least Fixpoint). *Let $\mathcal{M} = (\mathcal{S}, P, rew)$ be a DTMC with only non-negative rewards. Then $tr^{\mathcal{M}}$ is the least fixpoint of $\mathcal{L}^{\mathcal{M}}$, i.e., for any $s_0 \in \mathcal{S}$ we have $\operatorname{lfp}(\mathcal{L}^{\mathcal{M}})(s_0) = tr^{\mathcal{M}}_{s_0}$.*

The following theorem shows that $\mathcal{L}^{\mathcal{M}}$ is continuous for any DTMC $\mathcal{M}$ with only non-negative rewards. This is needed to apply Kleene's Fixpoint Theorem, i.e., to show that the least fixpoint of $\mathcal{L}^{\mathcal{M}}$ is $\sup\{\mathfrak{o}, \mathcal{L}^{\mathcal{M}}(\mathfrak{o}), (\mathcal{L}^{\mathcal{M}})^2(\mathfrak{o}), \ldots\}$.

Theorem 59 (Continuity of $\mathcal{L}^{\mathcal{M}}$, cf. [32, Lemma 7.1.5.c]). *Let $\mathcal{M}$ be a DTMC with only non-negative rewards. Then $\mathcal{L}^{\mathcal{M}}$ is continuous.*

Proof. Let $S = \{f_0, f_1, \ldots\}$ be a chain in $\mathcal{S} \to \overline{\mathbb{R}_{\geq 0}}$, i.e., we have $f_j \leq f_{j+1}$ for all $j \in \mathbb{N}$. Then $(\sup S)$ is the function $(\sup S) : \mathcal{S} \to \overline{\mathbb{R}_{\geq 0}}$ with $(\sup S)(s) = \sup\limits_{j \in \mathbb{N}} (f_j(s))$ for all $s \in \mathcal{S}$. Therefore, for any $s \in \mathcal{S}$ we have

$$\begin{aligned}
\mathcal{L}^{\mathcal{M}}(\sup S)(s) &= rew(s) + \sum_{s' \in \mathcal{S}} P(s, s') \cdot (\sup S)(s') \\
&= rew(s) + \sum_{s' \in \mathcal{S}} P(s, s') \cdot \sup_{j \in \mathbb{N}} (f_j(s')) \\
&= \sup_{j \in \mathbb{N}} \left(rew(s) + \sum_{s' \in \mathcal{S}} P(s, s') \cdot f_j(s') \right) \text{as all operations are linear} \\
&= \sup_{j \in \mathbb{N}} \left(\mathcal{L}^{\mathcal{M}}(f_j)(s) \right) \\
&= (\sup \mathcal{L}^{\mathcal{M}}(S))(s),
\end{aligned}$$

where $\mathcal{L}^{\mathcal{M}}(S) = \{\mathcal{L}^{\mathcal{M}}(f_0), \mathcal{L}^{\mathcal{M}}(f_1), \ldots\}$. □

Now we can prove Thm. 9 which states that the expected runtime of a program $\mathcal{P}$ is the least fixpoint of its expected runtime transformer $\mathcal{L}^{\mathcal{P}}$ and that it can be obtained as the supremum of $\{\mathfrak{o}, \mathcal{L}^{\mathcal{P}}(\mathfrak{o}), (\mathcal{L}^{\mathcal{P}})^2(\mathfrak{o}), \ldots\}$. As usual, a function $f : \mathbb{Z}^r \to \overline{\mathbb{R}_{\geq 0}}$ is a *fixpoint* of $\mathcal{L}^{\mathcal{P}}$ if $\mathcal{L}^{\mathcal{P}}(f) = f$. Such a fixpoint f is the *least* fixpoint of $\mathcal{L}^{\mathcal{P}}$ (i.e., $f = \operatorname{lfp}(\mathcal{L}^{\mathcal{P}})$) if $f \leq g$ for any other fixpoint g of $\mathcal{L}^{\mathcal{P}}$.

Theorem 9 (Connection Between Expected Runtime and Least Fixpoint of $\mathcal{L}^{\mathcal{P}}$, cf. [32]). *For any $\mathcal{P}$ as in Def. 2, the expected runtime transformer $\mathcal{L}^{\mathcal{P}}$ is continuous. Thus, it has a least fixpoint $\operatorname{lfp}(\mathcal{L}^{P}) : \mathbb{Z}^r \to \overline{\mathbb{R}_{\geq 0}}$ with $\operatorname{lfp}(\mathcal{L}^{\mathcal{P}}) = \sup\{\mathfrak{o}, \mathcal{L}^{\mathcal{P}}(\mathfrak{o}), (\mathcal{L}^{\mathcal{P}})^2(\mathfrak{o}), \ldots\}$. Moreover, the least fixpoint of $\mathcal{L}^{\mathcal{P}}$ is the expected runtime of $\mathcal{P}$, i.e., for any $\boldsymbol{x}_0 \in \mathbb{Z}^r$, we have $\operatorname{lfp}(\mathcal{L}^{\mathcal{P}})(\boldsymbol{x}_0) = rt^{\mathcal{P}}_{\boldsymbol{x}_0}$.*

Proof. By Thm. 53, the expected runtime of $\mathcal{P}$ is the same as the expected total reward of the corresponding DTMC $\mathcal{M}_P$. Cor. 58 showed that the expected total reward is the least fixpoint of the transformer $\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}$, and $\mathcal{L}^{\mathcal{M}_{\mathcal{P}}}$ is the same as the expected runtime transformer $\mathcal{L}^{\mathcal{P}}$ due to Cor. 55.
As the continuity of $\mathcal{L}^{\mathcal{M}_P} = \mathcal{L}^{P}$ was shown in Thm. 59, by Kleene's Fixpoint Theorem we have $\operatorname{lfp}(\mathcal{L}^{P}) = \sup\{\mathfrak{o}, \mathcal{L}^{P}(\mathfrak{o}), (\mathcal{L}^{P})^2(\mathfrak{o}), \ldots\}$. $\square$

C Proofs for Sect. 3

Theorem 10 (PAST and Expected Runtime for Programs With Direct Termination). *Let $\mathcal{P}$ be a program as in Def. 2 where there is a $p' > 0$ such that $p'(\boldsymbol{x}) \geq p'$ for all $\boldsymbol{x} \in \mathbb{Z}^r$ with $\boldsymbol{a} \bullet \boldsymbol{x} > b$. Then $\mathcal{P}$ is PAST and its expected runtime is at most $\frac{1}{p'}$, i.e., $rt^{\mathcal{P}}_{\boldsymbol{x}_0} \leq \frac{1}{p'}$ if $\boldsymbol{a} \bullet \boldsymbol{x}_0 > b$, and $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = 0$ if $\boldsymbol{a} \bullet \boldsymbol{x}_0 \leq b$.*

Proof. The expected runtime transformer $\mathcal{L}^{\mathcal{P}}$ is continuous (and thus, monotonic) by Thm. 9. Hence, by induction on j one can show that $f \geq \mathcal{L}^{\mathcal{P}}(f)$ implies $f \geq (\mathcal{L}^{\mathcal{P}})^j(\mathfrak{o})$ for any function $f : \mathbb{Z}^r \to \overline{\mathbb{R}_{\geq 0}}$ and any $j \in \mathbb{N}$. So $f \geq \mathcal{L}^{\mathcal{P}}(f)$ implies $f \geq \sup\{\mathfrak{o}, \mathcal{L}^{\mathcal{P}}(\mathfrak{o}), (\mathcal{L}^{\mathcal{P}})^2(\mathfrak{o}), \ldots\} = \operatorname{lfp}(\mathcal{L}^{\mathcal{P}})$. By Thm. 9, this means that $f(\boldsymbol{x}_0) \geq \operatorname{lfp}(\mathcal{L}^{\mathcal{P}})(\boldsymbol{x}_0) = rt^{\mathcal{P}}_{\boldsymbol{x}_0}$ for all $\boldsymbol{x}_0 \in \mathbb{Z}^r$.

Hence, to prove Thm. 10, it suffices to show $f \geq \mathcal{L}^{\mathcal{P}}(f)$ for the function $f : \mathbb{Z}^r \to \overline{\mathbb{R}_{\geq 0}}$ with $f(\boldsymbol{x}) = \frac{1}{p'}$ if $\boldsymbol{a} \bullet \boldsymbol{x} > b$ and $f(\boldsymbol{x}) = 0$ if $\boldsymbol{a} \bullet \boldsymbol{x} \leq b$.

For $\boldsymbol{x}$ with $\boldsymbol{a} \bullet \boldsymbol{x} \leq b$, we have $\mathcal{L}^{\mathcal{P}}(f)(\boldsymbol{x}) = f(\boldsymbol{x})$. If $\boldsymbol{a} \bullet \boldsymbol{x} > b$, then we get

$$\begin{aligned}
\mathcal{L}^{\mathcal{P}}(f)(\boldsymbol{x}) &= \sum_{1 \leq j \leq n} p_{\boldsymbol{c}_j}(\boldsymbol{x}) \cdot f(\boldsymbol{x} + \boldsymbol{c}_j) + p'(\boldsymbol{x}) \cdot f(\boldsymbol{d}) + 1 \\
&\leq \sum_{1 \leq j \leq n} p_{\boldsymbol{c}_j}(\boldsymbol{x}) \cdot \tfrac{1}{p'} + p'(\boldsymbol{x}) \cdot 0 + 1 \\
&= \tfrac{1}{p'} \cdot \sum_{1 \leq j \leq n} p_{\boldsymbol{c}_j}(\boldsymbol{x}) + 1 \\
&= \tfrac{1 - p'(\boldsymbol{x})}{p'} + 1 \quad \leq \quad \tfrac{1-p'}{p'} + 1 \quad = \quad \tfrac{1}{p'} \quad = \quad f(\boldsymbol{x}) \qquad \square
\end{aligned}$$

D Proofs for Sect. 4

In this section we present the proofs of Sect. 4. It is divided into three subsections in which we will give the proofs for the respective subsections of Sect. 4.

D.1 Proofs for Sect. 4.1

To prove Thm. 15, we need an auxiliary lemma.

Lemma 60 (Connections between $\mathcal{P}$ and $\mathcal{P}^{rdw}$). *Let $\mathcal{P}$ be a CP program as in Def. 2 and let $rdw_{\mathcal{P}}^{\omega}$ be the function which applies $rdw_{\mathcal{P}}$ componentwise to runs. Then we have:*

(a) $T^{\mathcal{P}^{rdw}} \circ rdw_{\mathcal{P}}^{\omega} = T^{\mathcal{P}}$
(b) *Let $\boldsymbol{x}_0 \in \mathbb{Z}^r$. Then for any prefix run $\langle y_0, \dots, y_j \rangle \in \mathbb{Z}^{j+1}$ we have:*

$$\mathbb{P}^{\mathcal{P}^{rdw}}_{rdw_{\mathcal{P}}(\boldsymbol{x}_0)}(Cyl^{\mathbb{Z}}(\langle y_0, \dots, y_j \rangle)) = \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}((rdw_{\mathcal{P}}^{\omega})^{-1}(Cyl^{\mathbb{Z}}(\langle y_0, \dots, y_j \rangle))).$$

Here, for any $M \subseteq \mathbb{Z}^\omega$ we have $(rdw_{\mathcal{P}}^{\omega})^{-1}(M) = \{\pi \in (\mathbb{Z}^r)^\omega \mid rdw_{\mathcal{P}}^{\omega}(\pi) \in M\}$.

Proof. (a) Let $\langle \boldsymbol{z}_0, \boldsymbol{z_1}, \dots \rangle \in (\mathbb{Z}^r)^\omega$ such that $T^{\mathcal{P}}(\langle \boldsymbol{z}_0, \boldsymbol{z_1}, \dots \rangle) = j \in \overline{\mathbb{N}}$. So if $j \in \mathbb{N}$, then $rdw_{\mathcal{P}}(\boldsymbol{z}_0), \dots, rdw_{\mathcal{P}}(\boldsymbol{z}_{j-1}) > 0$ and $rdw_{\mathcal{P}}(\boldsymbol{z}_j) \leq 0$. Similarly, if $j = \infty$, then $rdw_{\mathcal{P}}(\boldsymbol{z}_j) > 0$ for every $j \in \mathbb{N}$. So in both cases, we have

$$\begin{aligned} j \;=\; T^{\mathcal{P}^{rdw}}(\langle rdw_{\mathcal{P}}(\boldsymbol{z}_0), rdw_{\mathcal{P}}(\boldsymbol{z}_1), \dots \rangle) &= T^{\mathcal{P}^{rdw}}(rdw_{\mathcal{P}}^{\omega}(\langle \boldsymbol{z}_0, \boldsymbol{z_1}, \dots \rangle)) \\ &= (T^{\mathcal{P}^{rdw}} \circ rdw_{\mathcal{P}}^{\omega})(\langle \boldsymbol{z}_0, \boldsymbol{z_1}, \dots \rangle). \end{aligned}$$

(b) First note that for any prefix run $\langle y_0, \dots, y_j \rangle \in \mathbb{Z}^{j+1}$, we have

$$(rdw_{\mathcal{P}}^{\omega})^{-1}(Cyl^{\mathbb{Z}}(\langle y_0, \dots, y_j \rangle)) \;=\; \biguplus_{\substack{\boldsymbol{z_0}, \dots, \boldsymbol{z_j} \in \mathbb{Z}^r \text{ such that} \\ rdw_{\mathcal{P}}(\boldsymbol{z}_0) = y_0, \dots, rdw_{\mathcal{P}}(\boldsymbol{z}_j) = y_j}} Cyl^{\mathbb{Z}^r}(\langle \boldsymbol{z}_0, \dots, \boldsymbol{z}_j \rangle). \tag{13}$$

As usual, "$\uplus$" denotes the disjoint union, i.e., we have $Cyl^{\mathbb{Z}^r}(\pi) \cap Cyl^{\mathbb{Z}^r}(\pi')$ $= \varnothing$ for prefix runs $\pi \neq \pi'$ of the same length. Note that both sides of the equality (13) can be empty, i.e., there might not be any $\boldsymbol{z}_u$ with $rdw_{\mathcal{P}}(\boldsymbol{z}_u) = y_u$ for some $1 \leq u \leq j$. For $x_0 = rdw_{\mathcal{P}}(\boldsymbol{x}_0)$, we prove that

$$\mathbb{P}^{\mathcal{P}^{rdw}}_{x_0}(Cyl^{\mathbb{Z}}(\langle y_0, \dots, y_j \rangle) \;=\; \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(\biguplus_{\substack{\boldsymbol{z}_0, \dots, \boldsymbol{z}_j \in \mathbb{Z}^r \text{ such that} \\ rdw_{\mathcal{P}}(\boldsymbol{z}_0) = y_0, \dots, rdw_{\mathcal{P}}(\boldsymbol{z}_j) = y_j}} Cyl^{\mathbb{Z}^r}(\langle \boldsymbol{z}_0, \dots, \boldsymbol{z}_j \rangle) \right).$$

The result then follows by (13). For the left-hand side we get $\mathbb{P}^{\mathcal{P}^{rdw}}_{x_0}(Cyl^{\mathbb{Z}}(\langle y_0, \dots, y_j \rangle) = 0$ if $y_0 \neq x_0$ and otherwise, we have

$$\begin{aligned} \mathbb{P}^{\mathcal{P}^{rdw}}_{x_0}(Cyl^{\mathbb{Z}}(\langle y_0, \dots, y_j \rangle) &= \prod_{1 \leq u \leq j} (p^{rdw}_{y_u - y_{u-1}} + \delta_{y_u, rdw_{\mathcal{P}}(\boldsymbol{d})} \cdot p') \\ &= \prod_{1 \leq u \leq j} \left(\sum_{1 \leq v \leq n,\; \boldsymbol{a} \bullet \boldsymbol{c}_v = y_u - y_{u-1}} p_{\boldsymbol{c_t}} + \delta_{y_u, rdw_{\mathcal{P}}(\boldsymbol{d})} \cdot p' \right). \end{aligned}$$

For the right-hand side recall that $rdw_{\mathcal{P}}(\boldsymbol{x}_0) = x_0$ and that we only regard tuples $\boldsymbol{z}_0$ where $rdw_{\mathcal{P}}(\boldsymbol{z}_0) = y_0$. So if $y_0 \neq x_0$, then all of these tuples $\boldsymbol{z}_0$ are different from $\boldsymbol{x}_0$. Hence, then the right-hand side is also 0. Otherwise, we have the following, where $d_{\mathcal{P}} = rdw_{\mathcal{P}}(\boldsymbol{d})$:

$$
\begin{aligned}
&\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(\biguplus_{\boldsymbol{z}_0,\dots,\boldsymbol{z}_j\in\mathbb{Z}^r \text{ such that } rdw_{\mathcal{P}}(\boldsymbol{z}_0)=y_0,\dots,rdw_{\mathcal{P}}(\boldsymbol{z}_j)=y_j} Cyl^{\mathbb{Z}^r}(\langle \boldsymbol{z}_0,\dots,\boldsymbol{z}_j\rangle)\right)\\
&= \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(\biguplus_{\boldsymbol{z}_1,\dots,\boldsymbol{z}_j\in\mathbb{Z}^r \text{ such that } rdw_{\mathcal{P}}(\boldsymbol{z}_1)=y_1,\dots,rdw_{\mathcal{P}}(\boldsymbol{z}_j)=y_j} Cyl^{\mathbb{Z}^r}(\langle \boldsymbol{x}_0,\boldsymbol{z}_1,\dots,\boldsymbol{z}_j\rangle)\right)\\
&= \sum_{\boldsymbol{z}_1,\dots,\boldsymbol{z}_j\in\mathbb{Z}^r \text{ such that } rdw_{\mathcal{P}}(\boldsymbol{z}_1)=y_1,\dots,rdw_{\mathcal{P}}(\boldsymbol{z}_j)=y_j} \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(Cyl^{\mathbb{Z}^r}(\langle \boldsymbol{x}_0,\boldsymbol{z}_1,\dots,\boldsymbol{z}_j\rangle)\right)\\
&= \sum_{\substack{\boldsymbol{z}_1,\dots,\boldsymbol{z}_j\in\mathbb{Z}^r \text{ such that}\\ rdw_{\mathcal{P}}(\boldsymbol{z}_1)=y_1,\dots,rdw_{\mathcal{P}}(\boldsymbol{z}_j)=y_j}} (p_{\boldsymbol{z}_1-\boldsymbol{x}_0} + \delta_{\boldsymbol{z}_1,\boldsymbol{d}}\cdot p')\cdot \prod_{2\le u\le j}(p_{\boldsymbol{z}_u-\boldsymbol{z}_{u-1}} + \delta_{\boldsymbol{z}_u,\boldsymbol{d}}\cdot p')\\
&= \sum_{\substack{\boldsymbol{c}_{v_1},\dots,\boldsymbol{c}_{v_j}\in\{\boldsymbol{c}_1,\dots,\boldsymbol{c}_n\} \text{ such that } rdw_{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{c}_{v_1})=y_1,\\ rdw_{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{c}_{v_1}+\boldsymbol{c}_{v_2})=y_2,\dots,rdw_{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{c}_{v_1}+\dots+\boldsymbol{c}_{v_j})=y_j}} (p_{\boldsymbol{c}_{v_1}} + \delta_{y_1,d_{\mathcal{P}}}\cdot p')\ \cdot\ \dots\ \cdot\ (p_{\boldsymbol{c}_{v_j}} + \delta_{y_j,d_{\mathcal{P}}}\cdot p')\\
&\overset{(\dagger)}{=} \sum_{\boldsymbol{c}_{v_1},\dots,\boldsymbol{c}_{v_j}\in\{\boldsymbol{c}_1,\dots,\boldsymbol{c}_n\} \text{ such that } y_1-y_0=\boldsymbol{a}\bullet\boldsymbol{c}_{v_1},\dots,y_j-y_{j-1}=\boldsymbol{a}\bullet\boldsymbol{c}_{v_j}} (p_{\boldsymbol{c}_{v_1}} + \delta_{y_1,d_{\mathcal{P}}}\cdot p')\ \cdot\ \dots\ \cdot\ (p_{\boldsymbol{c}_{v_j}} + \delta_{y_j,d_{\mathcal{P}}}\cdot p')\\
&= \Big(\sum_{\boldsymbol{c}\in\{\boldsymbol{c}_1,\dots,\boldsymbol{c}_n\} \text{ such that } y_1-y_0=\boldsymbol{a}\bullet\boldsymbol{c}} p_{\boldsymbol{c}} + \delta_{y_1,d_{\mathcal{P}}}\cdot p'\Big)\ \cdot\ \dots\ \cdot\ \Big(\sum_{\boldsymbol{c}\in\{\boldsymbol{c}_1,\dots,\boldsymbol{c}_n\} \text{ such that } y_j-y_{j-1}=\boldsymbol{a}\bullet\boldsymbol{c}} p_{\boldsymbol{c}} + \delta_{y_j,d_{\mathcal{P}}}\cdot p'\Big)\\
&= \prod_{1\le u\le j}\left(\sum_{1\le t\le n,\ \boldsymbol{a}\bullet\boldsymbol{c}_t=y_u-y_{u-1}} p_{\boldsymbol{c}_t} + \delta_{y_u,d_{\mathcal{P}}}\cdot p'\right).
\end{aligned}
$$

For Equation (†), note that $rdw_{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{c}_{v_1}) = \boldsymbol{a}\bullet(\boldsymbol{x}_0+\boldsymbol{c}_{v_1}) - b = \boldsymbol{a}\bullet\boldsymbol{x}_0 + \boldsymbol{a}\bullet\boldsymbol{c}_{v_1} - b = rdw_{\mathcal{P}}(\boldsymbol{x}_0) + \boldsymbol{a}\bullet\boldsymbol{c}_{v_1} = y_0 + \boldsymbol{a}\bullet\boldsymbol{c}_{v_1}$. Hence, $rdw_{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{c}_{v_1}) = y_1$ means that $y_1 - y_0 = \boldsymbol{a}\bullet\boldsymbol{c}_{v_1}$. Similarly, $rdw_{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{c}_{v_1}+\boldsymbol{c}_{v_2}) = y_0 + \boldsymbol{a}\bullet\boldsymbol{c}_{v_1} + \boldsymbol{a}\bullet\boldsymbol{c}_{v_2} = y_1 + \boldsymbol{a}\bullet\boldsymbol{c}_{v_2}$. So $rdw_{\mathcal{P}}(\boldsymbol{x}_0+\boldsymbol{c}_{v_1}+\boldsymbol{c}_{v_2}) = y_2$ means that $y_2 - y_1 = \boldsymbol{a}\bullet\boldsymbol{c}_{v_2}$, etc. □

Theorem 15 (Transformation Preserves Termination & Expected Runtime). *Let $\mathcal{P}$ be a CP program as in Def. 2. Then the termination times $T^{\mathcal{P}}$ and $T^{\mathcal{P}^{rdw}}$ are identically distributed w.r.t. $rdw_{\mathcal{P}}$, i.e., for all $\boldsymbol{x}_0 \in \mathbb{Z}^r$ with $x_0 = rdw_{\mathcal{P}}(\boldsymbol{x}_0)$ and all $j \in \overline{\mathbb{N}}$ we have $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}}{=}j) = \mathbb{P}^{\mathcal{P}^{rdw}}_{x_0}(T^{\mathcal{P}^{rdw}}{=}j)$. So in particular, $\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}}{<}\infty) = \mathbb{P}^{\mathcal{P}^{rdw}}_{x_0}(T^{\mathcal{P}^{rdw}}{<}\infty)$ and $rt^{\mathcal{P}}_{\boldsymbol{x}_0} = \mathbb{E}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}}) = \mathbb{E}^{\mathcal{P}^{rdw}}_{x_0}(T^{\mathcal{P}^{rdw}}) = rt^{\mathcal{P}^{rdw}}_{x_0}$. Thus, the expected runtimes of $\mathcal{P}$ on the input $\boldsymbol{x}_0$ and of $\mathcal{P}^{rdw}$ on x_0 coincide.*

Proof. For any $j \in \mathbb{N}$ and any $\boldsymbol{x}_0 \in \mathbb{Z}^r$ we obtain the following.

$$
\begin{aligned}
&\mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}} = j)\\
&= \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}(T^{\mathcal{P}^{rdw}} \circ rdw^{\omega}_{\mathcal{P}} = j) && \text{by Lemma 60 (a)}
\end{aligned}
$$

$$
\begin{aligned}
&= \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left((rdw^{\omega}_{\mathcal{P}})^{-1}\left((T^{\mathcal{P}^{rdw}})^{-1}(\{j\})\right)\right)\\
&= \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left((rdw^{\omega}_{\mathcal{P}})^{-1}\left(\biguplus_{y_0,\dots,y_{j-1}\in\mathbb{Z}_{>0},y_j\in\mathbb{Z}_{\leq 0}} Cyl^{\mathbb{Z}}(\langle y_0,\dots,y_j\rangle)\right)\right)\\
&= \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left(\biguplus_{y_0,\dots,y_{j-1}\in\mathbb{Z}_{>0},y_j\in\mathbb{Z}_{\leq 0}} (rdw^{\omega}_{\mathcal{P}})^{-1}\left(Cyl^{\mathbb{Z}}(\langle y_0,\dots,y_j\rangle)\right)\right)\\
&= \sum_{y_0,\dots,y_{j-1}\in\mathbb{Z}_{>0},y_j\in\mathbb{Z}_{\leq 0}} \mathbb{P}^{\mathcal{P}}_{\boldsymbol{x}_0}\left((rdw^{\omega}_{\mathcal{P}})^{-1}\left(Cyl^{\mathbb{Z}}(\langle y_0,\dots,y_j\rangle)\right)\right)\\
&= \sum_{y_0,\dots,y_{j-1}\in\mathbb{Z}_{>0},y_j\in\mathbb{Z}_{\leq 0}} \mathbb{P}^{\mathcal{P}^{rdw}}_{rdw_{\mathcal{P}}(\boldsymbol{x}_0)}\left(Cyl^{\mathbb{Z}}(\langle y_0,\dots,y_j\rangle)\right) \qquad \text{by Lemma 60 (b)}\\
&= \mathbb{P}^{\mathcal{P}^{rdw}}_{rdw_{\mathcal{P}}(\boldsymbol{x}_0)}\left(\biguplus_{y_0,\dots,y_{j-1}\in\mathbb{Z}_{>0},y_j\in\mathbb{Z}_{\leq 0}} Cyl^{\mathbb{Z}}(\langle y_0,\dots,y_j\rangle)\right)\\
&= \mathbb{P}^{\mathcal{P}^{rdw}}_{rdw_{\mathcal{P}}(\boldsymbol{x}_0)}\left(T^{\mathcal{P}^{rdw}} = j\right).
\end{aligned}
$$

As the above equality holds for every $j \in \mathbb{N}$ it also holds for $j = \infty$. □

D.2 Proofs for Sect. 4.2

For the proof of Thm. 18, we use results on random walks [17,21,33]. We first recapitulate the required notions from probability theory.

Consider a probability space $(\Omega, \mathfrak{F}, \mathbb{P})$ (i.e., for every $A \in \mathfrak{F} \subseteq 2^{\Omega}$, $\mathbb{P}(A)$ is the probability that an event from the set Ω is in the subset A) and a stochastic process $\mathbf{Y} = (Y_j)_{j\in\mathbb{N}}$ where each $Y_j : \Omega \to \mathbb{Z}$ is a random variable. $\mathbf{Y}$ is *independent and identically distributed (i.i.d.)* on $(\Omega, \mathfrak{F}, \mathbb{P})$ if for all $j, j' \in \mathbb{N}$ with $j \neq j'$ and all $y, z \in \mathbb{Z}$:

- Y_j and $Y_{j'}$ are identically distributed, i.e., $\mathbb{P}(Y_j = z) = \mathbb{P}(Y_{j'} = z)$
- Y_j and $Y_{j'}$ are independent, i.e., $\mathbb{P}(Y_j = y, Y_{j'} = z) = \mathbb{P}(Y_j = y) \cdot \mathbb{P}(Y_{j'} = z)$

Here, $\mathbb{P}(Y_j = y, Y_{j'} = z) = \mathbb{P}(Y_j^{-1}(\{y\}) \cap Y_{j'}^{-1}(\{z\}))$ is the probability that an event $\pi \in \Omega$ satisfies both $Y_j(\pi) = y$ and $Y_{j'}(\pi) = z$. So independence means that one random variable does not influence the value of the other.

Now we recapitulate the notion of a random walk created by an i.i.d. stochastic process.

Definition 61 (Random Walk [21]). *Let $\mathbf{Y} = (Y_j)_{j\in\mathbb{N}}$ be an i.i.d. stochastic process for a probability space $(\Omega, \mathfrak{F}, \mathbb{P})$ with $Y_j : \Omega \to \mathbb{Z}$ and let $X_0 : \Omega \to \mathbb{Z}$ be a random variable such that $\mathbb{P}(X_0 = x_0) = 1$ for some $x_0 \in \mathbb{Z}$. The (one-dimensional)* random walk *for $(\Omega, \mathfrak{F}, \mathbb{P})$ induced by $\mathbf{Y}$ with starting point X_0 is the sequence $\mathbf{S} = (S_j)_{j\in\mathbb{N}}$ of random variables*[6] *$S_j = X_0 + \sum_{0\leq u\leq j-1} Y_u$. We denote the random walk $\mathbf{S}$ by $(X_0, \mathbf{Y})$.*

[6] Note that we define $S_j = X_0 + \sum_{0\leq u\leq j-1} Y_u$ instead of $S_j = x_0 + \sum_{0\leq u\leq j-1} Y_u$. In this way, the random variables $X_0, Y_0, Y_1, \dots$ only generate a single random walk that does not depend on x_0. Instead, the different possible initial values x_0 are taken care of by choosing different probability spaces $(\Omega, \mathfrak{F}, \mathbb{P}_{x_0})$ where $\mathbb{P}_{x_0}(X_0 = x_0) = 1$.

Analogous to the termination time for programs from Def. 3, the *hitting time* is the time when the random walk "hits" a certain subset of $\mathbb{Z}$ for the first time.

Definition 62 (Hitting Time). *The* hitting time *for a random walk* $(S_j)_{j\in\mathbb{N}}$ *is the random variable* $T^{hit} : \Omega \to \overline{\mathbb{N}}$ *with* $T^{hit}(\pi) = \inf\{j \in \mathbb{N} \mid S_j(\pi) \leq 0\}$.

If $\mathbf{Y} = (Y_j)_{j\in\mathbb{N}}$ is i.i.d., then $\mathbb{E}(Y_0) = \mathbb{E}(Y_j)$ for all $j \in \mathbb{N}$. Hence, we define $\mu = \mathbb{E}(Y_0)$ to be the *drift*, i.e., the expected change in each step of the random walk. For such random walks, a result similar to Thm. 18 is already known.

Lemma 63 (Drift and Hitting Time [33, Thm. 17.1, Prop. 18.1]). *Let* $\mathbf{Y}$ *be i.i.d. for a probability space* $(\Omega, \mathfrak{F}, \mathbb{P})$ *and let* $(X_0, \mathbf{Y})$ *be a random walk for* $(\Omega, \mathfrak{F}, \mathbb{P})$ *such that* $\mu = \mathbb{E}(Y_0) < \infty$ *(note that the drift* μ *does not depend on* X_0*). Let* T^{hit} *be the hitting time for* $(X_0, \mathbf{Y})$*. Then we have:*

- *If* $\mu > 0$*, then* $\mathbb{P}(T^{hit} = \infty) > 0$*.*
- *If* $\mu = 0$ *and* $\mathbb{P}(Y_0 = 0) \neq 1$*, then* $\mathbb{P}(T^{hit} = \infty) = 0$ *but* $\mathbb{E}(T^{hit}) = \infty$*.*
- *If* $\mu < 0$*, then* $\mathbb{E}(T^{hit}) < \infty$*.*

In order to use Lemma 63 to prove Thm. 18, our aim is to represent the stochastic process $\mathbf{X}^{\mathbb{Z}}$ from Def. 46 (for $r = 1$) as a random walk $\mathbf{X}^{\mathbb{Z}} = (X_0^{\mathbb{Z}}, \mathbf{Y}^{\mathbb{Z}})$ for a suitable stochastic process $\mathbf{Y}^{\mathbb{Z}}$.

To this end, we take the stochastic process $\mathbf{Y}^{\mathbb{Z}} = (Y_j^{\mathbb{Z}})_{j\in\mathbb{N}}$ with $Y_j^{\mathbb{Z}} = (X_{j+1}^{\mathbb{Z}} - X_j^{\mathbb{Z}})$ for all $j \in \mathbb{N}$, i.e., $Y_j^{\mathbb{Z}}$ is the change of the program variable in the $(j+1)$-th loop iteration. Then $\mathbf{X}^{\mathbb{Z}}$ can be obtained as the random walk $(X_0^{\mathbb{Z}}, \mathbf{Y}^{\mathbb{Z}})$, since $\mathbb{P}_{x_0}^{\mathcal{P}}(X_0^{\mathbb{Z}} = x_0) = 1$ and $X_j^{\mathbb{Z}} = X_0^{\mathbb{Z}} + \sum_{0\leq u\leq j-1}(X_{u+1}^{\mathbb{Z}} - X_u^{\mathbb{Z}}) = X_0^{\mathbb{Z}} + \sum_{0\leq u\leq j-1} Y_u^{\mathbb{Z}}$ for all $j \in \mathbb{N}$.

Unfortunately, $\mathbf{Y}^{\mathbb{Z}}$ is not i.i.d. for the probability measure $\mathbb{P}_{x_0}^{\mathcal{P}}$, because the probability that $Y_j^{\mathbb{Z}} = 0$ (i.e., that $X_{j+1}^{\mathbb{Z}} = X_j^{\mathbb{Z}}$ holds) depends on j. More precisely, the probability for $X_{j+1}^{\mathbb{Z}} = X_j^{\mathbb{Z}}$ is p_0 plus the probability that the program has already reached a value $x \leq 0$ (i.e., that the program's termination time is at most j). The reason is that according to the probability measure $\mathbb{P}_{x_0}^{\mathcal{P}}$, the value of x remains unchanged as soon as $x \leq 0$. Thus, we obtain $\mathbb{P}_{x_0}^{\mathcal{P}}(Y_j^{\mathbb{Z}} = 0) = p_0 + \mathbb{P}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}} \leq j)$, where $\mathbb{P}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}} \leq j)$ clearly depends on j. Therefore, we now introduce a new adapted probability measure $\mathbf{P}_{x_0}^{\mathcal{P}}$ such that $\mathbf{Y}^{\mathbb{Z}}$ is i.i.d. on the probability space $(\mathbb{Z}^{\omega}, \mathfrak{F}^{\mathbb{Z}}, \mathbf{P}_{x_0}^{\mathcal{P}})$ and at the same time, $\mathbf{E}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}}) = \mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right)$, where $\mathbf{E}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}})$ denotes the expected value of the termination time $T^{\mathcal{P}}$ under the probability measure $\mathbf{P}_{x_0}^{\mathcal{P}}$. In the following definition, $q_{x_0}^{\mathcal{P}}$ corresponds to the function $pr_{x_0}^{\mathcal{P}}$ from Def. 42 that maps any prefix run to its probability if x_0 is the initial value of the program variable. When defining $pr_{x_0}^{\mathcal{P}}$, the probability for a prefix run $\langle z_0, \ldots, z_{j-1}, z_j\rangle$ where $z_{j-1} \leq 0$ and $z_{j-1} \neq z_j$ was 0. In contrast, for $q_{x_0}^{\mathcal{P}}$ we continue to execute the program also if $x \leq 0$. This corresponds to a variant of the program where the loop condition $x > 0$ is replaced by *true*.

Definition 64 (Probability Measure $\mathbf{P}_{x_0}^{\mathcal{P}}$). *For any random walk program $\mathcal{P}$ as in Def. 12 without direct termination, any $x_0 \in \mathbb{Z}$, and any prefix run $\langle z_0, z_1, \dots, z_j \rangle$, let $q_{x_0}^{\mathcal{P}}(\langle z_0 \rangle) = \delta_{x_0,z_0}$ and if $j \geq 1$, we define:*

$$q_{x_0}^{\mathcal{P}}(\langle z_0, \dots, z_j \rangle) = q_{x_0}^{\mathcal{P}}(\langle z_0, \dots, z_{j-1} \rangle) \cdot p_{z_j - z_{j-1}}$$

$\mathbf{P}_{x_0}^{\mathcal{P}}$ is the probability measure with $\mathbf{P}_{x_0}^{\mathcal{P}}(Cyl^{\mathbb{Z}}(\pi)) = q_{x_0}^{\mathcal{P}}(\pi)$ for any prefix run π.

Example 65 (Adapted Probability Measure for $\mathcal{P}_{race}^{rdw}$). Consider runs that start with 1, -2, and -6. Here, we have $Y_0^{\mathbb{Z}}(\langle 1, -2, -6, \dots \rangle) = (-2) - 1 = -3$ and $Y_1^{\mathbb{Z}}(\langle 1, -2, -6, \dots \rangle) = (-6) - (-2) = -4$. For $\mathcal{P}_{race}^{rdw}$ of Ex. 14, when using the probability measure $\mathbb{P}_1^{\mathcal{P}_{race}^{rdw}}$ from Def. 44, we obtain $\mathbb{P}_1^{\mathcal{P}_{race}^{rdw}}(Cyl^{\mathbb{Z}}(\langle 1, -2, -6 \rangle)) = pr_1^{\mathcal{P}_{race}^{rdw}}(\langle 1, -2, -6 \rangle) = p_{-3}^{rdw} \cdot p_{-4}^{rdw} \cdot \delta_{-2,-6} = 0$, since the value of x should not change anymore after reaching the non-positive value -2. In contrast, the adapted probability measure $\mathbf{P}_1^{\mathcal{P}_{race}^{rdw}}$ from Def. 64 yields $\mathbf{P}_1^{\mathcal{P}_{race}^{rdw}}(Cyl^{\mathbb{Z}}(\langle 1, -2, -6 \rangle)) = q_1^{\mathcal{P}_{race}^{rdw}}(Cyl^{\mathbb{Z}}(\langle 1, -2, -6 \rangle)) = p_{-3}^{rdw} \cdot p_{-4}^{rdw} = \frac{1}{22} \cdot \frac{1}{22} = \frac{1}{484}$.

For the termination time $T^{\mathcal{P}}$ one only regards the time that it takes until the program variable x is non-positive for the first time. Thus, it does not matter whether x is kept unchanged afterwards (as in the probability measure $\mathbb{P}_{x_0}^{\mathcal{P}}$) or whether the loop body is executed further afterwards (as in $\mathbf{P}_{x_0}^{\mathcal{P}}$). So the expected runtime is the same, no matter whether one uses $\mathbb{E}_{x_0}^{\mathcal{P}}$ or $\mathbf{E}_{x_0}^{\mathcal{P}}$.

Lemma 66 ($T^{\mathcal{P}}$ is Identically Distributed Under $\mathbb{P}_{x_0}^{\mathcal{P}}$ and $\mathbf{P}_{x_0}^{\mathcal{P}}$). *For any random walk program $\mathcal{P}$ without direct termination, any $x_0 \in \mathbb{Z}$, and any $j \in \overline{\mathbb{N}}$, we have $\mathbb{P}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}} = j\right) = \mathbf{P}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}} = j\right)$. Thus, $\mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right) = \mathbf{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right)$.*

Proof. First of all, by the definition of $T^{\mathcal{P}}$, for any $j \in \mathbb{N}$ we have

$$(T^{\mathcal{P}})^{-1}(\{j\}) = \biguplus_{\pi = \langle z_0, \dots, z_j \rangle \in \mathbb{Z}_{>0}^j \times \mathbb{Z}_{\leq 0}} Cyl^{\mathbb{Z}}(\pi). \tag{14}$$

First, we consider $x_0 \leq 0$. Then any cylinder set with positive probability w.r.t. $\mathbb{P}_{x_0}^{\mathcal{P}}$ resp. $\mathbf{P}_{x_0}^{\mathcal{P}}$ has the form $Cyl^{\mathbb{Z}}(\pi)$ where π starts with $x_0 \leq 0$. But for any run $\tau \in Cyl^{\mathbb{Z}}(\pi)$ we have $T^{\mathcal{P}}(\tau) = 0$. Therefore, we conclude $\mathbb{P}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}} = 0) = 1 = \mathbf{P}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}} = 0)$.

We now show that for $x_0 > 0$

$$\mathbb{P}_{x_0}^{\mathcal{P}}\left(Cyl^{\mathbb{Z}}(\pi)\right) = \mathbf{P}_{x_0}^{\mathcal{P}}\left(Cyl^{\mathbb{Z}}(\pi)\right) \quad \text{for any } \pi = \langle z_0, \dots, z_j \rangle \in \mathbb{Z}_{>0}^j \times \mathbb{Z}_{\leq 0}. \tag{15}$$

The reason is that we have:

$$\begin{aligned}
\mathbb{P}_{x_0}^{\mathcal{P}}(Cyl^{\mathbb{Z}}(\pi)) &= pr_{x_0}^{\mathcal{P}}(\pi) && \text{by Def. 44} \\
&= \delta_{x_0,z_0} \cdot \prod_{0 \leq u \leq j-1} p_{z_{u+1} - z_u} && \text{by Def. 42 as } z_0, \dots, z_{j-1} > 0 \\
&= q_{x_0}^{\mathcal{P}}(\pi) \\
&= \mathbf{P}_{x_0}^{\mathcal{P}}(Cyl^{\mathbb{Z}}(\pi)) && \text{by Def. 64}
\end{aligned}$$

Therefore, for all $j \in \mathbb{N}$ we obtain:

$$
\begin{aligned}
& \mathbb{P}^{\mathcal{P}}_{x_0}\left(T^{\mathcal{P}} = j\right) \\
={}& \mathbb{P}^{\mathcal{P}}_{x_0}((T^{\mathcal{P}})^{-1}(\{j\})) \\
={}& \mathbb{P}^{\mathcal{P}}_{x_0}\left(\biguplus_{\pi=\langle z_0,\dots,z_j\rangle\in\mathbb{Z}^j_{>0}\times\mathbb{Z}_{\leq 0}} Cyl^{\mathbb{Z}}(\pi)\right) && \text{by (14)} \\
={}& \sum_{\pi=\langle z_0,\dots,z_j\rangle\in\mathbb{Z}^j_{>0}\times\mathbb{Z}_{\leq 0}} \mathbb{P}^{\mathcal{P}}_{x_0}\left(Cyl^{\mathbb{Z}}(\pi)\right) && \text{by additivity of prob. measures} \\
={}& \sum_{\pi=\langle z_0,\dots,z_j\rangle\in\mathbb{Z}^j_{>0}\times\mathbb{Z}_{\leq 0}} \mathbf{P}^{\mathcal{P}}_{x_0}\left(Cyl^{\mathbb{Z}}(\pi)\right) && \text{by (15)} \\
={}& \mathbf{P}^{\mathcal{P}}_{x_0}\left(\biguplus_{\pi=\langle z_0,\dots,z_j\rangle\in\mathbb{Z}^j_{>0}\times\mathbb{Z}_{\leq 0}} Cyl^{\mathbb{Z}}(\pi)\right) && \text{by additivity of prob. measures} \\
={}& \mathbf{P}^{\mathcal{P}}_{x_0}\left(T^{\mathcal{P}} = j\right) && \text{by (14)}
\end{aligned}
$$

Finally, $\mathbb{P}^{\mathcal{P}}_{x_0}\left(T^{\mathcal{P}} = \infty\right) = 1 - \sum_{j\in\mathbb{N}} \mathbb{P}^{\mathcal{P}}_{x_0}\left(T^{\mathcal{P}} = j\right) = 1 - \sum_{j\in\mathbb{N}} \mathbf{P}^{\mathcal{P}}_{x_0}\left(T^{\mathcal{P}} = j\right) = \mathbf{P}^{\mathcal{P}}_{x_0}\left(T^{\mathcal{P}} = \infty\right)$. □

Now we show that the process $\mathbf{Y}^{\mathbb{Z}}$ with $Y^{\mathbb{Z}}_j = X^{\mathbb{Z}}_{j+1} - X^{\mathbb{Z}}_j$ is i.i.d. w.r.t. the probability measure $\mathbf{P}^{\mathcal{P}}_{x_0}$ and thus, $(X^{\mathbb{Z}}_0, \mathbf{Y}^{\mathbb{Z}})$ is a random walk for $(\mathbb{Z}^\omega, \mathfrak{F}^{\mathbb{Z}}, \mathbf{P}^{\mathcal{P}}_{x_0})$. So the expected value of $Y^{\mathbb{Z}}_j$ under $\mathbf{P}^{\mathcal{P}}_{x_0}$, is the same for all j. In fact, this expected value is the drift $\mu_{\mathcal{P}}$ of the program, irrespective of the start value x_0.

Lemma 67 (Y is i.i.d. and its Expected Value is the Drift of the Program). *Let $\mathbf{X}^{\mathbb{Z}}$ be the stochastic process as in Def. 46. We define the process $\mathbf{Y}^{\mathbb{Z}} = (Y^{\mathbb{Z}}_j)_{j\in\mathbb{N}}$ by $Y^{\mathbb{Z}}_j = X^{\mathbb{Z}}_{j+1} - X^{\mathbb{Z}}_j$ for all $j \in \mathbb{N}$. Then for any random walk program $\mathcal{P}$ without direct termination and any $x_0 \in \mathbb{Z}$, $\mathbf{Y}^{\mathbb{Z}}$ is i.i.d. w.r.t. $(\mathbb{Z}^\omega, \mathfrak{F}^{\mathbb{Z}}, \mathbf{P}^{\mathcal{P}}_{x_0})$ and thus, $(X^{\mathbb{Z}}_0, \mathbf{Y}^{\mathbb{Z}})$ is a random walk for this probability space. Furthermore, for any $x_0 \in \mathbb{Z}$ and any $j \in \mathbb{N}$, we have $\mathbf{E}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_j) = \mu_{\mathcal{P}}$.*

Proof. We first show that the $Y^{\mathbb{Z}}_j$ are identically distributed. More precisely, we prove that for all $u, x_0 \in \mathbb{Z}$ and all $j \in \mathbb{N}$ we have $\mathbf{P}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_j = u) = p_u$. Similar to our handling of multivariate programs in App. B, for any random walk program $\mathcal{P}$ as in Def. 12 we define $p_v = 0$ for $v > m$ or $v < -k$.

$$
\begin{aligned}
& \mathbf{P}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_j = u) \\
={}& \mathbf{P}^{\mathcal{P}}_{x_0}(X^{\mathbb{Z}}_{j+1} - X^{\mathbb{Z}}_j = u) \\
={}& \mathbf{P}^{\mathcal{P}}_{x_0}\left(\{\langle z_0,\dots\rangle \in \mathbb{Z}^\omega \mid z_{j+1} - z_j = u\}\right) \\
={}& \mathbf{P}^{\mathcal{P}}_{x_0}\left(\biguplus_{\pi=\langle z_0,\dots,z_{j+1}\rangle\in\mathbb{Z}^{j+2}, z_{j+1}-z_j=u} Cyl^{\mathbb{Z}}(\pi)\right)
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{\pi=\langle z_0,\dots z_{j+1}\rangle\in\mathbb{Z}^{j+2}, z_{j+1}-z_j=u} \mathbf{P}^{\mathcal{P}}_{x_0}\left(Cyl^{\mathbb{Z}}(\pi)\right) && \text{by additivity of prob. measures}\\
&= \sum_{\pi=\langle z_0,\dots,z_{j+1}\rangle\in\mathbb{Z}^{j+2}, z_{j+1}-z_j=u} q^{\mathcal{P}}_{x_0}(\pi) && \text{by Def. 64}\\
&= \sum_{\pi=\langle z_0,z_1,\dots,z_{j+1}\rangle\in\mathbb{Z}^{j+2}, z_0=x_0, z_{j+1}-z_j=u} \prod_{0\le v\le j} p_{z_{v+1}-z_v} && \text{by Def. 64}\\
&= \sum_{\pi=\langle z_0,z_1,\dots,z_j\rangle\in\mathbb{Z}^{j+1}, z_0=x_0} \left(p_u \cdot \prod_{0\le v\le j-1} p_{z_{v+1}-z_v}\right)\\
&= \left(\sum_{\pi=\langle z_0,z_1,\dots,z_j\rangle\in\mathbb{Z}^{j+1}, z_0=x_0} \prod_{0\le v\le j-1} p_{z_{v+1}-z_v}\right) \cdot p_u\\
&= \left(\sum_{v_1,\dots,v_j\in\mathbb{Z}} p_{v_1}\cdot\ldots\cdot p_{v_j}\right)\cdot p_u\\
&= \left(\sum_{v\in\mathbb{Z}} p_v\right)^j \cdot p_u\\
&= 1^j\cdot p_u = p_u. && \text{as } \mathcal{P} \text{ does not have direct termination}
\end{aligned}
$$

As p_u is independent of j, the $Y^{\mathbb{Z}}_j$ are identically distributed. Furthermore, the expected value of $Y^{\mathbb{Z}}_j$ under $\mathbf{P}^{\mathcal{P}}_{x_0}$ is

$$
\mathbf{E}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_j) = \sum_{-k\le u\le m} u\cdot p_u = \mu_{\mathcal{P}},
$$

which is the drift of the program.

It remains to show the independence of the random variables. Let $j \neq j' \in \mathbb{N}$ and w.l.o.g. assume $j' > j$.

$$
\begin{aligned}
&\mathbf{P}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_j = u, Y^{\mathbb{Z}}_{j'} = u')\\
&= \mathbf{P}^{\mathcal{P}}_{x_0}(X^{\mathbb{Z}}_{j+1} - X^{\mathbb{Z}}_j = u, X^{\mathbb{Z}}_{j'+1} - X^{\mathbb{Z}}_{j'} = u')\\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\{\langle z_0,\dots\rangle\in\mathbb{Z}^\omega \mid z_{j+1}-z_j = u, z_{j'+1}-z_{j'} = u'\}\\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(\biguplus_{\pi=\left\langle z_0,\dots,z_{j'+1}\right\rangle\in\mathbb{Z}^{j'+2}, z_{j+1}-z_j=u, z_{j'+1}-z_{j'}=u'} Cyl^{\mathbb{Z}}(\pi)\right)\\
&= \sum_{\pi=\left\langle z_0,\dots,z_{j'+1}\right\rangle\in\mathbb{Z}^{j'+2}, z_{j+1}-z_j=u, z_{j'+1}-z_{j'}=u'} \mathbf{P}^{\mathcal{P}}_{x_0}\left(Cyl^{\mathbb{Z}}(\pi)\right)\\
&= \sum_{\pi=\left\langle z_0,\dots,z_{j'+1}\right\rangle\in\mathbb{Z}^{j'+2}, z_{j+1}-z_j=u, z_{j'+1}-z_{j'}=u'} q^{\mathcal{P}}_{x_0}(\pi)\\
&= \sum_{\pi=\left\langle z_0,z_1,\dots z_{j'+1}\right\rangle\in\mathbb{Z}^{j'+2}, z_0=x_0, z_{j+1}-z_j=u, z_{j'+1}-z_{j'}=u'} \prod_{0\le v\le j'} p_{z_{v+1}-z_v}
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{\substack{\pi=\langle z_0,z_1,\dots z_{j'+1}\rangle\in\mathbb{Z}^{j'+2},\\ z_0=x_0, z_{j+1}-z_j=u,\\ z_{j'+1}-z_{j'}=u'}} p_u \cdot \left(\prod_{0\le v\le j-1} p_{z_{v+1}-z_v}\right) \cdot p_{u'} \cdot \left(\prod_{j+1\le v\le j'-1} p_{z_{v+1}-z_v}\right)\\
&= \left(\sum_{\pi=\langle z_0,z_1,\dots z_{j'}\rangle\in\mathbb{Z}^{j'+1}, z_0=x_0, z_{j+1}-z_j=u} \quad \prod_{0\le v\le j'-1,\, v\neq j} p_{z_{v+1}-z_v}\right) \cdot p_u \cdot p_{u'}\\
&= \left(\sum_{v_1,\dots,v_{j'-1}\in\mathbb{Z}} p_{v_1}\cdot\ldots\cdot p_{v_{j'-1}}\right) \cdot p_u \cdot p_{u'}\\
&= \left(\sum_{v\in\mathbb{Z}} p_v\right)^{j'-1} \cdot p_u \cdot p_{u'}\\
&= 1^{j'-1} \cdot p_u \cdot p_{u'} = p_u \cdot p_{u'}\\
&= \mathbf{P}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_j = u) \cdot \mathbf{P}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_{j'} = u'). \qquad \square
\end{aligned}
$$

Now we can prove Thm. 18 based on the results of Lemma 63 for random walks.

Theorem 18 (Decision Procedure for (P)AST of Random Walk Programs). *Let $\mathcal{P}$ be a non-trivial random walk program without direct termination.*

- *If $\mu_{\mathcal{P}} > 0$, then the program is* not *AST.*
- *If $\mu_{\mathcal{P}} = 0$, then the program is AST but* not *PAST.*
- *If $\mu_{\mathcal{P}} < 0$, then the program is PAST.*

Proof. Due to Lemma 67, $\mathbf{Y}^{\mathbb{Z}}$ is i.i.d. w.r.t. $(\mathbb{Z}^\omega, \mathfrak{F}^{\mathbb{Z}}, \mathbf{P}^{\mathcal{P}}_{x_0})$ and thus, $\mathbf{S}^{\mathbb{Z}} = (X^{\mathbb{Z}}_0, \mathbf{Y}^{\mathbb{Z}})$ is a random walk w.r.t. this probability space for any $x_0 \in \mathbb{Z}$. By Def. 61 we have $S^{\mathbb{Z}}_j = X^{\mathbb{Z}}_0 + \sum_{0\le u\le j-1} Y^{\mathbb{Z}}_u = X^{\mathbb{Z}}_j$ for any $j \in \mathbb{N}$. Hence, the hitting time T^{hit} for the random walk $\mathbf{S}^{\mathbb{Z}}$ as defined in Def. 62 is exactly the termination time $T^{\mathcal{P}}$. As we proved in Lemma 67 that $\mathbf{E}^{\mathcal{P}}_{x_0}(Y_0) = \mu_{\mathcal{P}}$ holds independent of $x_0 \in \mathbb{Z}$, we can use Lemma 63 for all x_0. So we get for all $x_0 \in \mathbb{Z}$:

- If $\mu_{\mathcal{P}} > 0$, then $\mathbf{P}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}}{=}\infty) \overset{\text{Lemma 66}}{=} \mathbb{P}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}}{=}\infty) > 0$, i.e., $\mathcal{P}$ is not AST.
- Note that as $\mathcal{P}$ is non-trivial (i.e., $p_0 \neq 1$), we have $\mathbf{P}^{\mathcal{P}}_{x_0}(Y^{\mathbb{Z}}_0 = 0) \neq 1$. So if $\mu_{\mathcal{P}} = 0$, then Lemma 63 implies $\mathbf{P}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}} = \infty) \overset{\text{Lemma 66}}{=} \mathbb{P}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}} = \infty) = 0$ but $\mathbf{E}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}}) \overset{\text{Lemma 66}}{=} \mathbb{E}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}}) = \infty$, i.e., $\mathcal{P}$ is AST but not PAST.
- If $\mu_{\mathcal{P}} < 0$, then $\mathbf{E}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}}) \overset{\text{Lemma 66}}{=} \mathbb{E}^{\mathcal{P}}_{x_0}(T^{\mathcal{P}}) < \infty$, i.e., $\mathcal{P}$ is PAST. $\square$

Example 68 (Termination of Variations of $\mathcal{P}^{rdw}_{race}$). We showed already in Sect. 4.2 that the drift of the program $\mathcal{P}^{rdw}_{race}$ in Ex. 14 is $-\frac{3}{2} < 0$. So by Thm. 18 this program is PAST, i.e., the hare is expected to overtake the tortoise in a finite number of iterations.

Now consider the modified program $\mathcal{P}$:

$$
\begin{array}{l}
\texttt{while}\ (x > 0)\ \{\\
\quad x = x + 1 \quad [\tfrac{6}{11}];\\
\quad x = x \qquad\quad [\tfrac{3}{11}];\\
\quad x = x - 1 \quad [0];\\
\quad x = x - 2 \quad [\tfrac{1}{11}];\\
\quad x = x - 3 \quad [0];\\
\quad x = x - 4 \quad [\tfrac{1}{11}];\\
\}
\end{array}
$$

The distance still increases with probability $\frac{6}{11}$ *but it decreases by at most* 4*. Its drift is* $\mu_{\mathcal{P}} = 1 \cdot \frac{6}{11} + 0 \cdot \frac{3}{11} + (-2) \cdot \frac{1}{11} + (-4) \cdot \frac{1}{11} = 0$*. Hence, on average the distance* x *between the tortoise and the hare remains unchanged after each loop iteration. By Thm. 18 this program is AST but not PAST. Hence, the hare wins with probability* 1*, but the expected number of required loop iterations is infinite.*

Finally, we change the probabilities to obtain the program $\mathcal{P}'$*:*

$$
\begin{array}{l}
\texttt{while}\ (x > 0)\ \{\\
\quad x = x + 1 \quad [\tfrac{6}{11}];\\
\quad x = x \qquad\quad [\tfrac{3}{11}];\\
\quad x = x - 1 \quad [\tfrac{1}{22}];\\
\quad x = x - 2 \quad [\tfrac{1}{22}];\\
\quad x = x - 3 \quad [\tfrac{1}{22}];\\
\quad x = x - 4 \quad [\tfrac{1}{22}];\\
\}
\end{array}
$$

Its drift is $\mu_{\mathcal{P}'} = 1 \cdot \frac{6}{11} + 0 \cdot \frac{3}{11} + \frac{1}{22} \cdot \sum_{-4 \leq j \leq -1} j = \frac{1}{11} > 0$*. Thus,* $\mathcal{P}'$ *is not AST by Thm. 18. So there is a positive probability that the hare never catches up with the tortoise and the race takes forever.*

Corollary 20 (Decision Procedure for (P)AST of CP programs). *For a non-trivial CP program* $\mathcal{P}$*,* $\mathcal{P}$ *is (P)AST iff* $\mathcal{P}^{rdw}$ *is (P)AST. Hence, Thm. 15 and 18 yield a decision procedure for AST and PAST of CP programs.*

Proof. If $\mathcal{P}$ has direct termination (i.e., $p' \neq 0$), then $\mathcal{P}$ and $\mathcal{P}^{rdw}$ are PAST by Thm. 10. Otherwise, by Thm. 15 we can reduce the termination of $\mathcal{P}$ to the termination of $\mathcal{P}^{rdw}$ on inputs which are in the image of $rdw_{\mathcal{P}}$. Note that the termination behavior of $\mathcal{P}^{rdw}$ is the same for all $x > 0$. Hence, to show that $\mathcal{P}$ is (P)AST iff $\mathcal{P}^{rdw}$ (P)AST, we prove that $rdw_{\mathcal{P}}$'s image also includes positive values. To see this, note that $\boldsymbol{a} \neq \boldsymbol{0}$ implies $\boldsymbol{a} \bullet \boldsymbol{a} > 0$. Hence, for any natural number $u > \frac{b}{\boldsymbol{a} \bullet \boldsymbol{a}}$ we obtain $rdw_{\mathcal{P}}(u \cdot \boldsymbol{a}) = u \cdot \boldsymbol{a} \bullet \boldsymbol{a} - b > \frac{b}{\boldsymbol{a} \bullet \boldsymbol{a}} \cdot \boldsymbol{a} \bullet \boldsymbol{a} - b = 0$. □

D.3 Proofs for Sect. 4.3

We now show that for CP programs $\mathcal{P}$ without direct termination, one can not only decide termination, but the construction for the proof of Thm. 18 also directly yields asymptotically exact bounds on their expected runtime. More precisely, we show that $rt_{x_0}^{\mathcal{P}}$ is asymptotically linear whenever $\mathcal{P}$ is PAST (and we even provide actual upper and lower bounds). To prove this result, we use *Wald's Lemma* from probability theory. Again, we first consider random walk programs and then use the reduction of Sect. 4.1 to lift our result to arbitrary CP programs.

Recall that if a stochastic process $\mathbf{Y} = (Y_j)_{j\in\mathbb{N}}$ on a probability space $(\Omega, \mathfrak{F}, \mathbb{P})$ is i.i.d., then $\mathbb{E}(Y_0) = \mathbb{E}(Y_j)$ for all $j \in \mathbb{N}$. Thus, we obtain

$$\mathbb{E}\left(\sum_{0\le j\le c-1} Y_j\right) = \sum_{0\le j\le c-1} \mathbb{E}(Y_j) = c\cdot\mathbb{E}(Y_0) \quad \text{for any constant } c\in\mathbb{N}.$$

By Wald's Lemma, a similar statement even holds if instead of the constant c we use a random variable T, provided that T is independent from the stochastic process $\mathbf{Y}$. We use a consequence of Wald's Lemma where T does not need to be independent from the whole process $\mathbf{Y}$ but for every j, the random variable Y_j is independent of whether T is greater or equal to $j+1$. The required independence can be expressed formally by demanding that Y_j must be independent of $\mathbb{I}_{\{T\ge j+1\}} : \Omega \to \{0,1\}$, where $\mathbb{I}_{\{T\ge j+1\}}(\pi) = 1$ if $T(\pi) \ge j+1$ and $\mathbb{I}_{\{T\ge j+1\}}(\pi) = 0$ otherwise. Then, to compute $\mathbb{E}\left(\sum_{0\le j\le T-1} Y_j\right)$, by Wald's Lemma one can apply $\mathbb{E}$ to both T and Y_n separately, i.e., one can compute $\mathbb{E}(T)\cdot\mathbb{E}(Y_0)$.

Lemma 69 (Consequence of Wald's Lemma, cf. [21, Lemma 10.2(9)]). *Let $\mathbf{Y} = (Y_j)_{j\in\mathbb{N}}$ be a stochastic process on a probability space $(\Omega, \mathfrak{F}, \mathbb{P})$ which is i.i.d. and let $T : \Omega \to \overline{\mathbb{N}}$ be a random variable. Define the random variable $(\sum_{0\le j\le T-1} Y_j) : \Omega \to \mathbb{R}, \pi \mapsto \sum_{0\le j\le T(\pi)-1} Y_j(\pi)$. If $\mathbb{E}(Y_0) < \infty$, $\mathbb{E}(T) < \infty$, and the random variables Y_j and $\mathbb{I}_{\{T\ge j+1\}}$ are independent for all $j\in\mathbb{N}$, then*

$$\mathbb{E}\left(\sum_{0\le j\le T-1} Y_j\right) = \mathbb{E}(T)\cdot\mathbb{E}(Y_0).$$

Proof. In [3, Thm. 17.7] it is shown that

$$\mathbb{E}\left(\sum_{0\le j\le T-1} Y_j\right) < \infty, \tag{16}$$

i.e., the expected value of $\sum_{0\le j\le T-1} Y_j$ exists. The proof of Lemma 69 is similar to the proof of [21, Lemma (9) in Sect. 10.2], but it is done under different preconditions.

$$\mathbb{E}\left(\sum_{0\le j\le T-1} Y_j\right)$$

$$
\begin{aligned}
&= \mathbb{E}\left(\sum_{0 \le j} Y_j \cdot \mathbb{I}_{\{T \ge j+1\}}\right) \\
&= \sum_{0 \le j} \mathbb{E}\left(Y_j \cdot \mathbb{I}_{\{T \ge j+1\}}\right) && \text{by the existence (16), i.e.,} \\
& && \text{since this expected value is} < \infty \\
&= \sum_{0 \le j} \mathbb{E}\left(Y_j\right) \cdot \mathbb{E}\left(\mathbb{I}_{\{T \ge j+1\}}\right) && \text{by independence of } Y_j \text{ and } \mathbb{I}_{\{T \ge j+1\}} \\
&= \sum_{0 \le j} \mathbb{E}\left(Y_0\right) \cdot \mathbb{E}\left(\mathbb{I}_{\{T \ge j+1\}}\right) && \text{as } \mathbb{E}(Y_j) = \mathbb{E}(Y_0)\text{, since } \mathbf{Y} \text{ is i.i.d.} \\
&= \mathbb{E}\left(Y_0\right) \cdot \sum_{0 \le j} \mathbb{E}\left(\mathbb{I}_{\{T \ge j+1\}}\right) \\
&= \mathbb{E}\left(Y_0\right) \cdot \sum_{0 \le j} \left(0 \cdot \mathbb{P}\left(\mathbb{I}_{\{T \ge j+1\}} = 0\right) + 1 \cdot \mathbb{P}\left(\mathbb{I}_{\{T \ge j+1\}} = 1\right)\right) \\
&= \mathbb{E}\left(Y_0\right) \cdot \sum_{0 \le j} \mathbb{P}\left(T \ge j+1\right) \\
&= \mathbb{E}\left(Y_0\right) \cdot \sum_{0 \le j} \sum_{j+1 \le u} \mathbb{P}\left(T = u\right) \\
&= \mathbb{E}\left(Y_0\right) \cdot \sum_{1 \le u} \sum_{0 \le j \le u-1} \mathbb{P}\left(T = u\right) \\
&= \mathbb{E}\left(Y_0\right) \cdot \sum_{1 \le u} u \cdot \mathbb{P}\left(T = u\right) \\
&= \mathbb{E}\left(Y_0\right) \cdot \mathbb{E}\left(T\right) && \square
\end{aligned}
$$

In our setting, we consider the stochastic process $\mathbf{Y}^{\mathbb{Z}}$ from Lemma 67 and the termination time $T^{\mathcal{P}}$. When regarding $\mathbb{P}^{\mathcal{P}}_{x_0}$, Y_j (i.e., the difference between the $(j+1)$-th and the j-th element of a run) is clearly not independent of the question whether the run already terminated in (or before) the j-th element. The reason is that under the probability measure $\mathbb{P}^{\mathcal{P}}_{x_0}$, the elements of a run do not change anymore after termination. However, Lemma 70 shows that when regarding $\mathbf{P}^{\mathcal{P}}_{x_0}$ instead, the independence requirement of Lemma 69 is fulfilled.

Lemma 70 (Independence of $Y^{\mathbb{Z}}_j$ and $\mathbb{I}_{\{T^{\mathcal{P}} \ge j+1\}}$). *Let $\mathbf{Y}^{\mathbb{Z}} = (Y^{\mathbb{Z}}_j)_{j \in \mathbb{N}}$ be the stochastic process from Lemma 67. Then for any random walk program $\mathcal{P}$ without direct termination, any $x_0 \in \mathbb{Z}$, and any $j \in \mathbb{N}$, the random variables $Y^{\mathbb{Z}}_j$ and $\mathbb{I}_{\{T^{\mathcal{P}} \ge j+1\}}$ are independent w.r.t. the probability measure $\mathbf{P}^{\mathcal{P}}_{x_0}$.*

Proof. We show that for any $x, y \in \mathbb{Z}$, we have

$$\mathbf{P}^{\mathcal{P}}_{x_0}\left(Y^{\mathbb{Z}}_j = x,\ \mathbb{I}_{\{T^{\mathcal{P}} \ge j+1\}} = y\right) \;=\; \mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x\right) \cdot \mathbf{P}^{\mathcal{P}}_{x_0}\left(\mathbb{I}_{\{T^{\mathcal{P}} \ge j+1\}} = y\right).$$

Note that the left- and the right-hand side are both zero whenever $y \notin \{0, 1\}$. Thus, it is enough to show the claim for $y = 0$ and $y = 1$.

Case 1: $y = 0$

$$\mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x,\ \mathbb{I}_{\{T^{\mathcal{P}} \ge j+1\}} = 0\right)$$

$$
\begin{aligned}
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(\biguplus_{0 \le u \le j} \ \biguplus_{\substack{\pi = \langle z_0,\dots,z_{j+1}\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u+1},\\ z_{j+1}-z_j = x}} Cyl^{\mathbb{Z}}(\pi) \right) \\
&= \sum_{0 \le u \le j} \ \sum_{\substack{\pi = \langle z_0,\dots,z_{j+1}\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u+1},\\ z_{j+1}-z_j = x}} \mathbf{P}^{\mathcal{P}}_{x_0}\left(Cyl^{\mathbb{Z}}(\pi)\right) && \text{as } \mathbf{P}^{\mathcal{P}}_{x_0} \text{ is a prob. measure} \\
&= \sum_{0 \le u \le j} \ \sum_{\substack{\pi = \langle z_0,\dots,z_{j+1}\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u+1},\\ z_{j+1}-z_j = x}} q^{\mathcal{P}}_{x_0}(\pi) && \text{by Def. 64} \\
&= \sum_{0 \le u \le j} \ \sum_{\substack{\pi = \langle z_0,\dots,z_{j+1}\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u+1},\\ z_{j+1}-z_j = x}} \delta_{x_0,z_0} \cdot \prod_{0 \le v \le j} p_{z_{v+1}-z_v} && \text{by Def. 64} \\
&= \left(\sum_{0 \le u \le j} \ \sum_{\pi = \langle z_0,\dots,z_j\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u}} \delta_{x_0,z_0} \cdot \prod_{0 \le v \le j-1} p_{z_{v+1}-z_v} \right) \cdot p_x \\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x\right) \cdot \left(\sum_{0 \le u \le j} \ \sum_{\pi = \langle z_0,\dots,z_j\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u}} \delta_{x_0,z_0} \cdot \prod_{0 \le v \le j-1} p_{z_{v+1}-z_v} \right) \\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x\right) \cdot \left(\sum_{0 \le u \le j} \ \sum_{\pi = \langle z_0,\dots,z_j\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u}} q^{\mathcal{P}}_{x_0}(\pi) \right) \\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x\right) \cdot \left(\sum_{0 \le u \le j} \ \sum_{\pi = \langle z_0,\dots,z_j\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u}} \mathbf{P}^{\mathcal{P}}_{x_0}\left(Cyl^{\mathbb{Z}}(\pi)\right) \right) && \text{by Def. 64} \\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x\right) \cdot \mathbf{P}^{\mathcal{P}}_{x_0}\left(\biguplus_{0 \le u \le j} \ \biguplus_{\pi = \langle z_0,\dots,z_j\rangle \in \mathbb{Z}^u_{>0} \times \mathbb{Z}_{\le 0} \times \mathbb{Z}^{j-u}} Cyl^{\mathbb{Z}}(\pi) \right) && \text{prob. measure} \\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x\right) \cdot \mathbf{P}^{\mathcal{P}}_{x_0}\left(\mathbb{I}_{\{T^{\mathcal{P}} \ge j+1\}} = 0\right)
\end{aligned}
$$

Case 2: $y = 1$

$$
\begin{aligned}
&\mathbf{P}^{\mathcal{P}}_{x_0}\left(Y_j = x,\ \mathbb{I}_{\{T^{\mathcal{P}} \ge j+1\}} = 1\right) \\
&= \mathbf{P}^{\mathcal{P}}_{x_0}\left(\biguplus_{\substack{\pi = \langle z_0,\dots,z_{j+1}\rangle \in \mathbb{Z}^{j+1}_{>0} \times \mathbb{Z},\\ z_{j+1}-z_j = x}} Cyl^{\mathbb{Z}}(\pi) \right) \\
&= \sum_{\substack{\pi = \langle z_0,\dots,z_{j+1}\rangle \in \mathbb{Z}^{j+1}_{>0} \times \mathbb{Z},\\ z_{j+1}-z_j = x}} \mathbf{P}^{\mathcal{P}}_{x_0}\left(Cyl^{\mathbb{Z}}(\pi)\right) && \text{as } \mathbf{P}^{\mathcal{P}}_{x_0} \text{ is a prob. measure} \\
&= \sum_{\substack{\pi = \langle z_0,\dots,z_{j+1}\rangle \in \mathbb{Z}^{j+1}_{>0} \times \mathbb{Z},\\ z_{j+1}-z_j = x}} q^{\mathcal{P}}_{x_0}(\pi) && \text{by Def. 64}
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{\substack{\pi=\langle z_0,\ldots,z_{j+1}\rangle\in\mathbb{Z}_{>0}^{j+1}\times\mathbb{Z},\\ z_{j+1}-z_j=x}} \delta_{x_0,z_0}\cdot \prod_{0\le v\le j} p_{z_{v+1}-z_v} && \text{by Def. 64}\\
&= \left(\sum_{\pi=\langle z_0,\ldots,z_j\rangle\in\mathbb{Z}_{>0}^{j+1}} \delta_{x_0,z_0}\cdot \prod_{0\le v\le j-1} p_{z_{v+1}-z_v}\right)\cdot p_x\\
&= \mathbf{P}_{x_0}^{\mathcal{P}}\left(Y_j=x\right)\cdot \left(\sum_{\pi=\langle z_0,\ldots,z_j\rangle\in\mathbb{Z}_{>0}^{j+1}} \delta_{x_0,z_0}\cdot \prod_{v=0}^{j-1} p_{z_{v+1}-z_v}\right)\\
&= \mathbf{P}_{x_0}^{\mathcal{P}}\left(Y_j=x\right)\cdot \left(\sum_{\pi=\langle z_0,\ldots,z_j\rangle\in\mathbb{Z}_{>0}^{j+1}} q_{x_0}^{\mathcal{P}}(\pi)\right) && \text{by Def. 64}\\
&= \mathbf{P}_{x_0}^{\mathcal{P}}\left(Y_j=x\right)\cdot \left(\sum_{\pi=\langle z_0,\ldots,z_j\rangle\in\mathbb{Z}_{>0}^{j+1}} \mathbf{P}_{x_0}^{\mathcal{P}}\left(Cyl^{\mathbb{Z}}(\pi)\right)\right) && \text{by Def. 64}\\
&= \mathbf{P}_{x_0}^{\mathcal{P}}\left(Y_j=x\right)\cdot \mathbf{P}_{x_0}^{\mathcal{P}}\left(\biguplus_{\pi=\langle z_0,\ldots,z_j\rangle\in\mathbb{Z}_{>0}^{j+1}} Cyl^{\mathbb{Z}}(\pi)\right) && \text{as } \mathbf{P}_{x_0}^{\mathcal{P}} \text{ is a prob. measure}\\
&= \mathbf{P}_{x_0}^{\mathcal{P}}\left(Y_j=x\right)\cdot \mathbf{P}_{x_0}^{\mathcal{P}}\left(\mathbb{I}_{\{T^{\mathcal{P}}\ge j+1\}}=1\right) && \square
\end{aligned}
$$

Now we can use Lemma 69 to infer linear upper and lower bounds for the expected runtime if the random walk program $\mathcal{P}$ is PAST (i.e., if $\mu_{\mathcal{P}} < 0$).

Theorem 71 (Bounds on the Expected Runtime of Random Walk Programs). *Let $\mathcal{P}$ be a random walk program as in Def. 12 without direct termination where $\mu_{\mathcal{P}} < 0$. Then $rt_{x_0}^{\mathcal{P}} = 0$ for $x_0 \le 0$ and for $x_0 > 0$, we have*

$$-\tfrac{1}{\mu_{\mathcal{P}}}\cdot x_0 \quad\le\quad rt_{x_0}^{\mathcal{P}} \quad\le\quad -\tfrac{1}{\mu_{\mathcal{P}}}\cdot x_0 + \tfrac{1-k}{\mu_{\mathcal{P}}}.$$

So for $x_0 > 0$, $\mathcal{P}$'s expected runtime is asymptotically linear, i.e., $rt_{x_0}^{\mathcal{P}} \in \Theta(x_0)$.

Proof. All prerequisites are satisfied to apply Wald's Lemma (Lemma 69) for the stochastic process $\mathbf{Y}^{\mathbb{Z}}$ on the probability space $(\mathbb{Z}^\omega, \mathfrak{F}^{\mathbb{Z}}, \mathbf{P}_{x_0}^{\mathcal{P}})$ and the termination time $T^{\mathcal{P}}$: By Lemma 67, $\mathbf{Y}^{\mathbb{Z}}$ is i.i.d. w.r.t. $(\mathbb{Z}^\omega, \mathfrak{F}^{\mathbb{Z}}, \mathbf{P}_{x_0}^{\mathcal{P}})$ and $\mathbf{E}_{x_0}^{\mathbb{Z}}(Y_0^{\mathbb{Z}}) = \mu_{\mathcal{P}} < \infty$. Since $\mu_{\mathcal{P}} < 0$, Thm. 18 yields that $\mathcal{P}$ is PAST and hence $rt_{x_0}^{\mathcal{P}} = \mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right) < \infty$. By Lemma 66 this implies $\mathbf{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right) = \mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right) < \infty$. Furthermore, $Y_j^{\mathbb{Z}}$ and $\mathbb{I}_{\{T^{\mathcal{P}}\ge j+1\}}$ are independent by Lemma 70. Thus, Lemma 69 yields

$$\mathbf{E}_{x_0}^{\mathcal{P}}\left(\sum_{0\le j\le T^{\mathcal{P}}-1} Y_j^{\mathbb{Z}}\right) \;=\; \mathbf{E}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}})\cdot \mathbf{E}_{x_0}^{\mathcal{P}}(Y_0^{\mathbb{Z}}). \tag{17}$$

Let the random variable $X_{T^{\mathcal{P}}} : \Omega \to \mathbb{Z}$ map every run π to the first non-positive value in π, i.e., to the value of the program variable when $\mathcal{P}$ terminates, or 0 otherwise. So $X_{T^{\mathcal{P}}}(\pi) = X_{T^{\mathcal{P}}(\pi)}(\pi)$ if $T^{\mathcal{P}}(\pi) < \infty$ and $X_{T^{\mathcal{P}}}(\pi) = 0$ if $T^{\mathcal{P}}(\pi) = \infty$.

To infer linear bounds on the expected value of the termination time $\mathbf{E}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}})$ resp. $\mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right)$, we first infer bounds on $\mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}})$. Clearly, we have $X_{T^{\mathcal{P}}}(\pi) \leq 0$ for every $\pi \in \Omega$ by the definition of the termination time and of $X_{T^{\mathcal{P}}}$. Hence, this implies $\mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}}) \leq 0$, i.e., 0 is an upper bound for $\mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}})$.

To infer a lower bound for $\mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}})$, note that if $x_0 > 0$, then for every run $\pi = \langle z_0, \ldots, z_{j-1}, z_j, \ldots \rangle$ where $\mathbf{P}_{x_0}^{\mathcal{P}}(Cyl^{\mathbb{Z}}(\pi)) = q_{x_0}^{\mathcal{P}}(\pi) > 0$ and z_j is the first non-positive value in π, we have $j \geq 1$ and z_j is at most k smaller than z_{j-1}. Thus, $z_{j-1} \geq 1$ implies $z_j \geq z_{j-1} - k \geq 1 - k$. Hence, for all these runs we have $X_{T^{\mathcal{P}}}(\pi) = z_j \geq 1 - k$. Moreover, for runs π without non-positive values, we also have $X_{T^{\mathcal{P}}}(\pi) \geq 1 - k$, since $X_{T^{\mathcal{P}}}(\pi) = 0$ and since $\mu_{\mathcal{P}} < 0$ implies $k \geq 1$. Thus, we obtain $\mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}}) \geq 1 - k$ whenever $x_0 > 0$.

So to summarize, we get the following upper and lower bounds for $\mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}})$ if $x_0 > 0$:

$$1 - k \quad \leq \quad \mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}}) \quad \leq \quad 0 \tag{18}$$

Recall that for every $j \geq 0$ we have $X_j^{\mathbb{Z}} = X_0^{\mathbb{Z}} + \sum_{0 \leq u \leq j-1} Y_u^{\mathbb{Z}}$. Hence, we also have $X_{T^{\mathcal{P}}} = X_0^{\mathbb{Z}} + \sum_{0 \leq u \leq T^{\mathcal{P}}-1} Y_u^{\mathbb{Z}}$. This implies:

$$\begin{aligned}
\mathbf{E}_{x_0}^{\mathcal{P}}(X_{T^{\mathcal{P}}}) &= \mathbf{E}_{x_0}^{\mathcal{P}}(X_0^{\mathbb{Z}}) + \mathbf{E}_{x_0}^{\mathcal{P}}\left(\sum_{0 \leq u \leq T^{\mathcal{P}}-1} Y_u^{\mathbb{Z}}\right) & \\
&= x_0 + \mathbf{E}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}}) \cdot \mathbf{E}_{x_0}^{\mathcal{P}}(Y_0^{\mathbb{Z}}) & \text{by (17)} \\
&= x_0 + \mathbf{E}_{x_0}^{\mathcal{P}}(T^{\mathcal{P}}) \cdot \mu_{\mathcal{P}} & \text{by Lemma 67} \\
&= x_0 + \mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right) \cdot \mu_{\mathcal{P}} & \text{by Lemma 66.}
\end{aligned}$$

Hence, by (18) we obtain $-\frac{1}{\mu_{\mathcal{P}}} \cdot x_0 \;\leq\; \mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right) \;\leq\; -\frac{1}{\mu_{\mathcal{P}}} \cdot x_0 + \frac{1-k}{\mu_{\mathcal{P}}}$ for any $x_0 > 0$. This implies the theorem, since $rt_{x_0}^{\mathcal{P}} = \mathbb{E}_{x_0}^{\mathcal{P}}\left(T^{\mathcal{P}}\right)$. □

Theorem 21 (Bounds on the Expected Runtime of CP Programs). *Let $\mathcal{P}$ be a non-trivial CP program as in Def. 2 without direct termination which is PAST (i.e., $\mu_{\mathcal{P}^{rdw}} < 0$). Moreover, let $k_{\mathcal{P}}$ be obtained according to the transformation from Def. 13. If $rdw_{\mathcal{P}}(\boldsymbol{x}_0) \leq 0$, then $rt_{\boldsymbol{x}_0}^{\mathcal{P}} = 0$. If $rdw_{\mathcal{P}}(\boldsymbol{x}_0) > 0$, then $\mathcal{P}$'s expected runtime is asymptotically linear and we have*

$$-\frac{1}{\mu_{\mathcal{P}^{rdw}}} \cdot rdw_{\mathcal{P}}(\boldsymbol{x}_0) \quad \leq \quad rt_{\boldsymbol{x}_0}^{\mathcal{P}} \quad \leq \quad -\frac{1}{\mu_{\mathcal{P}^{rdw}}} \cdot rdw_{\mathcal{P}}(\boldsymbol{x}_0) + \frac{1-k_{\mathcal{P}}}{\mu_{\mathcal{P}^{rdw}}}.$$

Proof. The result directly follows from Thm. 15 and 71.

E Proofs for Sect. 5

Lemma 26 (Number of Roots With Absolute Value ≤ 1). *Let $\mathcal{P}$ be a random walk program as in Def. 12 that is PAST. Then the characteristic polynomial $\chi_{\mathcal{P}}$ has k roots $\lambda \in \mathbb{C}$ (counted with multiplicity) with $|\lambda| \leq 1$.*

Proof. We use Rouché's Theorem: For a univariate polynomial $a_v \cdot x^v + \ldots + a_1 \cdot x + a_0$, if there is a number $w \in \mathbb{R}_{>0}$ and an index $u \in \mathbb{N}$ with $0 \leq u \leq v$ such that

$$|a_u| \cdot w^u \ > \sum_{0 \leq j \leq v,\ j \neq u} |a_j| \cdot w^j, \tag{19}$$

then the polynomial has exactly u (possibly complex) roots (counted with multiplicity) of absolute value less than w.

We now apply Rouché's Theorem to the characteristic polynomial and proceed by case analysis. First, we consider the case where $p' > 0$. Here, we choose $w = 1$ and $u = k$. Then (19) becomes

$$|p_0 - 1| \ > \sum_{-k \leq j \leq m,\ j \neq 0} |p_j|.$$

As $|p_0 - 1| = 1 - p_0$ and $|p_j| = p_j$ for all j, this is equivalent to

$$1 \ > \sum_{-k \leq j \leq m} p_j \ = \ 1 - p'$$

which is true since $p' > 0$. So by Rouché's Theorem, the characteristic polynomial $\chi_{\mathcal{P}}$ has k roots λ with $|\lambda| < 1$.

However, we would like to conclude that there are no more than k roots λ with $|\lambda| \leq 1$. Thus, we still need to show that $\chi_{\mathcal{P}}$ has no root λ with $|\lambda| = 1$. Clearly, $0 = \chi_{\mathcal{P}}(\lambda)$ is equivalent to $0 = \sum_{-k \leq j \leq m} p_j \cdot \lambda^{k+j} - \lambda^k$. If $|\lambda| = 1$ were true, then $1 = \sum_{-k \leq j \leq m} p_j \cdot \lambda^j$ and

$$1 \ = \ |1| \ \leq \sum_{-k \leq j \leq m} |p_j| \cdot |\lambda|^j \ = \sum_{-k \leq j \leq m} p_j \ = \ 1 - p'$$

by using $|p_j| = p_j$. However, this is a contradiction to $p' > 0$.

Now we consider the case where $p' = 0$ and thus $\sum_{-k \leq j \leq m} p_j = 1$. Our goal is to show that for all small enough $\varepsilon > 0$, the inequality (19) holds if we set $w = 1 + \varepsilon$ and $u = k$. Then (19) becomes

$$|p_0 - 1| \cdot w^k \ > \sum_{-k \leq j \leq m,\ j \neq 0} |p_j| \cdot w^{j+k}.$$

As $|p_0 - 1| = 1 - p_0$, $w = 1 + \varepsilon$, and $|p_j| = p_j$ for all j, this is equivalent to

$$(1 - p_0) \cdot (1 + \varepsilon)^k \ > \sum_{-k \leq j \leq m,\ j \neq 0} p_j \cdot (1 + \varepsilon)^{j+k}.$$

Note that[7] $(1+\varepsilon)^j = 1 + j \cdot \varepsilon + \mathcal{O}(\varepsilon^2)$ for any $j \geq 0$. Hence, we obtain

$$\begin{aligned} &(1-p_0) + k \cdot (1-p_0) \cdot \varepsilon + \mathcal{O}(\varepsilon^2) \\ > &\left(\sum_{-k \leq j \leq m,\ j \neq 0} p_j \right) + \left(\sum_{-k \leq j \leq m,\ j \neq 0} p_j \cdot (j+k) \cdot \varepsilon \right) + \mathcal{O}(\varepsilon^2). \end{aligned}$$

By using $\sum_{-k \leq j \leq m} p_j = 1$, this simplifies to

$$k \cdot (1-p_0) \cdot \varepsilon + \mathcal{O}(\varepsilon^2) \ > \ \left(\sum_{-k \leq j \leq m,\ j \neq 0} p_j \cdot (j+k) \cdot \varepsilon \right) + \mathcal{O}(\varepsilon^2).$$

When dividing by $\varepsilon > 0$, we get

$$k \cdot (1-p_0) + \mathcal{O}(\varepsilon) \ > \ \left(\sum_{-k \leq j \leq m,\ j \neq 0} p_j \cdot (j+k) \right) + \mathcal{O}(\varepsilon).$$

To satisfy this, it is sufficient to have

$$k \cdot (1-p_0) \ > \ \left(\sum_{-k \leq j \leq m,\ j \neq 0} p_j \cdot (j+k) \right) + \mathcal{O}(\varepsilon).$$

This is equivalent to

$$\begin{aligned} k &> \sum_{-k \leq j \leq m} p_j \cdot (j+k) + \mathcal{O}(\varepsilon) \\ &= \sum_{-k \leq j \leq m} p_j \cdot j + k + \mathcal{O}(\varepsilon) \\ &= \mu_{\mathcal{P}} + k + \mathcal{O}(\varepsilon). \end{aligned}$$

Since $\mu_{\mathcal{P}} < 0$ as $\mathcal{P}$ is PAST (cf. Thm. 18), this is true for all sufficiently small ε. Hence, there are exactly k roots of absolute value less than $1+\varepsilon$, where ε is sufficiently small, so in particular k roots of absolute value ≤ 1. $\square$

Lemma 28 (Unique Solution of (4) and (5) when Disregarding Roots With Absolute Value > 1). *Let $\mathcal{P}$ be a random walk program as in Def. 12 that is PAST. Then there is exactly one function $f : \mathbb{Z} \to \mathbb{C}$ which satisfies both (4) and (5) (thus, it has the form (9)) and has $a_{j,u} = 0$ whenever $|\lambda_j| > 1$.*

Proof. To encode the requirement on the $a_{j,u}$, we modify (5) into a new constraint (20) which ensures $a_{j,u} = 0$ whenever $|\lambda_j| > 1$. More precisely, this new constraint

[7] This notation means that $(1+\varepsilon)^j = 1 + j \cdot \varepsilon + f(\varepsilon)$ for a function f with $f(x) \in \mathcal{O}(x^2)$. Here, k, m, and the p_j are considered to be constants, i.e., we write $\mathcal{O}(\varepsilon^2)$ instead of $(1-p_0) \cdot \mathcal{O}(\varepsilon^2)$ or $\sum_{-k \leq j \leq m,\ j \neq 0} p_j \cdot \mathcal{O}(\varepsilon^2)$.

(20) is a recurrence equation such that the characteristic polynomial χ_0 of its homogeneous part has all the roots of $\chi_{\mathcal{P}}$ except those whose absolute value is greater than 1, i.e., $\chi_0(\lambda) = \prod_{1 \leq j \leq c,\ |\lambda_j| \leq 1} (\lambda - \lambda_j)^{v_j}$. Thus, we can define the coefficients $q_j \in \mathbb{C}$ by

$$\chi_0(\lambda) \quad = \quad \prod_{1 \leq j \leq c,\ |\lambda_j| \leq 1} (\lambda - \lambda_j)^{v_j} \quad = \quad \lambda^k - \sum_{-k \leq j \leq -1} q_j \cdot \lambda^{k+j}.$$

Note that the degree of the polynomial χ_0 is indeed k, because by Lemma 26 we have $\sum_{1 \leq j \leq c,\ |\lambda_j| \leq 1} v_j = k$.

Moreover, the constant add-on of the new recurrence equation is constructed in such a way that the particular solutions C_{const} resp. $C_{lin} \cdot x$ of (5) are also solutions of the inhomogeneous recurrence equation. Thus, let $D_{const} = C_{const} \cdot \left(1 - \sum_{-k \leq j \leq -1} q_j\right)$ and $D_{lin} = -C_{lin} \cdot \sum_{-k \leq j \leq -1} j \cdot q_j$. Instead of (5), we now consider the constraint

$$f(x) \;=\; \sum_{-k \leq j \leq -1} q_j \cdot f(x+j) + D \quad \text{for all } x > 0, \tag{20}$$

where we choose $D = D_{const}$ if $p' > 0$ and $D = D_{lin}$ if $p' = 0$. We show the following two claims:

(a) There is exactly one function $f : \mathbb{Z} \to \mathbb{C}$ which satisfies (4) and (20).
(b) A function $f : \mathbb{Z} \to \mathbb{C}$ satisfies (20) iff f satisfies (5) (thus, it has the form (9)) where $a_{j,u} = 0$ whenever $|\lambda_j| > 1$.

These two claims imply the statement of the lemma. To see this, note that by (a) there exists a function which satisfies (4) and (20) and by (b) this function also satisfies (5) and it has $a_{j,u} = 0$ whenever $|\lambda_j| > 1$. This function is unique, because if there were two different functions f_1 and f_2 that satisfy (4) and (5) and have $a_{j,u} = 0$ whenever $|\lambda_j| > 1$, then by (b) these two functions would also both satisfy (20). But this would be a contradiction to the uniqueness stated in (a).

We now prove the claims (a) and (b). For (a), note that the recurrence equation (20) is formulated in such a way that $f(x)$ only depends on the values of f on the smaller values $x - 1, \ldots, x - k$ (i.e., it is a recurrence of order k). By the constraint (4), the initial value of f on negative values is uniquely determined (i.e., $f(0) = f(-1) = \ldots = f(-k+1) = 0$). Hence, by induction on x, one can easily prove that there is a single unique function $f : \mathbb{Z} \to \mathbb{C}$ that satisfies both (4) and (20).

For the claim (b), we only have to show that C_{const} is a solution of the inhomogeneous recurrence equation (20) if $p' > 0$ and $C_{lin} \cdot x$ is a solution of (20) if $p' = 0$. Once this is shown, it is clear that all solutions of (20) result from adding the particular solution C_{const} resp. $C_{lin} \cdot x$ of the inhomogeneous equation to a solution of the homogeneous variant of (20) (where D is replaced by 0). Any solution of this homogeneous variant can be represented as a linear combination of the solutions $\lambda_j^x \cdot x^u$ where $|\lambda_j| \le 1$ and $u \in \{0, \dots, v_j - 1\}$. That these are linearly independent solutions of the homogeneous variant of (20) follows from the fact that χ_0 is the corresponding characteristic polynomial. Thus, the solutions of (20) are all functions of the form (9) where $a_{j,u} = 0$ whenever $|\lambda_j| > 1$, which proves (b).

It remains to show that C_{const} resp. $C_{lin} \cdot x$ are particular solutions of the inhomogeneous recurrence equation (20). If $p' > 0$, then the definition of D_{const} indeed implies $C_{const} = C_{const} \cdot \sum_{-k \le j \le -1} q_j + D_{const}$. If $p' = 0$, then we have to show

$$C_{lin} \cdot x \;=\; C_{lin} \cdot \sum_{-k \le j \le -1} q_j \cdot (x + j) + D_{lin}. \tag{21}$$

Since 1 is a root of $\chi_{\mathcal{P}}$ (i.e., one of the λ_j with $|\lambda_j| \le 1$ is $\lambda_j = 1$), 1 is also a root of χ_0. So we have $0 = \chi_0(1) = 1 - \sum_{-k \le j \le -1} q_j$, which implies $\sum_{-k \le j \le -1} q_j = 1$. So (21) is equivalent to

$$\begin{aligned} C_{lin} \cdot x &= C_{lin} \cdot (x \cdot \sum_{-k \le j \le -1} q_j + \sum_{-k \le j \le -1} j \cdot q_j) + D_{lin} \\ &= C_{lin} \cdot (x + \sum_{-k \le j \le -1} j \cdot q_j) + D_{lin}. \end{aligned}$$

This holds due to the definition of D_{lin}. □

Theorem 29 (Exact Expected Runtime for Random Walk Programs). *Let $\mathcal{P}$ be a random walk program as in Def. 12 that is PAST and let $\lambda_1, \dots, \lambda_c$ be the roots of its characteristic polynomial with multiplicities $v_1, \dots, v_c$. Moreover, let $C(x) = C_{const} = \frac{1}{p'}$ if $p' > 0$ and $C(x) = C_{lin} \cdot x = -\frac{1}{\mu_{\mathcal{P}}} \cdot x$ if $p' = 0$. Then the expected runtime of $\mathcal{P}$ is $rt_x^{\mathcal{P}} = 0$ for $x \le 0$ and*

$$rt_x^{\mathcal{P}} \;=\; C(x) \;+\; \sum_{1 \le j \le c,\; |\lambda_j| \le 1} \; \sum_{0 \le u \le v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u \quad \text{for } x > 0,$$

where the coefficients $a_{j,u}$ are the unique solution of the k linear equations:

$$0 \;=\; C(x) + \sum_{1 \le j \le c,\; |\lambda_j| \le 1} \sum_{0 \le u \le v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u \quad \text{for } -k+1 \le x \le 0 \tag{11}$$

So in the special case where $k = 0$, we have $rt_x^{\mathcal{P}} = C(x) = C_{const} = \frac{1}{p'}$ for $x > 0$.

Proof. By Thm. 9, the expected runtime $rt_x^{\mathcal{P}}$ is the least fixpoint of the expected runtime transformer $\mathcal{L}^{\mathcal{P}}$, i.e., the smallest function $f(x) : \mathbb{Z} \to \overline{\mathbb{R}_{\ge 0}}$ which satisfies (3), or equivalently, the smallest function which satisfies (4) and (5).

Since f satisfies (5), it is a function of the form (9), i.e., there exist coefficients $a_{j,u} \in \mathbb{C}$ such that for all $x > -k$ we have

$$f(x) \;=\; C(x) \;+\; \sum_{1 \le j \le c} \; \sum_{0 \le u \le v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u.$$

If we had $a_{j,u} \neq 0$ for a coefficient where $|\lambda_j| > 1$, then $f(x)$ would not be bounded by a constant (if $p' > 0$) resp. by a linear function (if $p' = 0$). Thus, this would contradict Thm. 10 (if $p' > 0$) resp. Thm. 21 (if $p' = 0$).

By Lemma 28 there is a single unique function $f : \mathbb{Z} \to \mathbb{C}$ which satisfies both (4) and (5) and has $a_{j,u} = 0$ whenever $|\lambda_j| > 1$. So this function must be the expected runtime (and hence, it maps any integer to a non-negative real number). Due to (5) the function must be of the form (9) for all $x > -k$ but at the same time it also has to satisfy $f(x) = 0$ for all $x \leq 0$ due to (4). Therefore, it must satisfy the linear equations (11). On the other hand, the linear equations (11) cannot have more than one solution because otherwise this would yield two different functions that satisfy both (4) and (5) and have $a_{j,u} = 0$ whenever $|\lambda_j| > 1$, in contradiction to Lemma 28.

If $k = 0$, then $p' > 0$ as $\mathcal{P}$ is PAST. Lemma 26 implies that $\chi_{\mathcal{P}}$ has no root with $|\lambda| \leq 1$ and thus, $rt_x^{\mathcal{P}} = C_{const} + \sum_{1 \leq j \leq c,\ |\lambda_j| \leq 1} \ldots = C_{const}$ for $x > 0$. □

Corollary 31 (Exact Expected Runtime for CP Programs). *For any CP program, its expected runtime can be computed exactly.*

Proof. If $\mathcal{P}$ is trivial, then its expected runtime is obvious. Otherwise, by Cor. 20 one can decide if $\mathcal{P}$ is PAST and in that case, $\mathcal{P}^{rdw}$ is PAST as well. For any CP program $\mathcal{P}$, we have $rt_{\boldsymbol{x}}^{\mathcal{P}} = rt_{rdw_{\mathcal{P}}(\boldsymbol{x})}^{\mathcal{P}^{rdw}}$ due to Thm. 15. As $rt_{rdw_{\mathcal{P}}(\boldsymbol{x})}^{\mathcal{P}^{rdw}}$ can be computed exactly by Thm. 29, this also holds for $rt_{\boldsymbol{x}}^{\mathcal{P}}$. □

As mentioned in Sect. 5, Thm. 29 and Cor. 31 imply that for any $\boldsymbol{x}_0 \in \mathbb{Z}^r$, the expected runtime $rt_{\boldsymbol{x}_0}^{\mathcal{P}}$ of a CP program $\mathcal{P}$ that is PAST and has only *rational* probabilities $p_{\boldsymbol{c}_1}, \ldots, p_{\boldsymbol{c}_n}, p' \in \mathbb{Q}$ is always an algebraic number. This is due to the fact that $rt_{\boldsymbol{x}_0}^{\mathcal{P}}$ can be represented as a linear combination of algebraic numbers (the roots of the characteristic polynomial $\chi_{\mathcal{P}^{rdw}}$). The coefficients of this linear combination are the solution of a linear equation system (11) over algebraic numbers and hence, they are algebraic numbers themselves. Therefore, one could also compute a closed form for the exact expected runtime $rt_{\boldsymbol{x}}^{\mathcal{P}}$ using a representation with algebraic numbers instead of numerical approximations.

As also discussed in Sect. 5, while the exact computation of the expected runtime of a random walk program $\mathcal{P}$ according to Thm. 29 may yield a representation of $rt_x^{\mathcal{P}}$ with possibly complex number, one can easily obtain a more intuitive representation of $rt_x^{\mathcal{P}}$ that uses real numbers only.

As stated before, for any coefficients $a_{j,u}, a'_{j,u} \in \mathbb{C}$ with $j \in \{s+1, \ldots, s+t\}$ and $u \in \{0, \ldots, v_j - 1\}$ there exist coefficients $b_{j,u}$ and $b'_{j,u}$ such that

$$a_{j,u} \cdot \lambda_j^x + a'_{j,u} \cdot \overline{\lambda_j}^x \quad = \quad b_{j,u} \cdot \mathrm{Re}(\lambda_j^x) + b'_{j,u} \cdot \mathrm{Im}(\lambda_j^x)$$

holds for all $x \in \mathbb{Z}$. More precisely, $b_{j,u} = a_{j,u} + a'_{j,u}$ and $b'_{j,u} = (a_{j,u} - a'_{j,u}) \cdot i$. So any linear combination of the functions $\lambda_j^x \cdot x^u$ and $\overline{\lambda_j}^x \cdot x^u$ can be replaced by a linear combination of the functions $\mathrm{Re}(\lambda_j^x) \cdot x^u$ and $\mathrm{Im}(\lambda_j^x) \cdot x^u$. In this way, one obtains $k + m$ linearly independent *real* solutions of the corresponding homogeneous recurrence equation. Hence, by Thm. 29 we now get the representation of the expected runtime in (12):

$$rt_x^{\mathcal{P}} = \begin{cases} C(x) + \sum\limits_{1 \le j \le s,\ |\lambda_j| \le 1} \ \sum\limits_{0 \le u \le v_j - 1} a_{j,u} \cdot \lambda_j^x \cdot x^u \\ \quad + \sum\limits_{s+1 \le j \le s+t,\ |\lambda_j| \le 1} \ \sum\limits_{0 \le u \le v_j - 1} \big(b_{j,u} \cdot \mathrm{Re}(\lambda_j^x) + b'_{j,u} \cdot \mathrm{Im}(\lambda_j^x)\big) \cdot x^u, & \text{for } x > 0 \\ 0, & \text{for } x \le 0 \end{cases}$$

Since $rt_x^{\mathcal{P}}$ is real-valued, $\lambda_j^x \in \mathbb{R}$ for $j \in \{1, \dots, s\}$, and $\mathrm{Re}(\lambda_j^x), \mathrm{Im}(\lambda_j^x) \in \mathbb{R}$ for $j \in \{s+1, \dots, s+t\}$, all $a_{j,u}$ for $j \in \{1, \dots, s\}$ and all $b_{j,u}, b'_{j,u}$ for $j \in \{s+1, \dots, s+t\}$ are real numbers. As $b_{j,u} = a_{j,u} + a'_{j,u}$, this means that $a'_{j,u}$ is the conjugate of $a_{j,u}$, i.e., $a'_{j,u} = \overline{a_{j,u}}$ and thus, $b_{j,u} = 2 \cdot \mathrm{Re}(a_{j,u})$ and $b'_{j,u} = -2 \cdot \mathrm{Im}(a_{j,u})$.

As mentioned, to compute $\mathrm{Re}(\lambda_j^x)$ and $\mathrm{Im}(\lambda_j^x)$, we consider the polar representation of the non-real roots λ_j, i.e., for $j \in \{s+1, \dots, s+t\}$ let $\lambda_j = w_j \cdot e^{\theta_j \cdot i}$ with $w_j \in \mathbb{R}_{>0}$ and $\theta_j \in (0, 2\pi)$. Then $\lambda_j^x = w_j^x \cdot e^{\theta_j \cdot i \cdot x}$, and $\mathrm{Re}(\lambda_j^x) = w_j^x \cdot \cos(\theta_j \cdot x)$ and $\mathrm{Im}(\lambda_j^x) = w_j^x \cdot \sin(\theta_j \cdot x)$.

Note that in Alg. 33, one could also already use the representation in (12) with $\mathrm{Re}(\lambda_j^x) = w_j^x \cdot \cos(\theta_j \cdot x)$ and $\mathrm{Im}(\lambda_j^x) = w_j^x \cdot \sin(\theta_j \cdot x)$ here. Then one would only have to solve a system of linear equations over the reals and can compute $b_{j,u}$ and $b'_{j,u}$ directly.