The Big-O Problem for Labelled Markov Chains and Weighted Automata

Dmitry Chistikov
Centre for Discrete Mathematics and its Applications (DIMAP) and
Department of Computer Science, University of Warwick, Coventry, UK

Stefan Kiefer
Department of Computer Science, University of Oxford, UK

Andrzej S. Murawski
Department of Computer Science, University of Oxford, UK

David Purser
Centre for Discrete Mathematics and its Applications (DIMAP) and
Department of Computer Science, University of Warwick, Coventry, UK
Max Planck Institute for Software Systems, Saarland Informatics Campus, Saarbrücken, Germany

Abstract

Given two weighted automata, we consider the problem of whether one is big-O of the other, i.e., if the weight of every finite word in the first is not greater than some constant multiple of the weight in the second.

We show that the problem is undecidable, even for the instantiation of weighted automata as labelled Markov chains. Moreover, even when it is known that one weighted automaton is big-O of another, the problem of finding or approximating the associated constant is also undecidable.

Our positive results show that the big-O problem is polynomial-time solvable for unambiguous automata, **coNP**-complete for unlabelled weighted automata (i.e., when the alphabet is a single character) and decidable, subject to Schanuel's conjecture, when the language is bounded (i.e., a subset of $w_1^* \dots w_m^*$ for some finite words $w_1, \dots, w_m$).

On labelled Markov chains, the problem can be restated as a ratio total variation distance, which, instead of finding the maximum difference between the probabilities of any two events, finds the maximum ratio between the probabilities of any two events. The problem is related to ϵ-differential privacy, for which the optimal constant of the big-O notation is exactly $\exp(\epsilon)$.

2012 ACM Subject Classification Theory of computation → Probabilistic computation

Keywords and phrases weighted automata, labelled Markov chains, probabilistic systems

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2020.41

Related Version A full version of the paper is available at `https://arxiv.org/abs/2007.07694`.

Funding *Dmitry Chistikov*: Supported in part by the Royal Society International Exchanges scheme (IEC\R2\170123).
Stefan Kiefer: Supported by a Royal Society Research Fellowship.
Andrzej S. Murawski: Supported by a Royal Society Leverhulme Trust Senior Research Fellowship and the International Exchanges Scheme (IE161701).
David Purser: Supported by the UK EPSRC Centre for Doctoral Training in Urban Science (EP/L016400/1) and in part by the Royal Society International Exchanges scheme (IEC\R2\170123).

Acknowledgements The authors would like to thank to Engel Lefaucheux, Joël Ouaknine, and James Worrell for discussions during the development of this work.

1 Introduction

Weighted automata over finite words are a well-known and powerful model of computation, a quantitative analogue of finite-state automata. Special cases of weighted automata include nondeterministic finite automata and labelled Markov chains, two standard formalisms for modelling systems and processes. Algorithms for analysis of weighted automata have been studied both in the early theory of computing and more recently by the infinite-state systems and algorithmic verification communities.

Given two weighted automata $\mathcal{A}$, $\mathcal{B}$ over an algebraic structure $(\mathcal{S}, +, \times)$, the equivalence problem asks whether the two associated functions $f_{\mathcal{A}}, f_{\mathcal{B}} \colon \Sigma^* \to \mathcal{S}$ are equal: $f_{\mathcal{A}}(w) = f_{\mathcal{B}}(w)$ for all finite words w over the alphabet Σ. Over the ring $(\mathbb{Q}, +, \times)$, equivalence is decidable in polynomial time by the results of Schützenberger [34] and Tzeng [38]; subsequently, fast parallel (**NC** and **RNC**) algorithms have been found for this problem [39, 20]. In contrast, for semirings the equivalence problem is hard: undecidable [21, 1] for the semiring $(\mathbb{Q}, \max, +)$ and **PSPACE**-hard [28] for the Boolean semiring (for which weighted automata are usual nondeterministic finite automata and equivalence is equality of recognized languages). Replacing $=$ with $\leq$ makes the problem harder: even for the ring $(\mathbb{Q}, +, \times)$ the question of whether $f_{\mathcal{A}}(w) \leq f_{\mathcal{B}}(w)$ for all $w \in \Sigma^*$ is undecidable – even if $f_{\mathcal{A}}$ is constant [31]. This problem subsumes the universality problem for (Rabin) probabilistic automata, yet another subclass of weighted automata (see, e.g., [12]).

In this paper, we introduce and study another natural problem, in which the ordering is relaxed from exact (in)equality to (in)equality to within a constant factor. Given $\mathcal{A}$ and $\mathcal{B}$ as above, is it true that there exists a constant $c > 0$ such that

$$f_{\mathcal{A}}(w) \leq c \cdot f_{\mathcal{B}}(w) \qquad \text{for all } w \in \Sigma^* \,?$$

Using standard mathematical notation, this condition asserts that $f_{\mathcal{A}}(w) = O(f_{\mathcal{B}}(w))$ as $|w| \to \infty$, and we refer to this problem as the *big-O* problem accordingly.[1] The *big-*Θ problem (which turns out to be computationally equivalent to the big-O problem), in line with the $\Theta(\cdot)$ notation in analysis of algorithms, asks whether $f_{\mathcal{A}} = O(f_{\mathcal{B}})$ and $f_{\mathcal{B}} = O(f_{\mathcal{A}})$.

We restrict our attention to the ring $(\mathbb{Q}, +, \times)$ and only consider *non-negative weighted automata*, i.e., those in which all transitions have non-negative weights. We remark that, even under this restriction, weighted automata still form a superclass of (Rabin) probabilistic automata, a non-trivial and rich model of computation. Our initial motivation to study the big-O problem came from yet another formalism, labelled Markov chains (LMCs). One can think of the semantics of LMCs as giving a probability distribution or subdistribution on the set of all finite words. LMCs, often under the name Hidden Markov Models, are widely employed in a diverse range of applications; in computer-aided verification, they are perhaps the most fundamental model for probabilistic systems, with model-checking tools such as Prism [22] or Storm [10] based on analyzing LMCs efficiently. All the results in our paper (including hardness results) hold for LMCs too. Our main findings are as follows.

- The big-O problem for non-negative WA and LMCs turns out to be **undecidable in general**, by a reduction from nonemptiness for probabilistic automata.
- For **unambiguous automata**, i.e., where every word has at most one accepting path, the big-O problem becomes decidable and can be solved in polynomial time.

[1] There also exists a related but slightly different definition of big-O; see Remark 12 for details on the corresponding version of our big-O problem.

- In the **unary case**, i.e., if the input alphabet Σ is a singleton, the big-O problem is also decidable and, in fact, complete for the complexity class **coNP**. Unary LMCs are a simple and pure probabilistic model of computation: they run in discrete time and can terminate at any step; the big-O problem refers to this termination probability in two LMCs (or two WA). Our upper bound argument refines an analysis of growth of entries in powers of non-negative matrices by Friedland and Schneider [33], and the lower bound is obtained by a reduction from unary NFA universality [37].
- In a more general **bounded case**, i.e., if the languages of all words w associated with non-zero weight are included in $w_1^* w_2^* \dots w_m^*$ for some finite words $w_1, \dots, w_m \in \Sigma^*$ (that is, are *bounded in the sense of Ginsburg and Spanier*; see [16, Chapter 5] and [17]), the big-O problem is decidable subject to Schanuel's conjecture. This is a well-known conjecture in transcendental number theory [23], which implies that the first-order theory of the real numbers with the exponential function is decidable [24]. Intuitively, our reliance on this conjecture is linked to the expressions for the growth rate in powers of non-negative matrices. These expressions are sums of terms of the form $\rho^n \cdot n^k$, where n is the length of a word, $k \in \mathbb{N}$, and ρ is an algebraic number. Our algorithms (however implicitly) need to compare for equality pairs of real numbers of the form $\log \rho_1 / \log \rho_2$, where ρ_i are algebraic, and it is an open problem in number theory whether there is an effective procedure for this task (the four exponentials conjecture asks whether two such ratios can ever be equal; see, e.g., Waldschmidt [40, Sections 1.3 and 1.4]).
  Bounded languages form a well-known subclass of regular languages. In fact, a regular (or even context-free) language L is bounded if and only if the number of words of length n in L is at most polynomial in n. All other regular languages have, in contrast, exponential growth rate (a fact rediscovered multiple times; see, e.g., references in Gawrychowski et al. [14]). Bounded languages have been studied from combinatorial and algorithmic points of view since the 1960s [17, 14], and have recently been used, e.g., in the analysis of quantitative information flow problems in computer security [27, 26]. In the context of labelled Markov chains, languages that are subsets of $a_1^* a_2^* \dots a_m^*$ (for individual letters $a_1, \dots, a_m \in \Sigma$) model consecutive arrival of m events in a discrete-time system. It is curious that natural decision problems for such simple systems can lead to intricate algorithmic questions in number theory at the border of decidability.

Further motivation and related work

In the labelled Markov chain setting, the big-O problem can be reformulated as a boundedness problem for the following function. For two LMCs $\mathcal{A}$ and $\mathcal{B}$, define the (asymmetric) *ratio variation function* by

$$r(\mathcal{A}, \mathcal{B}) = \sup_{E \subseteq \Sigma^*} (f_{\mathcal{A}}(E) / f_{\mathcal{B}}(E)),$$

where $f_{\mathcal{A}}(E)$ and $f_{\mathcal{B}}(E)$ denote the total probability mass associated with an arbitrary set of finite words $E \subseteq \Sigma^*$ in $\mathcal{A}$ and $\mathcal{B}$, respectively. Here we assume $\frac{0}{0} = 0$ and $\frac{x}{0} = \infty$ for $x > 0$. Observe that, because $\max(\frac{a}{b}, \frac{c}{d}) \geq \frac{a+c}{b+d}$ for $a, b, c, d \geq 0$, the supremum over $E \subseteq \Sigma^*$ can be replaced with supremum over $w \in \Sigma^*$. Consequently, the big-O problem for LMCs is equivalent to deciding whether $r(\mathcal{A}, \mathcal{B}) < \infty$.

Finding the value of r amounts to asking for the optimal (minimal) constant in the big-O notation. Further, one can consider a symmetric variant, the *ratio distance*: $rd(\mathcal{A}, \mathcal{B}) = \max\{r(\mathcal{A}, \mathcal{B}), r(\mathcal{B}, \mathcal{A})\}$, in an analogy with big-Θ. Now, rd is a ratio-oriented variant of the classic *total variation distance* tv, defined by $tv(\mathcal{A}, \mathcal{B}) = \sup_{E \subseteq \Sigma^*}(f_{\mathcal{A}}(E) - f_{\mathcal{B}}(E))$, which is

a well-established way of comparing two labelled Markov chains [5, 19]. We also consider the problem of approximating r (as well as rd) to a given precision and the problem of comparing it with a given constant (threshold problem), showing that both are undecidable.

The ratio distance rd is also equivalent to the exponential of the *multiplicative total variation distance* defined in [4, 36] in the context of differential privacy. Consider a system $\mathcal{M}$, modelled by a single labelled Markov chain, where output words are observable to the environment but we want to protect the privacy of the starting configuration. Let $R \subseteq Q \times Q$ be a symmetric relation, which relates the starting configurations intended to remain indistinguishable. Given $\epsilon \geq 0$, we say that $\mathcal{M}$ is ϵ-differentially private (with respect to R) if, for all $(s, s') \in R$, we have $f_s(E) \leq e^\epsilon \cdot f_{s'}(E)$ for every observable set of traces $E \subseteq \Sigma^*$ [11, 6]. **Here in the subscript of f and elsewhere, references to states s and s' replace references to LMCs/automata: $\mathcal{M}$ stays implicit, and we specify which state it is executed from.** Note that there exists such an ϵ if and only if $r(s, s') < \infty$ for all $(s, s') \in R$ or, equivalently, (the LMC $\mathcal{M}$ executed from) s is big-O of (the LMC $\mathcal{M}$ executed from) s' for all $(s, s') \in R$. In fact, the minimal such ϵ satisfies $e^\epsilon = \max_{(s,s')\in R} r(s, s')$, thus r captures the level of differential privacy between s and s'.

Our results show that even deciding whether the multiplicative total variation distance is finite or $+\infty$ is, in general, impossible. Likewise, it is undecidable whether a system modelled by a labelled Markov chain provides any degree of differential privacy, however low.

2 Preliminaries

▶ **Definition 1.** *A* weighted automaton $\mathcal{W}$ *over the* $(\mathbb{Q}, +, \times)$ *semi-ring is a 4-tuple* $\langle Q, \Sigma, M, F\rangle$*, where* Q *is a finite set of* states*,* Σ *is a finite* alphabet*,* $M : \Sigma \to \mathbb{Q}^{Q\times Q}$ *is a* transition weighting function*, and* $F \subseteq Q$ *is a set of* final states*. We consider only non-negative weighted automata, i.e.* $M(a)(q, q') \geq 0$ *for all* $a \in \Sigma$ *and* $q, q' \in Q$*.*

In complexity-theoretic arguments, we assume that each weight is given as a pair of integers (numerator and denominator) in binary. The description size is then the number of bits required to represent $\langle Q, \Sigma, M, F\rangle$, including the bit size of the weights.

Each weighted automaton defines functions $f_s : \Sigma^* \to \mathbb{R}$, where for all $s \in Q$

$$f_s(w) = \sum_{t\in F} (M(a_1) \times M(a_2) \times \cdots \times M(a_n))_{s,t} \qquad \text{for } w = a_1 a_2 \dots a_n \in \Sigma^*$$

and $A \times B$ is standard matrix multiplication. We refer to $f_s(w)$ as *the weight of* w *from state* s. Without loss of generality, a weighted automaton can have a single final state. If not, introduce a new unique final state t s.t. $M(a)(q, t) = \sum_{q'\in F} M(a)(q, q')$ for all $q \in Q$,$a \in \Sigma$.

▶ **Definition 2.** *We denote by* $\mathcal{L}_s(\mathcal{W})$ *the set of* $w \in \Sigma^*$ *with* $f_s(w) > 0$*, that is, with positive weight from* s*. Equivalently, this is the language of* $\mathcal{N}_s(\mathcal{W})$*, the non-deterministic finite automaton (NFA) formed from the same set of states (and final states) as* $\mathcal{W}$*, start state* s*, and transitions* $q \xrightarrow{a} q'$ *whenever* $M(a)(q, q') > 0$*.*

Given $s, s' \in Q$, we say that ***s is big-O of s'*** if there exists $C > 0$ such that $f_s(w) \leq C \cdot f_{s'}(w)$ for all $w \in \Sigma^*$. The paper studies the following problem.

▶ **Definition 3** (Big-O Problem)**.**

INPUT	*Weighted automaton* $\langle Q, \Sigma, M, F\rangle$ *and* $s, s' \in Q$
OUTPUT	*Is* s *big-O of* s'*?*

▶ Remark 4. One could consider whether s is big-Θ of s', defined as s is big-O of s' and s' is big-O of s; equivalently, whether $rd(s,s') < \infty$ for LMCs. We note that these two notions reduce to each other, justifying our consideration of only the big-O problem. There is an obvious reduction from big-Θ to big-O making two oracle calls (a Cook reduction), but this can be strengthened to a single call preserving the answer (a Karp reduction). This, however, requires at least two characters. In the other direction, one can ask if s big-O of s' using big-Θ by asking if a linear combination of s and s' is big-Θ of s'.

In the paper we also work with labelled Markov chains. In particular, they will appear in examples and hardness (including undecidability) arguments. As they are a special class of weighted automata, this will imply hardness (resp. undecidability) for weighted automata in general. On the other hand, our decidability results will be phrased using weighted automata, which makes them applicable to labelled Markov chains.

▶ **Definition 5.** *A* labelled Markov chain *(LMC) is a (non-negative) weighted automaton* $\langle Q, \Sigma, M, F\rangle$ *such that, for all* $q \in Q \setminus F$*, we have* $\sum_{q' \in Q} \sum_{a \in \Sigma} M(a)(q,q') = 1$ *and* $M(a)(q,q') = 0$ *for all for all* $a \in \Sigma, q \in F$ *and* $q' \in Q$*.*

Since final states have no outgoing transitions, w.l.o.g., one can assume a unique final state. For LMCs, the function f_s can be extended to a measure on the powerset of Σ^* by $f_s(E) = \sum_{w \in E} f_s(w)$, where $E \subseteq \Sigma^*$. The measure is a subdistribution: $\sum_{w \in \Sigma^*} f_s(w) \leq 1$.

We will also consider *unary* weighted automata, and similarly LMCs, where $|\Sigma| = 1$. Then we will often omit Σ on the understanding that $\Sigma = \{a\}$, and describe transitions with a single matrix $A = M(a)$ so that $f_s(a^n) = A^n_{s,t}$, where t is the unique final state. Note that $A^n_{s,t}$ stands for $(A^n)(s,t)$, and not $(A(s,t))^n$. Using the notation of regular expressions, we can write $\mathcal{L}_s(\mathcal{W}) \subseteq a^*$. It will turn out fruitful to consider several larger classes of languages:

▶ **Definition 6.** *Let* $L \subseteq \Sigma^*$*. L is* bounded *[17] if* $L \subseteq w_1^* w_2^* \cdots w_m^*$ *for some* $w_1, \ldots, w_m \in \Sigma^*$*. L is* letter-bounded *if* $L \subseteq a_1^* a_2^* \ldots a_m^*$ *for some* $a_1, \ldots, a_m \in \Sigma$*. L is* plus-letter-bounded *if* $L \subseteq a_1^+ a_2^+ \ldots a_m^+$ *for some* $a_1, \ldots, a_m \in \Sigma$*.*

In each case, if the language of an NFA is suitably bounded, one can extract a corresponding bounding regular expression [14].

3 Big-O, Threshold and Approximation problems are undecidable

We show that the big-O problem is undecidable. We also establish undecidability for several other problems related to computing and approximating the ratio variation distance. Recall that this corresponds to identifying the optimal constant for positive instances of the big-O problem or the level of differential privacy between two states in a labelled Markov chain.

▶ **Definition 7.** *The* asymmetric threshold problem *takes an LMC along with two states* s, s' *and a constant* θ*, and asks if* $r(s,s') \leq \theta$*. The variant under the promise of boundedness promises that* $r(s,s') < \infty$*. The strict variant of each problem replaces* $\leq$ *with* $<$*.*

The asymmetric additive approximation task *takes an LMC, two states* s, s' *and a constant* γ*, and asks for* x *such that* $|r(s,s') - x| \leq \gamma$*. The* asymmetric multiplicative approximation task *takes an LMC, two states* s, s' *and a constant* γ*, and asks for* x *such that* $1 - \gamma \leq \frac{x}{r(s,s')} \leq 1 + \gamma$*.*

In each case, the symmetric variant is obtained by replacing r *with* rd*.*

▶ **Theorem 8.**

- *The big-O problem is undecidable, even for LMCs.*
- *Each variant of the threshold problem (asymmetric/symmetric, non-strict/strict) is undecidable, even under the promise of boundedness.*
- *All variants of the approximation tasks (asymmetric/symmetric, additive/multiplicative) are unsolvable, even under the promise of boundedness.*

Probabilistic automata are similar to LMCs, except that $M(a)$ is stochastic for every a, rather than $\sum_{a\in\Sigma} M(a)$ being stochastic. Formally, a *probabilistic automaton* is a *non-negative weighted automaton* with a distinguished start state q_s such that $\sum_{q'\in Q} M(a)(q,q') = 1$ for all $q \in Q$ and $a \in \Sigma$. The problem Empty asks if $f_{q_s}(w) \leq \frac{1}{2}$ for all words w. It is known to be undecidable [31, 12].

Proof sketch of Theorem 8. We reduce from Empty. The construction creates two branches of a labelled Markov chain. The first simulates the probabilistic automaton using the original weights multiplied by a scalar ($\frac{1}{4}$ in the case $|\Sigma| = 2$). The other branch will process each letter from Σ with equal weight (also $\frac{1}{4}$ in an infinite loop). Consequently, if there is a word accepted with probability greater than $\frac{1}{2}$, the ratio between the two branches will be greater than 1. The construction will enable words to be processed repeatedly, so that the ratio can then be pumped unboundedly. Certain linear combinations of the branches enable a gap promise, entailing undecidability of the threshold and approximation tasks. ◀

▶ Remark. The classic *non-strict* threshold problem for the total variation distance (i.e. whether $tv(s,s') \leq \theta$) is known to be undecidable [19], like our distances. However, it is not known if its strict variant (i.e. whether $tv(s,s') < \theta$) is also undecidable. In contrast, in our case, both variants are undecidable. Further note that (additive) approximation of tv is possible [19, 5], but this is not the case for our distances r and rd.

▶ Remark. We have shown the undecidability of the big-O problem using the undecidability of the emptiness problem for probabilistic automata. Another proof of undecidability can be obtained using the Value-1 problem (shown to be undecidable in [15]): indeed the big-O problem and the Value-1 problem are interreducible. However, the reduction from big-O to Value-1 does not entail decidability for subclasses of weighted automata (such as those with bounded languages), as the image of these subclasses does not fall into the known decidable fragments of the Value-1 problem. Further details are available in the full version.

4 The LC condition

Towards decidability results, we identify a simple necessary (but insufficient) condition for s being big-O of s'.

▶ **Definition 9** (LC condition). *A weighted automaton $\mathcal{W} = \langle Q, \Sigma, M, F\rangle$ and $s, s' \in Q$ satisfy the* language containment condition *(LC) if for all words w with $f_s(w) > 0$ we also have $f_{s'}(w) > 0$. Equivalently, $\mathcal{L}_s(\mathcal{W}) \subseteq \mathcal{L}_{s'}(\mathcal{W})$.*

The condition can be verified by constructing NFA $\mathcal{N}_s(\mathcal{W}), \mathcal{N}_{s'}(\mathcal{W})$ that accept $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ respectively and verifying $\mathcal{L}(\mathcal{N}_s(\mathcal{W})) \subseteq \mathcal{L}(\mathcal{N}_{s'}(\mathcal{W}))$.

▶ Remark 10. Recall that NFA language containment is **NL**-complete if the automata are in fact deterministic, in **P** if they are unambiguous [8, Theorem 3], **coNP**-complete if they are unary [37] and **PSPACE**-complete in general [28]. In all cases this complexity level will match, or be lower than that for our respective algorithm for the big-O problem.

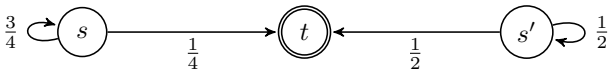

Figure 1 Unbounded ratio but language equivalent.

We observe that, if s is big-O of s', the LC condition must hold and so the LC condition is the first step in each of our verification routines. Example 11 shows that the condition alone is not sufficient to solve the big-O problem, because two states can admit the same set of words with non-zero weight, yet the weight ratios become unbounded.

▶ **Example 11.** Consider the unary automaton $\mathcal{W}$ in Figure 1. We have $\mathcal{L}_s(\mathcal{W}) = \mathcal{L}_{s'}(\mathcal{W}) = \{a^n \mid n \geq 1\}$, but $\frac{f_s(a^n)}{f_{s'}(a^n)} = \frac{(0.75)^{n-1} \cdot 0.25}{(0.5)^{n-1} \cdot 0.5} = 0.5 \cdot 1.5^{n-1} \xrightarrow[n\to\infty]{} \infty$.

▶ Remark 12. The original big-O notation on $f, g : \mathbb{N} \to \mathbb{N}$, states that f is $O(g)$ if $\exists C, k > 0$ $\forall n > k$ $f(n) \leq C\, g(n)$. Despite excluding finitely many points, when $g(n) \geq 1$, it is equivalent to $\exists C > 0$ $\forall n > 0$ $f(n) \leq C\, g(n)$ by taking C large enough to deal with the finite prefix.

In the paper, though, we formally consider s to not be big-O of s' if there exists even a single word w such that $f_s(w) > 0$ and $f_{s'}(w) = 0$. However, for weighted automata, we could amend our definition to "eventually big-O" as follows: $\exists C > 0, k > 0 : \forall w \in \Sigma^{\geq k}$ $f_s(w) \leq C \cdot f_{s'}(w)$.

The big-O problem reduces to its eventual variant by checking both the LC condition and the eventually big-O condition. Thus our undecidability (and hardness) results transfer to the eventually big-O problem. The eventually big-O problem can be solved via the big-O problem by "fixing" the LC condition through the addition of a branch from s' that accepts all appropriate words with very low probability. Further details are available in the full version.

4.1 Application: unambiguous weighted automata

In this section, we prove the first decidability result, that is, polynomial-time solvability in the unambiguous case. We say a weighted automaton $\mathcal{W}$ is *unambiguous from a state s* if every word has at most one accepting path in $\mathcal{N}_s(\mathcal{W})$.

▶ **Lemma 13.** *If a weighted automaton $\mathcal{W}$ is unambiguous from states s and s', the big-O problem is decidable in polynomial time.*

Proof sketch. We construct a product weighted automaton, with edge weights of the form $M'(a)((q_1, q_1'), (q_2, q_2')) = \frac{M(a)(q_1, q_2)}{M(a)(q_1', q_2')}$ and ask if there is a cycle on a path from (s, s') to (t, t) with weight > 1, which can be detected in polynomial time using a variation on the Bellman-Ford algorithm. ◀

Note the relevant behaviours are those on cycles – transitions which are taken at most once are of little significance to the big-O problem. Such transitions have at most a constant multiplicative effect on the ratio. This is the case whether or not the system is unambiguous.

5 The big-O problem for unary weighted automata is coNP-complete

In this section we show **coNP**-completeness in the unary case.

▶ **Theorem 14.** *The big-O problem for unary weighted automata is* **coNP***-complete. It is* **coNP***-hard even for unary labelled Markov chains.*

For the upper bound, our analysis will refine the analysis of the growth of powers of non-negative matrices of Friedland and Schneider [13, 33] which gives the asymptotic order of growth of $A^n_{s,t} + A^{n+1}_{s,t} + \cdots + A^{n+q}_{s,t} \approx \rho^n n^k$ for some ρ, k and q, which smooths over the periodic behaviour (see Theorem 18). Our results require a non-smoothed analysis, valid for each n. This isn't provided in [13, 33], where the smoothing forces the existence of a single limit – which we don't require. Our big-Θ lemma (Lemma 21) will accurately characterise the asymptotic behaviour of $A^n_{s,t}$ by exhibiting the correct value of ρ and k for every word.

5.1 Preliminaries

Let $\mathcal{W}$ be a unary non-negative weighted automaton with states Q, transition matrix A and a unique final state t. When we refer to a *path* in $\mathcal{W}$, we mean a path in the NFA of $\mathcal{W}$, i.e. paths only use transitions with non-zero weights and states on a path may repeat.

▶ **Definition 15.**

- *A state q can reach q' if there is a path from q to q'. In particular, any state q can always reach itself.*
- *A strongly connected component (SCC) $\varphi \subseteq Q$ is a maximal set of states such that for each $q, q' \in \varphi$, q can reach q'. We denote by $SCC(q)$ the SCC of state q and by A^φ, the $|\varphi| \times |\varphi|$ transition matrix of φ. Note every state is in a SCC, even if it is a singleton.*
- *The DAG of $\mathcal{W}$ is the directed acyclic graph of strongly connected components. Components φ, φ' are connected by an edge if there exist $q \in \varphi$ and $q' \in \varphi'$ with $A(q, q') > 0$.*
- *The spectral radius of an $m \times m$ matrix A is the largest absolute value of its eigenvalues. Recall the eigenvalues of A are $\{\lambda \in \mathbb{C} \mid$ exists vector $\vec{x} \in \mathbb{C}^m, \vec{x} \neq 0$ with $A\vec{x} = \lambda\vec{x}\}$. The spectral radius of φ, denoted by ρ_φ, is the spectral radius of A^φ. By $\rho(q)$ we denote the spectral radius of the SCC in which q is a member.*
- *We denote by T^φ the period of the SCC φ: the greatest common divisor of return times for some state $s \in \varphi$, i.e. $\gcd\{t \in \mathbb{N} \mid A^t(s, s) > 0\}$. It is known that any choice of state in the SCC gives the same value (see e.g. [35, Theorem 1.20]). If $A^\varphi = [0]$ then $T^\varphi = 0$.*
- *Let $\mathscr{P}(s, s')$ be the set of paths from the SCC of s to the SCC of s' in the DAG of $\mathcal{W}$. Thus a path $\pi \in \mathscr{P}(s, s')$ is a sequence of SCCs $\varphi_1, \ldots, \varphi_m$.*
- *$T(s, s')$, called the local period between s and s', is defined by $T(s, s') = \operatorname{lcm}_{\pi \in \mathscr{P}(s,s')} \gcd_{\varphi \in \pi} T^\varphi$.*
- *The spectral radius between states s and s', written $\rho(s, s')$, is the largest spectral radius of any SCC seen on a path from s to s': $\rho(s, s') = \max_{\pi \in \mathscr{P}(s,s')} \rho(\pi)$, where $\rho(\pi) = \max_{\varphi \in \pi} \rho_\varphi$ for $\pi \in \mathscr{P}(s, s')$.*
- *The following function captures the number of SCCs which attain the largest spectral radius on the path that has the most SCCs of maximal spectral radius. Let $k(s, s') = \max_{\pi \in \mathscr{P}(s,s')} k(\pi) - 1$, where, for $\pi \in \mathscr{P}(s, s')$, $k(\pi) = |\{\varphi \in \pi \mid \rho_\varphi = \rho(s, s')\}|$.*

▶ Remark 16. Since our weighted automata have rational weights, the spectral radius of an SCC is an algebraic number, as the absolute value of a root of a polynomial with rational coefficients. In general, an algebraic number $z \in \mathbb{A}$ can be represented by a tuple $(p_z, a, b, r) \in \mathbb{Q}[x] \times \mathbb{Q}^3$, where p_z is a polynomial over x and a, b, r specify an approximation to distinguish z from all other roots: z is the only root of $p_z(x)$ with $|z - (a + bi)| \leq r$. This representation, which admits standard operations (addition, multiplication, absolute value, (in)equality testing, etc.), can be found in polynomial time (see, e.g. [29]). Henceforth, when we refer to the spectral radius we will implicitly mean representation in this form.

The asymptotic behaviours of weighted automata will be characterised using (ρ, k)-pairs:

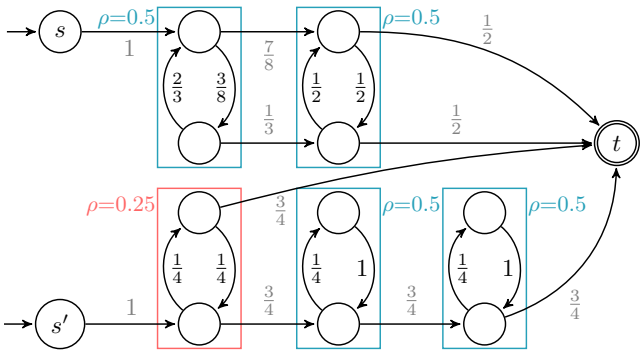

Figure 2 Different rates for different phases.

▶ **Definition 17.** *A (ρ, k)-pair is an element of $\mathbb{R}\times\mathbb{N}$. The ordering on $\mathbb{R}\times\mathbb{N}$ is lexicographic, i.e.* $(\rho_1, k_1) \leq (\rho_2, k_2) \iff \rho_1 < \rho_2 \vee (\rho_1 = \rho_2 \wedge k_1 \leq k_2)$.

Friedland and Schneider [13, 33] essentially use (ρ, k)-pairs to show the asymptotic behaviour of the powers of non-negative matrices. In particular they find the asymptotic behaviour of the sum of several $A^n_{s,s'}$, smoothing the periodic behaviour of the matrix.

▶ **Theorem 18** (Friedland and Schneider [13, 33])**.** *Let A be an $m \times m$ non-negative matrix, inducing a unary weighted automaton $\mathcal{W}$ with states $Q = \{1, \ldots, m\}$. Given $s, t \in Q$, let $B^n_{s,t} = A^n_{s,t} + A^{n+1}_{s,t} + \cdots + A^{n+T(s,t)-1}_{s,t}$. Then $\lim_{n\to\infty} \frac{B^n_{s,t}}{\rho(s,t)^n n^{k(s,t)}} = c, \quad 0 < c < \infty$.*

In the case where the local period is 1 ($T(s,t) = T(s',t) = 1$), Theorem 18 can already be used to solve the big-O problem (in particular if the matrix A is aperiodic). In this case $A^n_{s,t} = B^n_{s,t} = \Theta(\rho(s,t)^n n^{k(s,t)})$. Then to establish that s is big-O of s' we check that the language containment condition holds and that $(\rho(s,t), k(s,t)) \leq (\rho(s',t), k(s',t))$. However, this is not sufficient if the local period is not 1.

▶ **Example 19.** Consider the chains shown in Figure 2 with local period 2. The behaviour for $n \geq 3$ is $A^n_{s,t} = \Theta(0.5^n n)$ and $A^n_{s',t} = \Theta(0.25^n)$ when n is odd and $A^n_{s',t} = \Theta(0.5^n n)$ when n is even. However, Theorem 18 tells us $B^n_{s,t} = \Theta(0.5^n n)$ and $B^n_{s',t} = \Theta(0.5^n n)$ suggesting the ratio is bounded, but in fact s is not big-O s' (although s' is big-O of s) because $\frac{A^{2n+1}_{s,t}}{A^{2n+1}_{s',t}} \xrightarrow[n\to\infty]{} \infty$.

5.2 Upper bound: The unary big-O problem is in coNP

Let $\mathcal{W}$ be a unary weighted automaton and suppose we are asked whether s is big-O of s'. We assume w.l.o.g. (a) that there is a unique final state t with no outgoing transitions, and (b) that s, s' do not appear on any cycle (if this is not the case, copies of s, s' and their transitions can be taken).

Next we define a "degree function", which captures the asymptotic behaviour of each word a^n by a (ρ, k)-pair, capturing the exponential and polynomial behaviours respectively.

▶ **Definition 20.** *Given a unary weighted automaton $\mathcal{W}$, let $d_{s,t} : \mathbb{N} \to \mathbb{R}\times\mathbb{N}$ be defined by $d_{s,t}(n) = (\rho, k)$, where:*

- *ρ is the largest spectral radius of any vertex visited on any path of length n from s to t;*
- *the path from s to t that visits the most SCCs of spectral radius ρ visits $k+1$ such SCCs;*
- *if there is no length-n path from s to t, then (ρ, k)=$(0, 0)$.*

Let $s, t \in Q$ be fixed. We are now ready to state the key technical lemma of this subsection (cf. Theorem 18, Friedland and Schneider [13, 33]), where we assume the functions $\rho(n), k(n)$, defined by $d_{s,t}(n) = (\rho(n), k(n))$.

▶ **Lemma 21** (The big-Θ lemma). *There exist $c, C > 0$ such that, for every $n > |Q|$,*

$$c \cdot \rho(n)^n n^{k(n)} \leq A^n_{s,t} \leq C \cdot \rho(n)^n n^{k(n)}.$$

The set of *admissible* (ρ, k)-pairs is the image of $d_{s,t}$. Observe that this set is finite and of size at most $|Q|^2$: there can be no more than $|Q|$ values of ρ (if at worst each state were its own SCC) and the value of k is also bounded by the number of SCCs and thus $|Q|$.

We next define the (ρ, k)-annotated version of $\mathcal{W}$, i.e. in each state we record the relevant value of (ρ, k) corresponding to the current run to the state.

▶ **Definition 22** (The weighted automaton $\mathcal{W}^\dagger$). *Given $\mathcal{W} = \langle Q, \Sigma, A, \{t\}\rangle$ and $s \in Q$, the weighted automaton $\mathcal{W}^\dagger$ has states of the form (q, ρ, k) for all $q \in Q$ and all admissible (ρ, k)-pairs, the same Σ and no final states. For every transition $q \xrightarrow{p} q'$ from $\mathcal{W}$ denoting $A(q, q') = p$, include the following transition in $\mathcal{W}^\dagger$ for each admissible (ρ, k):*

- $(q, \rho, k) \xrightarrow{p} (q', \rho, k)$ *if* $SCC(q)$=$SCC(q')$,
- $(q, \rho, k) \xrightarrow{p} (q', \rho, k+1)$ *if* $SCC(q)\neq SCC(q')$ *and* $\rho = \rho(q')$,
- $(q, \rho, k) \xrightarrow{p} (q', \rho, k)$ *if* $SCC(q)\neq SCC(q')$ *and* $\rho > \rho(q')$,
- $(q, \rho, k) \xrightarrow{p} (q', \rho(q'), 0)$ *if* $SCC(q)\neq SCC(q')$ *and* $\rho(q') > \rho$.

$\mathcal{W}^\dagger$ is constructable in polynomial time given $\mathcal{W}$. Indeed, the spectral radii of all SCCs can be computed and compared to each other in time polynomial in the size of $\mathcal{W}$ (see Remark 16).

For the following lemma, recall the language containment (LC) condition from Definition 9 and the ordering on (ρ, k)-pairs from Definition 17.

▶ **Lemma 23.** *A state s is big-O of s' if and only if the LC condition holds and, for all but finitely many $n \in \mathbb{N}$, we have $d_{s,t}(n) \leq d_{s',t}(n)$.*

Proof sketch. Whenever $d_{s,t}(n) \leq d_{s',t}(n)$, by Lemma 21, we have $f_s(a^n) \leq (\frac{C}{c}(\frac{\rho}{\rho'})^n n^{k-k'}) \cdot f_{s'}(a^n)$, in which case either $d_{s,t}(n) = d_{s',t}(n)$ and $(\frac{\rho}{\rho'})^n n^{k-k'} = 1$ or $\lim_{n\to\infty}(\frac{\rho}{\rho'})^n n^{k-k'} = 0$ and so $(\frac{\rho}{\rho'})^n n^{k-k'} \leq 1$ for all but finitely many n.

However, whenever $d_{s,t}(n) > d_{s',t}(n)$, Lemma 21 yields $f_s(a^n) \geq (\frac{c}{C}(\frac{\rho}{\rho'})^n n^{k-k'}) \cdot f_{s'}(a^n)$ but then $\lim_{n\to\infty}(\frac{\rho}{\rho'})^n n^{k-k'} = \infty$. ◀

We are going to use the characterisation from Lemma 23 to prove Theorem 14. As already discussed, the LC condition can be checked via NFA inclusion testing. To tackle the "for all but finitely many ..." condition, we introduce the concept of eventual inclusion.

▶ **Definition 24.** *Given sets A, B, we say A is eventually included in B, written $A \subsetneqq B$, if and only if $A \setminus B$ is finite.*

The next three lemmas relate deciding the big-O problem using the characterisation of Lemma 23 to eventual inclusion.

▶ **Lemma 25.** *Given unary NFAs $\mathcal{N}_1, \mathcal{N}_2$, the problem $\mathcal{L}(\mathcal{N}_1) \subsetneqq \mathcal{L}(\mathcal{N}_2)$ is in* **coNP**.

▶ **Lemma 26.** *Suppose $d_1, d_2 : \mathbb{N} \to X$, with $(X, \leq)$ a finite total order. Then $d_1(n) \leq d_2(n)$ for all but finitely many n if and only if $\{n \mid d_1(n) \geq x\} \subsetneqq \{n \mid d_2(n) \geq x\}$ for all $x \in X$.*

▶ **Lemma 27.** *Given a unary weighted automaton $\mathcal{W}$, the associated problem whether $d_{s,t}(n) \leq d_{s',t}(n)$ for all but finitely many $n \in \mathbb{N}$ is in* **coNP**.

Proof. Given an admissible pair $x = (\rho, k)$, we construct an NFA $\mathcal{N}_{s,x}$ accepting $\{a^n \mid d_{s,t}(n) \geq x\}$ (similarly $\mathcal{N}_{s',x}$ for s'), by taking the NFA $\mathcal{N}_s(\mathcal{W}^\dagger)$ (Definitions 2, 22) with a suitable choice of accepting states. Recall that states in $\mathcal{W}^\dagger$ are of the form (q, ρ', k'), where q is a state from $\mathcal{W}$ and (ρ', k') is admissible. If we designate states (t, ρ', k') with $(\rho', k') \geq x$ as accepting, it will accept $\{a^n \mid d_{s,t}(n) \geq x\}$. This is a polynomial-time construction.

Then, by Lemma 26, the problem whether $d_{s,t}(n) \leq d_{s',t}(n)$ for all but finitely many $n \in \mathbb{N}$ is equivalent to $\mathcal{L}(\mathcal{N}_{s,x}) \subsetneq \mathcal{L}(\mathcal{N}_{s',x})$ for all admissible x. As there are at most $|Q|^2$ values of x and each can be verified non-deterministically in **coNP**, it suffices to show that $\mathcal{L}(\mathcal{N}_{s,x}) \subsetneq \mathcal{L}(\mathcal{N}_{s',x})$ is in **coNP** for each x. This is the case by Lemma 25. ◀

Remark 10 and Lemma 27 together complete the upper bound result for Theorem 14.

▶ Remark. Lemma 26 may appear simpler using $\{n \mid f_1(n) = x\} \subsetneq \{n \mid f_2(n) \geq x\}$. However, it does not seem possible to construct an NFA for $\{a^n \mid d_{s,t}(n) = x\}$ in polynomial time. Taking just (t, ρ, k) as accepting would not be correct, as there could be paths of the same length ending in (t, ρ', k') with $(\rho', k') > (\rho, k)$. Using $\geq$ instead of $=$ avoids this problem.

▶ Remark. An alternative approach for obtaining an upper bound could be to compute the Jordan normal form of the transition matrix and consider its powers. Instead of the interplay of strongly connected components in the transition graph, we would need to consider linear combinations of the nth powers of complex numbers (such as roots of unity). It is not clear this algebraic approach leads to a representation more convenient for our purposes.

5.3 coNP-hardness for unary LMC

Given a unary NFA $\mathcal{N}$, the *NFA universality problem* asks if $\mathcal{L}(\mathcal{N}) = \{a^n \mid n \in \mathbb{N}\}$. This problem is **coNP**-complete [37]. We exhibit a polynomial-time reduction from (a variant of) the unary universality problem to the big-O problem on unary Markov chains. Further details are available in the full version.

6 Decidability for weighted automata with bounded languages

In this section we consider the big-O problem for a weighted automaton $\mathcal{W}$ and states s, s' such that $\mathcal{L}_s(\mathcal{W})$, $\mathcal{L}_{s'}(\mathcal{W})$ are bounded. Throughout the section, we assume that the LC condition has already been checked, i.e. $\mathcal{L}_s(\mathcal{W}) \subseteq \mathcal{L}_{s'}(\mathcal{W})$. We will show that the problem is conditionally decidable, subject to Schanuel's conjecture.

Logical theories of arithmetic and Schanuel's conjecture. In *first-order logical theories of arithmetic,* variables denote numbers (from $\mathbb{Z}$ or $\mathbb{R}$, as appropriate), and atomic predicates are equalities and inequalities between terms built from variables and function symbols. Nullary function symbols are constants, always from $\mathbb{Z}$. If binary addition and multiplication are available, then:

- for $\mathbb{R}$ we obtain the first-order theory of the reals, where the truth value of sentences is decidable due to the celebrated Tarski–Seidenberg theorem [3, Chapter 11 and Theorem 2.77];
- for $\mathbb{Z}$, the first-order theory of the integers is, in contrast, undecidable (see, e.g, [32]).

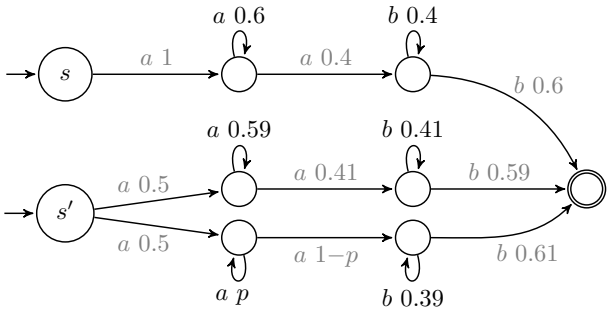

Figure 3 Relative orderings are the same, but the boundedness question is different.

In the case of $\mathbb{R}$, adding the unary symbol for the exponential function $x \mapsto e^x$, leads to *the first-order theory of the real numbers with exponential function* ($\mathrm{Th}(\mathbb{R}_{\mathsf{exp}})$). Logarithms base 2, for example, are easily expressible in $\mathrm{Th}(\mathbb{R}_{\mathsf{exp}})$. The decidability of $\mathrm{Th}(\mathbb{R}_{\mathsf{exp}})$ is an open problem and hinges upon Schanuel's conjecture [24].

Schanuel's conjecture [23] is a unifying conjecture of transcendental number theory, saying that for all $z_1, \ldots, z_n \in \mathbb{C}$ linearly independent over $\mathbb{Q}$ the field extension $\mathbb{Q}(z_1, \ldots, z_n, e^{z_1}, \ldots, e^{z_n})$ has transcendence degree at least n over $\mathbb{Q}$, meaning that for some $S \subseteq \{z_1, \ldots, z_n, e^{z_1}, \ldots, e^{z_n}\}$ of cardinality n, say $S = \{s_1, \ldots, s_n\}$, the only polynomial p over $\mathbb{Q}$ satisfying $p(s_1, \ldots, s_n) = 0$ is $p \equiv 0$. See, e.g., Waldschmidt's book [40, Section 1.4] for further context. If indeed true, this conjecture would generalise several known results, including the Lindemann–Weierstrass theorem and Baker's theorem, and would entail the decidability of $\mathrm{Th}(\mathbb{R}_{\mathsf{exp}})$. Our work follows an exciting line of research that reduces problems from verification [9, 25], linear dynamical systems [2, 7], and symbolic computation [18] to the decision problem for $\mathrm{Th}(\mathbb{R}_{\mathsf{exp}})$.

▶ **Theorem 28.** *Given a weighted automaton $\mathcal{W} = \langle Q, \Sigma, M, F \rangle$, $s, s' \in Q$, with $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ bounded, it is decidable whether s is big-O of s', subject to Schanuel's conjecture.*

In the unary case, it was sufficient to consider the *relative order* between spectral radii, with careful handling of the periodic behaviour. This approach is insufficient in the bounded case. Example 29 highlights that the actual values of the spectral radii have to be examined.

▶ **Example 29** (Relative orderings are insufficient)**.** Consider the LMC in Figure 3, with $0.61 \leq p \leq 0.62$. We have $f_s(a^m b^n) = \Theta(0.6^m 0.4^n)$ and $f_{s'}(a^m b^n) = \Theta(p^m 0.39^n + 0.59^m 0.41^n)$. Note that neither $0.59^m 0.41^n$ nor $p^m 0.39^n$ dominate, nor are dominated by, $0.6^m 0.4^n$ for any value of $0.61 \leq p \leq 0.62$. That is, there are values of m, n where $0.59^m 0.41^n \gg 0.6^m 0.4^n$ (in particular large n) and values of m, n where $0.59^m 0.41^n \ll 0.6^m 0.4^n$ (in particular large m); similarly for $p^m 0.39^n$ vs $0.6^m 0.4^n$ (but the cases in which n or m needs to be large are swapped). However, the big-O status can be different for different values of $p \in [0.61, 0.62]$, despite the same relative ordering between spectral radii. When $p = 0.62$, the ratio turns out to be bounded: $\frac{f_s(a^m b^n)}{f_{s'}(a^m b^n)} \leq \frac{1600}{1579}$ for all m, n (in particular, maximal at $m = n = 0$). In contrast, when $p = 0.61$, we have $\frac{f_s(a^m b^{0.66m})}{f_{s'}(a^m b^{0.66m})} \xrightarrow[m \to \infty]{} \infty$.

We first prove Theorem 28 for the plus-letter-bounded case, which is the most technically involved; the other bounded cases will be reduced to it. In the plus-letter-bounded case, we will characterise the behaviour of such automata, generalising (ρ, k)-pairs of the unary case. We will need to rely upon the first-order theory of the reals with exponentials to compare these behaviours.

6.1 The plus-letter-bounded case

We assume $\mathcal{L}_{s'}(\mathcal{W}) \subseteq a_1^+ \cdots a_m^+$, where $a_1, \cdots, a_n \in \Sigma$ and because the LC condition holds, we also have $\mathcal{L}_s(\mathcal{W}) \subseteq a_1^+ \cdots a_m^+$. In the plus-letter-bounded cases, without loss of generality, we assume $a_i \neq a_j$ for $i \neq j$. Then any word $w = a_1^{n_1} \dots a_m^{n_m}$ is uniquely specified by a vector $(n_1, \dots, n_m) \in \mathbb{N}_{>0}^m$, where n_i is the number of a_i's in w.

Like in Definition 20, we define a degree function d, which will be used to study the asymptotic behaviour of words. This time we will associate a separate (ρ, k) pair to each of the m characters and, consequently, words will induce sequences of the form $(\rho_1, k_1) \cdots (\rho_m, k_m)$.

Further, as there may be multiple, incomparable behaviours, words will induce sets of such sequences, i.e. $d\colon \mathbb{N}^m \to \mathcal{P}((\mathbb{R} \times \mathbb{N})^m)$. For the sake of comparisons, it will be convenient to focus on maximal elements with respect to the pointwise order on $(\mathbb{R} \times \mathbb{N})^m$, written $\leq$, where the lexicographic order (recall Definition 17) is used to compare elements of $\mathbb{R} \times \mathbb{N}$.

Recall Lemma 21 does not capture the asymptotics when $n \leq |Q|$. In the unary case this is inconsequential as small words are covered by the *finitely many* exceptions and the LC condition. However, here, a small number of one character may be used to enable access to a particular part of the automaton in another character. For this case, we introduce a new number $\delta = \frac{1}{2} \min_{\varphi : \rho_\varphi > 0} \rho_\varphi$ which is strictly smaller than the spectral radius of every non-zero SCC (so will not dominate with the partial order), but non-zero.

▶ **Definition 30.** *Let $\hat{\rho} = (\rho_1, k_1), \cdots, (\rho_m, k_m) \in (\mathbb{R} \times \mathbb{N})^m$. An $a_1^{n_1} a_2^{n_2} \dots a_m^{n_m}$-labelled path from s (to the final state) is* compatible with $\hat{\rho}$ *if, for each $i = 1, \dots, m$, it visits $k_i + 1$ SCCs with spectral radius ρ_i while reading a_i, unless the path visits only singletons with no loops, in which case $(\rho_i, k_i) = (\delta, 0)$. The notation $(\rho, k) \in \hat{\rho}$ is used for "(ρ, k) is an element of $\hat{\rho}$".*

▶ **Definition 31.** *Let $d_s : \mathbb{N}^m \to \mathcal{P}((\mathbb{R} \times \mathbb{N})^m)$ be s.t.: $\hat{\rho} \in d_s(n_1, \dots, n_m)$ if and only if*

(1) *there exists an $a_1^{n_1} a_2^{n_2} \dots a_m^{n_m}$-labelled path from s to the final state compatible with $\hat{\rho}$, and*

(2) *for every $a_1^{n_1} a_2^{n_2} \dots a_m^{n_m}$-labelled path from s compatible with $\hat{\sigma}$ s.t. $\hat{\rho} \leq \hat{\sigma}$, we have $\hat{\rho} = \hat{\sigma}$.*

Observe that $\hat{\rho}$ may range over at most $|Q|^{2m}$ possible values. We write $\mathcal{D}$ for the set containing them, so that $d_s : \mathbb{N}^m \to \mathcal{P}(\mathcal{D})$. In this extended setting, the big-Θ lemma (Lemma 21) may be generalised as follows.

▶ **Lemma 32.** *Denote $z(n_1, \cdots, n_m) = \sum_{\hat{\rho} \in d_s(n_1, \dots, n_m)} \prod_{(\rho_i, k_i) \in \hat{\rho}} \rho_i^{n_i} \cdot n_i^{k_i}$. There exist $c, C > 0$ such that for all $n_1 \dots, n_m \in \mathbb{N}$:*

$$c \cdot z(n_1, \cdots, n_m) \leq f_s(a_1^{n_1} a_2^{n_2} \dots a_m^{n_m}) \leq C \cdot z(n_1, \cdots, n_m).$$

The following lemma provides the key characterisation of negative instances of the big-O problem, in the plus-letter-bounded case and assuming the LC condition. Here and below, we write $n(t)$ to refer to the the tth vector in a sequence $n\colon \mathbb{N} \to \mathbb{N}^m$.

▶ **Lemma 33** (Main lemma). *Assume $\mathcal{L}_s(\mathcal{W}) \subseteq \mathcal{L}_{s'}(\mathcal{W})$. Then s is not big-O of s' if and only if there exists a sequence $n\colon \mathbb{N} \to \mathbb{N}^m$ and $X \in \mathcal{D}$, $\mathcal{Y} \subseteq \mathcal{D}$ such that*

(a) *$X \in d_s(n(t))$ and $\mathcal{Y} = d_{s'}(n(t))$ for all t, and*

(b) *for all $j \in h_{\mathcal{Y}}$, the sequence n satisfies*

$$\sum_{i=1}^{m} \alpha_{j,i} n(t)_i + p_{j,i} \log n(t)_i \xrightarrow[t \to \infty]{} -\infty,$$

where $h_{\mathcal{Y}} \subseteq \{1, \dots, |\mathcal{Y}|\}$, $\alpha_{j,i} \in \mathbb{R}$, $p_{j,i} \in \mathbb{Z}$ $(1 \leq i \leq m)$ are uniquely determined by X and $\mathcal{Y}$ (in a way detailed below), $h_{\mathcal{Y}}$ and $p_{j,i}$'s are effectively computable and $\alpha_{j,i}$'s are first-order expressible (with exponential function).

Proof. Observe that then s is *not* big-O of s' iff there exists an infinite sequence of words such that, for all $C > 0$, the sequence contains a word w such that $\frac{f_s(w)}{f_{s'}(w)} > C$. Thanks to Lemma 32, this is equivalent to the existence of a sequence $n : \mathbb{N} \to \mathbb{N}^m$ such that

$$\frac{\displaystyle\sum_{X \in d_s(n(t)_1,\ldots,n(t)_m)} \prod_{(\rho_i,k_i)\in X} \rho_i^{n(t)_i} \cdot n(t)_i^{k_i}}{\displaystyle\sum_{Y \in d_{s'}(n(t)_1,\ldots,n(t)_m)} \prod_{(\sigma_i,\ell_i)\in Y} \sigma_i^{n(t)_i} \cdot n(t)_i^{\ell_i}} \xrightarrow[t\to\infty]{} \infty,$$

where $n(t)_i$ denotes the ith component of $n(t)$. Since there are finitely many possible values of d_s and $d_{s'}$, it suffices to look for sequences n such that $d_s(n(t))$ and $d_{s'}(n(t))$ are fixed. Further, because of the sum in the numerator, only one $X \in \mathcal{X}$ is required such that $X \in d_s(n_1, \ldots, n_m)$. Thus, we need to determine whether there exist $X \in \mathcal{D}$, $\mathcal{Y} \subseteq \mathcal{D}$ and $n : \mathbb{N} \to \mathbb{N}^m$ such that $X \in d_s(n(t))$, $d_{s'}(n(t)) = \mathcal{Y}$ (for all t) and

$$\frac{\prod_{i=1}^m \rho_i^{n(t)_i} \cdot n(t)_i^{k_i}}{\sum_{j=1}^{h_\mathcal{Y}} \prod_{i=1}^m \sigma_{ji}^{n(t)_i} \cdot n(t)_i^{\ell_{ji}}} \xrightarrow[t\to\infty]{} \infty.$$

where $X = (\rho_1, k_1) \cdots (\rho_m, k_m)$, $\mathcal{Y} = \{Y_1, \cdots, Y_{|\mathcal{Y}|}\}$, and $Y_j = (\sigma_{j1}, \ell_{j1}) \cdots (\sigma_{jm}, \ell_{jm})$ ($1 \leq j \leq |\mathcal{Y}|$). Taking the reciprocal and requiring each of the summands to go to zero, we obtain

$$\frac{\prod_{i=1}^m \sigma_{ji}^{n(t)_i} \cdot n(t)_i^{\ell_{ji}}}{\prod_{i=1}^m \rho_i^{n(t)_i} \cdot n(t)_i^{k_i}} = \prod_{i=1}^m \left(\frac{\sigma_{ji}}{\rho_i}\right)^{n(t)_i} n(t)_i^{\ell_{ji}-k_i} \xrightarrow[t\to\infty]{} 0 \qquad \text{for all } 1 \leq j \leq |\mathcal{Y}|.$$

If we take logarithms, letting $\alpha_{j,i} = \log(\frac{\sigma_{ji}}{\rho_i})$ and $p_{j,i} = \ell_{ji} - k_i$, we get

$$\sum_{i=1}^m \alpha_{j,i} n(t)_i + p_{j,i} \log n(t)_i \xrightarrow[t\to\infty]{} -\infty$$

for all j in $h_\mathcal{Y} = \{1 \leq j \leq |\mathcal{Y}| \mid \sigma_{ji} > 0 \text{ for all } 1 \leq i \leq m\}$.

The number $\alpha_{j,i}$ is the logarithm of the ratio of two algebraic numbers, which are not given explicitly. However, they admit an unambiguous, first-order expressible characterisation (see Remark 16). The logarithm is encoded using the exponential function: $\log(z)$ is $\exists x \in \mathbb{R} : \exp(x) = z$. ◀

Lemma 33 identifies violation of the big-O property using two conditions. In the remainder of this subsection we will handle Condition (a) using automata-theoretic tools (the Parikh theorem and semi-linear sets) and Condition (b) using logics. In summary, the characterisation of Lemma 33 will be expressed in the first-order theory of the reals with exponentiation, which is decidable subject to Schanuel's conjecture.

Condition (a) via automata

It turns out that sequences n satisfying Condition (a) in Lemma 33 can be captured by a finite automaton. In more detail, for any $X \in \mathcal{D}$, there exists an automaton $\mathcal{N}_X^s$ such that $\mathcal{L}(\mathcal{N}_X^s) = \{a_1^{n_1} \cdots a_m^{n_m} \mid X \in d_s(n_1, \cdots, n_m)\}$. For any $\mathcal{Y} \subseteq \mathcal{D}$, there exists an automaton $\mathcal{N}_\mathcal{Y}^s$ such that $\mathcal{L}(\mathcal{N}_\mathcal{Y}^s) = \{a_1^{n_1} \cdots a_m^{n_m} \mid d_s(n_1, \cdots, n_m) = \mathcal{Y}\}$. The relevant automaton capturing X and $\mathcal{Y}$ is then found by taking the intersection of $\mathcal{L}(\mathcal{N}_X^s)$ and $\mathcal{L}(\mathcal{N}_\mathcal{Y}^{s'})$.

▶ **Lemma 34.** *For any $X \in \mathcal{D}$ and $\mathcal{Y} \subseteq \mathcal{D}$, there exists an automaton $\mathcal{N}_{X,\mathcal{Y}}$ such that $\mathcal{L}(\mathcal{N}_{X,\mathcal{Y}}) = \{a_1^{n_1} \cdots a_m^{n_m} \mid X \in d_s(n_1, \cdots, n_m),\ \mathcal{Y} = d_{s'}(n_1, \cdots, n_m)\}$.*

Because of our $a_i \neq a_j$ assumption, the vector $(n_1, \cdots, n_m)$ indicates the number of occurrences of each character. The set of such vectors derived from the language of an automaton is known as the Parikh image of this language [30]. It is well known that the Parikh image of an NFA is a semi-linear set, i.e. a finite union of linear sets (a linear set has the form $\{\vec{b} + \lambda_1 \vec{r^1} + \cdots + \lambda_s \vec{r^s} \mid \lambda_1, \ldots, \lambda_s \in \mathbb{N}\}$, where $\vec{b} \in \mathbb{N}^m$ is the base vector and $\vec{r^1}, \cdots \vec{r^s} \in \mathbb{N}^m$ are called period vectors). However, since $\mathcal{L}(\mathcal{N}_{X,\mathcal{Y}}) \subseteq a_1^+ a_2^+ \ldots a_m^+$, the linear sets are of a very particular form, where each $\vec{r^i}$ is a constant multiple of the ith unit vector.

▶ **Lemma 35.** *The language of $\mathcal{N}_{X,\mathcal{Y}}$ can be effectively decomposed as $\mathcal{L}(\mathcal{N}_{X,\mathcal{Y}}) = \bigcup_{k=1}^{S_{X,\mathcal{Y}}} \mathcal{L}_k$, where $\mathcal{L}_k = \left\{ a_1^{b_{k1}+r_{k1}\lambda_1} \cdots a_m^{b_{km}+r_{km}\lambda_m} \mid \lambda_1, \cdots, \lambda_m \in \mathbb{N} \right\}$, $S_{X,\mathcal{Y}} \in \mathbb{N}$ and $b_{ki}, r_{ki} \in \mathbb{N}$ $(1 \leq k \leq S_{X,\mathcal{Y}},\ 1 \leq i \leq m)$.*

Lemma 35 captures Condition (a) of Lemma 33 precisely.

Condition (b) via logic

With Lemma 35 in place, we now move on to add Condition (b) to the existing machinery. In fact, the logical formulae in the following lemmas will express the conjunction of both conditions of Lemma 33.

▶ **Lemma 36.** *Assume $\mathcal{L}_s(\mathcal{W}) \subseteq \mathcal{L}_{s'}(\mathcal{W})$. Then s is not big-O of s' if and only if there exists $X \in \mathcal{D}$, $\mathcal{Y} \subseteq \mathcal{D}$, $1 \leq k \leq S_{X,\mathcal{Y}}$ such that*

$$\forall C < 0\ \exists \vec{\lambda} \in \mathbb{N}^m \qquad \bigwedge_{j \in h_{\mathcal{Y}}} \sum_{i=1}^{m} \alpha_{j,i}(b_{ki} + r_{ki}\lambda_i) + p_{j,i} \log(b_{ki} + r_{ki}\lambda_i) < C,$$

where $h_{\mathcal{Y}}, \alpha_{j,i}, p_{j,i}$ (resp. b_{ki}, r_{ki}) satisfy the same conditions as in Lemma 33 (resp. 35).

Note that the formula of Lemma 36 uses quantification over natural numbers. Our next step will be to replace integer variables with real variables. In other words, we will obtain an equivalent condition in the first-order theory of the reals with exponentiation, as follows.

▶ **Lemma 37.** *Assume $\mathcal{L}_s(\mathcal{W}) \subseteq \mathcal{L}_{s'}(\mathcal{W})$. Then s is not big-O of s' if and only if there exist $X \in \mathcal{D}$, $\mathcal{Y} \subseteq \mathcal{D}$, $1 \leq k \leq S_{X,\mathcal{Y}}$ and $U \subseteq \{i \in \{1, \cdots, m\} \mid r_{ki} > 0\}$ such that*

$$\forall C < 0\ \exists \vec{x} \in \mathbb{R}^{|U|}_{\geq B_k} \qquad \bigwedge_{j \in h_{\mathcal{Y}}} \sum_{i \in U} \alpha_{j,i} r_{ki} x_i + p_{j,i} \log(x_i) < C,$$

where $B_k = \max_i b_{ki}$ and $h_{\mathcal{Y}}, \alpha_{j,i}, p_{j,i}, b_{ki}, r_{ki}$ are as in Lemma 36.

Proof Sketch. Compare the logical characterisation in Lemmas 36 and 37. The first difference to note is that the effect of b_{ki}'s is simply a constant offset, and so the sequence would tend to $-\infty$ with or without its presence. Similar simplifications can be made inside the logarithm: the multiplicative effect of r_{ki} inside the logarithm can be extracted as an additive offset and thus similarly be discarded.

The second crucial difference is to relax the variable domains from integers to reals. If each of the λ_i in the satisfying assignment is sufficiently large, we show we can relax the condition to real numbers rather than integers without affecting whether the sequence goes to $-\infty$. To do this, we test sets of indices U, where if $i \in U$ then λ_i needs to be arbitrarily large over all C (i.e. unbounded). The positions where λ_i is always bounded are again a constant offset and are omitted. ◀

By testing the LC condition and the condition from Lemma 37 for each possible $X, \mathcal{Y}, k, U$, in turn using the relevant (conditionally decidable) first-order theory of the reals, we have:

▶ **Lemma 38.** *Given a weighted automaton $\mathcal{W}$ and states s, s' such that $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ are plus-letter-bounded, it is decidable whether s is big-O s', subject to Schanuel's conjecture.*

6.2 The letter-bounded case

Here we consider the case where $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ are letter-bounded, $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ are subsets of $a_1^* \dots a_m^*$ for some $a_1, \dots, a_m \in \Sigma$, which is a relaxation of the preceding case. For the plus-letter-bounded case, we relied on a 1-1 correspondence between numeric vectors and words. This correspondence no longer holds in the letter-bounded case: for example, a^n matches $a^*b^*a^*$, but it could correspond to $(n, 0, 0)$, $(0, 0, n)$, as well as any $(n_1, 0, n_2)$ with $n_1 + n_2 = n$. Still, there is a reduction to the plus-letter-bounded case.

▶ **Lemma 39.** *The big-O problem for $\mathcal{W}, s, s'$ with $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ letter-bounded reduces to the plus-letter-bounded case.*

Proof. Suppose the LC condition holds and $\mathcal{L}_s(\mathcal{W}) \subseteq \mathcal{L}_{s'}(\mathcal{W}) \subseteq a_1^* \cdots a_m^*$. Let I be the set of strictly increasing sequences $\vec{\imath} = i_1 \cdots i_k$ of integers between 1 and m. Given $\vec{\imath} \in I$, let $\mathcal{W}_{\vec{\imath}}$ be the weighted automaton obtained by intersecting $\mathcal{W}$ with a DFA for $a_{i_1}^+ \cdots a_{i_k}^+$ whose initial state is q. Note that s is big-O of s' (in $\mathcal{W}$) iff (s, q) is big-O of (s', q) in $\mathcal{W}_{\vec{\imath}}$ for all $\vec{\imath} \in I$, because $a_1^* \cdots a_m^* = \bigcup_{\vec{\imath} \in I} a_{i_1}^+ \cdots a_{i_k}^+$. Because the big-O problem for each $\mathcal{W}_{\vec{\imath}}$, (s, q), (s', q) falls into the plus-letter-bounded case, the results follows from Lemma 38. ◀

6.3 The bounded case

Here we consider the case where $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ are bounded, which is a relaxation of letter-boundedness (see Definition 6): $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ are subsets of $w_1^* \dots w_m^*$ for some $w_1, \dots, w_m \in \Sigma^*$. We show a reduction to the letter-bounded case from Section 6.2.

To showcase the difference to the letter-bounded case, consider the language $(abab)^*a^*b^*(ab)^*$. Observe that, for example the word $(ab)^4$ can be decomposed in a number of ways: $(abab)^2a^0b^0(ab)^0$, $(abab)^1a^1b^1(ab)^1$, $(abab)^1a^0b^0(ab)^2$, $(abab)^0a^1b^1(ab)^3$ or $(abab)^0a^0b^0(ab)^4$. One must be careful to consider all such decompositions.

▶ **Lemma 40.** *The big-O problem for $\mathcal{W}, s, s'$ with $\mathcal{L}_s(\mathcal{W})$ and $\mathcal{L}_{s'}(\mathcal{W})$ bounded reduces to the letter-bounded case.*

Proof sketch. Suppose $\mathcal{W}$ is bounded over $w_1^* \dots w_m^*$, we will construct a new weighted automaton $\mathcal{W}'$ letter-bounded over a new alphabet $a_1^* \dots a_m^*$ with the following property. For *every* decomposition of a word w, as $w_1^{n_1} \dots w_m^{n_m}$, the weight of $a_1^{n_1} \dots a_m^{n_m}$ in $\mathcal{W}'$ is equal to the weight of w in $\mathcal{W}$. ◀

7 Conclusion

Despite undecidability results, we have identified several decidable cases of the big-O problem. However, for bounded languages, the result depends on a conjecture from number theory, leaving open the exact borderline between decidability and undecidability.

Natural directions for future work include the analogous problem for infinite words, further analysis on ambiguity (e.g., is the big-O problem decidable for k-ambiguous weighted automata?), and the extension to negative edge weights.

References

1 Shaull Almagor, Udi Boker, and Orna Kupferman. What's decidable about weighted automata? In *ATVA*, volume 6996 of *Lecture Notes in Computer Science*, pages 482–491. Springer, 2011.

2 Shaull Almagor, Dmitry Chistikov, Joël Ouaknine, and James Worrell. O-minimal invariants for linear loops. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, volume 107 of *LIPIcs*, pages 114:1–114:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.ICALP.2018.114.

3 Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and computation in mathematics*. Springer, 2nd edition, 2006.

4 Konstantinos Chatzikokolakis, Daniel Gebler, Catuscia Palamidessi, and Lili Xu. Generalized Bisimulation Metrics. In Paolo Baldan and Daniele Gorla, editors, *CONCUR 2014 - Concurrency Theory - 25th International Conference, CONCUR 2014*, volume 8704 of *Lecture Notes in Computer Science*, pages 32–46. Springer, 2014. doi:10.1007/978-3-662-44584-6_4.

5 Taolue Chen and Stefan Kiefer. On the total variation distance of labelled Markov chains. In Thomas A. Henzinger and Dale Miller, editors, *Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), CSL-LICS 2014*, pages 33:1–33:10. ACM, 2014. doi:10.1145/2603088.2603099.

6 Dmitry Chistikov, Andrzej S. Murawski, and David Purser. Asymmetric distances for approximate differential privacy. In Wan Fokkink and Rob van Glabbeek, editors, *30th International Conference on Concurrency Theory, CONCUR 2019*, volume 140 of *LIPIcs*, pages 10:1–10:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.CONCUR.2019.10.

7 Ventsislav Chonev, Joël Ouaknine, and James Worrell. On the Skolem problem for continuous linear dynamical systems. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 100:1–100:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.ICALP.2016.100.

8 Thomas Colcombet. Unambiguity in automata theory. In Jeffrey O. Shallit and Alexander Okhotin, editors, *Descriptional Complexity of Formal Systems - 17th International Workshop, DCFS 2015*, volume 9118 of *Lecture Notes in Computer Science*, pages 3–18. Springer, 2015. doi:10.1007/978-3-319-19225-3_1.

9 Laure Daviaud, Marcin Jurdzinski, Ranko Lazic, Filip Mazowiecki, Guillermo A. Pérez, and James Worrell. When is containment decidable for probabilistic automata? In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, volume 107 of *LIPIcs*, pages 121:1–121:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.ICALP.2018.121.

10 C. Dehnert, S. Junges, J.-P. Katoen, and M. Volk. A Storm is coming: A modern probabilistic model checker. In *Proceedings of Computer Aided Verification (CAV)*, pages 592–600. Springer, 2017.

11 Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer, 2006. doi:10.1007/11681878_14.

12 Nathanaël Fijalkow. Undecidability results for probabilistic automata. *SIGLOG News*, 4(4):10–17, 2017. URL: https://dl.acm.org/citation.cfm?id=3157833.

13 Shmuel Friedland and Hans Schneider. The growth of powers of a nonnegative matrix. *SIAM J. Matrix Analysis Applications*, 1(2):185–200, 1980. doi:10.1137/0601022.

14 Pawel Gawrychowski, Dalia Krieger, Narad Rampersad, and Jeffrey Shallit. Finding the growth rate of a regular or context-free language in polynomial time. *Int. J. Found. Comput. Sci.*, 21(4):597–618, 2010. doi:10.1142/S0129054110007441.

15 Hugo Gimbert and Youssouf Oualhadj. Probabilistic automata on finite words: Decidable and undecidable problems. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide, and Paul G. Spirakis, editors, *Automata, Languages and Programming, 37th International Colloquium, ICALP 2010, Bordeaux, France, July 6-10, 2010, Proceedings, Part II*, volume 6199 of *Lecture Notes in Computer Science*, pages 527–538. Springer, 2010. doi:10.1007/978-3-642-14162-1_44.

16 Seymour Ginsburg. *The Mathematical Theory of Context-Free Languages*. McGraw-Hill, 1966.

17 Seymour Ginsburg and Edwin H Spanier. Bounded algol-like languages. *Transactions of the American Mathematical Society*, 113(2):333–368, 1964.

18 Cheng-Chao Huang, Jing-Cao Li, Ming Xu, and Zhi-Bin Li. Positive root isolation for poly-powers by exclusion and differentiation. *Journal of Symbolic Computation*, 85:148–169, 2018. 41th International Symposium on Symbolic and Alge-braic Computation (ISSAC'16). doi:10.1016/j.jsc.2017.07.007.

19 Stefan Kiefer. On computing the total variation distance of hidden Markov models. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018*, volume 107 of *LIPIcs*, pages 130:1–130:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.ICALP.2018.130.

20 Stefan Kiefer, Andrzej S. Murawski, Joël Ouaknine, Björn Wachter, and James Worrell. On the Complexity of Equivalence and Minimisation for Q-weighted Automata. *Logical Methods in Computer Science*, 9(1), 2013. doi:10.2168/LMCS-9(1:8)2013.

21 Daniel Krob. The equality problem for rational series with multiplicities in the tropical semiring is undecidable. *International Journal of Algebra and Computation*, 4:405–425, 1994.

22 M. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of probabilistic real-time systems. In *Proceedings of Computer Aided Verification (CAV)*, volume 6806 of *LNCS*, pages 585–591. Springer, 2011.

23 Serge Lang. *Introduction to transcendental numbers*. Addison-Wesley Pub. Co., 1966.

24 Angus Macintyre and Alex J Wilkie. On the decidability of the real exponential field, 1996.

25 Rupak Majumdar, Mahmoud Salamati, and Sadegh Soudjani. On decidability of time-bounded reachability in CTMDPs. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020)*, volume 168 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 133:1–133:19, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ICALP.2020.133.

26 David Mestel. Quantifying information flow in interactive systems. In *32nd IEEE Computer Security Foundations Symposium, CSF 2019, Hoboken, NJ, USA, June 25-28, 2019*, pages 414–427. IEEE, 2019. doi:10.1109/CSF.2019.00035.

27 David Mestel. Widths of Regular and Context-Free Languages. In *39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2019)*, volume 150 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 49:1–49:14, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.FSTTCS.2019.49.

28 Albert R. Meyer and Larry J. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *Proceedings of the 13th Annual Symposium on Switching and Automata Theory, College Park, Maryland, USA, October 25-27, 1972*, pages 125–129. IEEE Computer Society, 1972.

29 Joël Ouaknine and James Worrell. Positivity problems for low-order linear recurrence sequences. In Chandra Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014*, pages 366–379. SIAM, 2014. doi:10.1137/1.9781611973402.27.

30 Rohit J Parikh. On context-free languages. *Journal of the ACM (JACM)*, 13(4):570–581, 1966.

31 Azaria Paz. *Introduction to probabilistic automata*. Academic Press, 2014.

32 Bjorn Poonen. Hilbert's tenth problem over rings of number-theoretic interest. *Note from the lecture at the Arizona Winter School on "Number Theory and Logic"*, 2003. URL: https://math.mit.edu/~poonen/papers/aws2003.pdf.

33 Hans Schneider. The influence of the marked reduced graph of a nonnegative matrix on the Jordan form and on related properties: A survey. *Linear Algebra and its Applications*, 84:161–189, 1986.

34 Marcel Paul Schützenberger. On the definition of a family of automata. *Information and Control*, 4(2-3):245–270, 1961. doi:10.1016/S0019-9958(61)80020-X.

35 Bruno Sericola. *Markov chains: theory and applications*. John Wiley & Sons, 2013.

36 Adam D. Smith. Efficient, Differentially Private Point Estimators. *CoRR*, abs/0809.4794, 2008. arXiv:0809.4794.

37 Larry J. Stockmeyer and Albert R. Meyer. Word problems requiring exponential time: Preliminary report. In Alfred V. Aho, Allan Borodin, Robert L. Constable, Robert W. Floyd, Michael A. Harrison, Richard M. Karp, and H. Raymond Strong, editors, *Proceedings of the 5th Annual ACM Symposium on Theory of Computing, 1973*, pages 1–9. ACM, 1973. doi:10.1145/800125.804029.

38 Wen-Guey Tzeng. A polynomial-time algorithm for the equivalence of probabilistic automata. *SIAM J. Comput.*, 21(2):216–227, 1992. doi:10.1137/0221017.

39 Wen-Guey Tzeng. On path equivalence of nondeterministic finite automata. *Inf. Process. Lett.*, 58(1):43–46, 1996. doi:10.1016/0020-0190(96)00039-7.

40 Michel Waldschmidt. *Diophantine Approximation on Linear Algebraic Groups*, volume 326 of *Grundlehren der mathematischen Wissenschaften (A Series of Comprehensive Studies in Mathematics)*. Springer, Berlin, Heidelberg, 2000.