

Stochastic Context-Free Grammars, Regular Languages, and Newton's Method

Kousha Etessami¹, Alistair Stewart¹, and Mihalis Yannakakis²

¹ School of Informatics, University of Edinburgh
kousha@inf.ed.ac.uk , stewart.al@gmail.com

² Department of Computer Science, Columbia University
mihalis@cs.columbia.edu

Abstract. We study the problem of computing the probability that a given stochastic context-free grammar (SCFG), G , generates a string in a given regular language $L(D)$ (given by a DFA, D). This basic problem has a number of applications in statistical natural language processing, and it is also a key necessary step towards quantitative ω -regular model checking of stochastic context-free processes (equivalently, 1-exit recursive Markov chains, or stateless probabilistic pushdown processes).

We show that the probability that G generates a string in $L(D)$ can be computed to within arbitrary desired precision in polynomial time (in the standard Turing model of computation), under a rather mild assumption about the SCFG, G , and with no extra assumption about D . We show that this assumption is satisfied for SCFG's whose rule probabilities are learned via the well-known inside-outside (EM) algorithm for maximum-likelihood estimation (a standard method for constructing SCFGs in statistical NLP and biological sequence analysis). Thus, for these SCFGs the algorithm always runs in P-time.

1 Introduction

Stochastic (or Probabilistic) Context-Free Grammars (SCFG) are context-free grammars where the rules (productions) have associated probabilities. They are a central stochastic model, widely used in natural language processing [14], with applications also in biology (e.g. [2, 12]). A SCFG G generates a language $L(G)$ (like an ordinary CFG) and assigns a probability to every string in the language. SCFGs have been extensively studied since the 1970's. A number of important problems on SCFGs can be viewed as instances of the following *regular pattern matching problem* for different regular languages:

Given a SCFG G and a regular language L , given e.g., by a deterministic finite automaton (DFA) D , compute the probability $\mathbb{P}_G(L)$ that G generates a string in L , i.e. compute the sum of the probabilities of all the strings in L .

A simple example is when $L = \Sigma^*$, the set of all strings over the terminal alphabet Σ of the SCFG G . Then this problem simply asks to compute the probability $\mathbb{P}_G(L(G))$ of the language $L(G)$ generated by the grammar G . Alternatively, if we view the SCFG as a stochastic process that starts from the

start nonterminal, repeatedly applies the probabilistic rules to replace (say, left-most) nonterminals, and terminates when a string of terminals is reached, then $\mathbb{P}_G(L(G))$ is simply the probability that this process terminates. Another simple example is when L is a singleton, $L = \{w\}$, for some string w ; in this case the problem corresponds to the basic parsing question of computing the probability that a given string w is generated by the SCFG G . Another basic well-studied problem is the computation of *prefix probabilities*: given a SCFG G and a string w , compute the probability that G generates a string with prefix w [11, 21]. This is useful in online processing in speech recognition [11] and corresponds to the case $L = w\Sigma^*$. A more complex problem is the computation of *infix probabilities* [1, 18], where we wish to compute the probability that G generates a string that contains a given string w as a substring, which corresponds to the language $L = \Sigma^*w\Sigma^*$. In general, even when rule probabilities of the SCFG G are rational, the probabilities we wish to compute can be irrational. Thus the typical aim for “computing” them is to approximate them to desired precision.

Stochastic context-free grammars are closely related to *1-exit recursive Markov chains* (1-RMC) [8], and to *stateless probabilistic pushdown automata* (also called pBPA) [5]; these are two equivalent models for a subclass of probabilistic programs with recursive procedures. The above regular pattern matching problem for SCFGs is equivalent to the problem of computing the probability that a computation of a given 1-RMC (or pBPA) terminates and satisfies a given regular property. In other words, it corresponds to the quantitative model checking problem for 1-RMCs with respect to regular *finite string* properties.

We first review some prior related work, and then describe our results.

Previous Work. As mentioned above, there has been, on the one hand, substantial work in the NLP literature on different cases of the problem for various regular languages L , and on the other hand, there has been work in the verification and algorithms literature on the analysis and model checking of recursive Markov chains and probabilistic pushdown automata. Nevertheless, even the simple special case of $L = \Sigma^*$, the question of whether it is possible to compute (approximately) in polynomial time the desired probability for a given SCFG G (i.e. the probability $\mathbb{P}_G(L(G))$ of $L(G)$) was open until very recently. In [7] we showed that $\mathbb{P}_G(L(G))$ can be computed to arbitrary precision in polynomial time in the size of the input SCFG G and the number of bits of precision. From a SCFG G , one can construct a multivariate system of equations $x = P_G(x)$, where x is a vector of variables and P_G is a vector of polynomials with positive coefficients which sum to (at most) 1. Such a system is called a *probabilistic polynomial system* (PPS), and it always has a non-negative solution that is smallest in every coordinate, called the *least fixed point* (LFP). A particular coordinate of the LFP of the system $x = P_G(x)$ is the desired probability $\mathbb{P}_G(L(G))$. To compute $\mathbb{P}_G(L(G))$, we used a variant of Newton’s method on $x = P_G(x)$, with suitable rounding after each step to control the bit-size of numbers, and showed that it converges in P-time to the LFP [7]. Building on this, we also showed that the probability $\mathbb{P}_G(\{w\})$ of string w under SCFG G can also be computed to any precision in P-time in the size of G , w and the number of bits of precision.

The use of Newton’s method was proposed originally in [8] for computing termination probabilities for (multi-exit) RMC’s, which requires the solution of equations from a more general class of polynomial systems $x = P(x)$, called *monotone polynomial systems* (MPS), where the polynomials of P have positive coefficients, but their sum is not restricted to ≤ 1 . An arbitrary MPS may not have any non-negative solution, but if it does then it has a LFP, and a version of Newton provably converges to the LFP [8]. There are now implementations of variants of Newton’s method in several tools [22, 16] and experiments show that they perform well on many instances. The rate of convergence of Newton for general MPSs was studied in detail in [4], and was further studied most recently in [20] (see below). In certain cases, Newton converges fast, but in general there are exponential bad examples. Furthermore, there are negative results indicating it is very unlikely that any non-trivial approximation of termination probabilities of multi-exit RMCs, and the LFP of MPSs, can be done in P-time (see [8]).

The model checking problem for RMCs (equivalently pPDAs) and ω -regular properties was studied in [5, 9]. This is of course a more general problem than the problem for SCFGs (which correspond to 1-RMCs) and regular languages (the finite string case of ω -regular languages). It was shown in [9] that in the case of 1-RMCs, the qualitative problem of determining whether the probability that a run satisfies the property is 0 or 1 can be solved in P-time in the size of the 1-RMC, but for the quantitative problem of approximating the probability, the algorithm runs in PSPACE, and no better complexity bound was known.

The particular cases of computing prefix and infix probabilities for a SCFG have been studied in the NLP literature, but no polynomial time algorithm for general SCFGs is known. Jelinek and Lafferty gave an algorithm for grammars in Chomsky Normal Form (CNF) [11]. Note that a general SCFG G may not have any equivalent CNF grammar with rational rule probabilities, thus one can only hope for an “approximately equivalent” CNF grammar; constructing such a grammar in the case of stochastic grammars G is non-trivial, at least as difficult as computing the probability of $L(G)$, and the first P-time algorithm was given in [7]. Another algorithm for prefix probabilities by Stolcke [21] applies to general SCFGs, but in the presence of unary and ϵ -rules, the algorithm does not run in polynomial time. The problem of computing infix probabilities was studied in [1, 16, 18], and in particular [16, 18] cast it in the general regular language framework, and studied the general problem of computing the probability $\mathbb{P}_G(L(D))$ of the language $L(D)$ of a DFA D under a SCFG G . From G and D they construct a product *weighted context-free grammar* (WCFG) G' : a CFG with (positive) weights on the rules, which may not be probabilities, in particular the weights on the rules of a nonterminal may sum to more than 1. The desired probability $\mathbb{P}_G(L(D))$ is the weight of $L(G')$. As in the case of SCFGs, this weight is given by the LFP of a monotone system of equations $y = P_{G'}(y)$, however, unlike the case of SCFGs the system now is not a probabilistic system (thus our result of [7] does not apply). Nederhof and Satta then solve the system using the decomposed Newton method from [8] and Broyden’s (quasi-Newton) method, and present experimental results for infix probability computations.

Most recently, in [20], we have obtained worst-case upper bounds on (rounded and exact) Newton’s method applied to arbitrary MPSs, $x = P(x)$, as a function of the input encoding size $|P|$ and $\log(1/\epsilon)$, to converge to within additive error $\epsilon > 0$ of the LFP solution q^* . However, our bounds in [20], even when $0 < q^* \leq 1$, are exponential in the depth of (not necessarily critical) strongly connected components of $x = P(x)$, and furthermore they also depend linearly on $\log(\frac{1}{q_{\min}^*})$, where $q_{\min}^* = \min_i q_i^*$, which can be $\approx \frac{1}{2^{2^{|P|}}}$. As we describe next, we do far better in this paper for the MPSs that arise from the “product” of a SCFG and a DFA.

Our Results. We study the general problem of computing the probability $\mathbb{P}_G(L(D))$ that a given SCFG G generates a string in the language $L(D)$ of a given DFA D . We show that, under a certain mild assumption on G , this probability can be computed to any desired precision in time polynomial in the encoding sizes of G & D and the number of bits of precision.

We now sketch briefly the approach and state the assumption on G . First we construct from G and D the product weighted CFG $G' = G \otimes D$ as in [16] and construct the corresponding MPS $y = P_{G'}(y)$, whose LFP contains the desired probability $\mathbb{P}_G(L(D))$ as one of its components. The system is monotone but not probabilistic. We eliminate (in P-time) those variables that have value 0 in the LFP, and apply Newton, with suitable rounding in every step. The heart of the analysis shows there is a tight algebraic correspondence between the behavior of Newton’s method on this MPS and its behavior on the probabilistic polynomial system (PPS) $x = P_G(x)$ of G . In particular, this correspondence shows that, with exact arithmetic, the two computations converge at the same rate. By exploiting this, and by extending recent results we established for PPSs, we obtain the conditional polynomial upper bound. Specifically, call a PPS $x = P(x)$ *critical* if the spectral radius of the Jacobian of $P(x)$, evaluated at the LFP q^* is equal to 1 (it is always ≤ 1). We can form a dependency graph between the variables of a PPS, and decompose the variables and the system into strongly connected components (SCCs); an SCC is called *critical* if the induced subsystem on that SCC is critical. The *critical depth* of a PPS is the maximum number of critical SCCs on any path of the DAG of SCCs (i.e. the max nesting depth of critical SCCs). We show that if the PPS of the given SCFG G has bounded (or even logarithmic) critical depth, then we can compute $\mathbb{P}_G(L(D))$ (for any DFA D) in polynomial time in the size of G , D and the number of bits of precision.

Furthermore, we show this condition is satisfied by a broad class of SCFGs used in applications. Specifically, a standard way the probabilities of rules of a SCFG are set is by using the EM (inside-outside) algorithm. We show that the SCFGs constructed in this way are guaranteed to be noncritical (i.e., have critical depth 0). So for these SCFGs, and any DFA, the algorithm runs in P-time.

The paper is organized as follows. Section 2 gives definitions and background. Section 3 establishes tight algebraic connections between the behavior of Newton on the PPS of the SCFG, and on the MPS of the product WCFG. Section 4 proves the claimed bounds on rounded Newton’s method. Section 5 shows the noncriticality of SCFGs obtained by the EM method. Proofs are in the Appendix.

2 Definitions and Background

A *weighted context-free grammar* (WCFG), $G = (V, \Sigma, R, p)$, has a finite set V of *nonterminals*, a finite set Σ of *terminals* (alphabet symbols), and a finite list of *rules*, $R \subset V \times (V \cup \Sigma)^*$, where each rule $r \in R$ is a pair (A, γ) , which we usually denote by $A \rightarrow \gamma$, where $A \in V$ and $\gamma \in (V \cup \Sigma)^*$. Finally $p : R \rightarrow \mathbb{R}^+$ maps each rule $r \in R$ to a positive *weight*, $p(r) > 0$. We often denote a rule $r = (A \rightarrow \gamma)$ together with its weight by writing $A \xrightarrow{p(r)} \gamma$. We will sometimes also specify a specific non-terminal $S \in V$ as the starting symbol.

Note that we allow $\gamma \in (V \cup \Sigma)^*$ to possibly be the empty string, denoted by ϵ . A rule of the form $A \rightarrow \epsilon$ is called an ϵ -rule. For a rule $r = (A \rightarrow \gamma)$, we let $\text{left}(r) := A$ and $\text{right}(r) := \gamma$. We let $R_A = \{r \in R \mid \text{left}(r) = A\}$. For $A \in V$, let $p(A) = \sum_{r \in R_A} p(r)$. A WCFG, G , is called a *stochastic* or *probabilistic context-free grammar* (SCFG or PCFG; we shall use SCFG), if for $\forall A \in V$, $p(A) \leq 1$. An SCFG is called *proper* if $\forall A \in V$, $p(A) = 1$.

We will say that an WCFG, $G = (V, \Sigma, R, p)$ is in *Simple Normal Form* (SNF) if every nonterminal $A \in V$ belongs to one of the following three types:

1. type L: every rule $r \in R_A$, has the form $A \xrightarrow{p(r)} B$.
2. type Q: there is a single rule in R_A : $A \xrightarrow{1} BC$, for some $B, C \in V$.
3. type T: there is a single rule in R_A : either $A \xrightarrow{1} \epsilon$, or $A \xrightarrow{1} a$ for some $a \in \Sigma$.

For a WCFG, G , strings $\alpha, \beta \in (V \cup \Sigma)^*$, and $\pi = r_1 \dots r_k \in R^*$, we write $\alpha \xRightarrow{\pi} \beta$ if the leftmost derivation starting from α , and applying the sequence π of rules, derives β . We let $p(\alpha \xRightarrow{\pi} \beta) = \prod_{i=1}^k p(r_k)$ if $\alpha \xRightarrow{\pi} \beta$, and $p(\alpha \xRightarrow{\pi} \beta) = 0$ otherwise. If $A \xRightarrow{\pi} w$ for $A \in V$ and $w \in \Sigma^*$, we say that π is a *complete* derivation from A and its *yield* is $y(\pi) = w$. There is a natural one-to-one correspondence between the complete derivations of w starting at A and the *parse trees* of w rooted at A , and this correspondence preserves weights.

For a WCFG, $G = (V, \Sigma, R, p)$, nonterminal $A \in V$, and terminal string $w \in \Sigma^*$, we let $p_A^{G,w} = \sum_{\{\pi \mid y(\pi)=w\}} p(A \xRightarrow{\pi} w)$. For a general WCFG, $p_A^{G,w}$ need not be a finite value (it may be $+\infty$, since the sum may not converge). Note however that if G is an SCFG, then $p_A^{G,w}$ defines the probability that, starting at nonterminal A , G generates w , and thus it is clearly finite.

The *termination probability* (*termination weight*) of an SCFG (WCFG), G , starting at nonterminal A , denoted q_A^G , is defined by $q_A^G = \sum_{w \in \Sigma^*} p_A^{G,w}$. Again, for an arbitrary WCFG q_A^G need not be a finite number. A WCFG G is called *convergent* if q_A^G is finite for all $A \in V$. We will only encounter convergent WCFGs in this paper, so when we say WCFG we mean convergent WCFG, unless otherwise specified. In G is an SCFG, then q_A^G is just the total probability with which the derivation process starting at A eventually generates a finite string and (thus) stops, so SCFGs are clearly convergent.

An SCFG, G , is called *consistent starting at A* if $q_A^G = 1$, and G is called *consistent* if it is consistent starting at every nonterminal. Note that even if a SCFG, G , is proper this does not necessarily imply that G is consistent. For an

SCFG, G , we can decide whether $q_A^G = 1$ in P-time ([8]). The same decision problem is PosSLP-hard for convergent WCFGs ([8]).

For any WCFG, $G = (V, \Sigma, R, p)$, with $n = |V|$, assume the nonterminals in V are indexed as A_1, \dots, A_n . We define the following **monotone polynomial system of equations** (MPS) associated with G , denoted $x = P_G(x)$. Here $x = (x_1, \dots, x_n)$ denotes an n -vector of variables. Likewise $P_G(x) = (P_G(x)_1, \dots, P_G(x)_n)$ denotes an n -vector of multivariate polynomials over the variables $x = (x_1, \dots, x_n)$. For a vector $\kappa = (\kappa_1, \kappa_2, \dots, \kappa_n) \in \mathbb{N}^n$, we use the notation x^κ to denote the monomial $x_1^{\kappa_1} x_2^{\kappa_2} \dots x_n^{\kappa_n}$. For a non-terminal $A_i \in V$, and a string $\alpha \in (V \cup \Sigma)^*$, let $\kappa_i(\alpha) \in \mathbb{N}$ denote the number of occurrences of A_i in the string α . We define $\kappa(\alpha) \in \mathbb{N}^n$ to be $\kappa(\alpha) = (\kappa_1(\alpha), \kappa_2(\alpha), \dots, \kappa_n(\alpha))$.

In the MPS $x = P_G(x)$, corresponding to each nonterminal $A_i \in V$, there will be one variable x_i and one equation, namely $x_i = P_G(x)_i$, where: $P_G(x)_i \equiv \sum_{r=(A \rightarrow \alpha) \in R_{A_i}} p(r) x^{\kappa(\alpha)}$. If there are no rules associated with A_i , i.e., if $R_{A_i} = \emptyset$, then by default we define $P_G(x)_i \equiv 0$. Note that if $r \in R_{A_i}$ is a terminal rule, i.e., $\kappa(r) = (0, \dots, 0)$, then $p(r)$ is one of the constant terms of $P_G(x)_i$.

Note: Throughout this paper, for any n -vector z , whose i 'th coordinate z_i "corresponds" to nonterminal A_i , we often find it convenient to use z_{A_i} to refer to z_i . So, e.g., we alternatively use x_{A_i} and $P_G(x)_{A_i}$, instead of x_i and $P_G(x)_i$.

Note that if G is a SCFG, then in $x = P_G(x)$, by definition, the sum of the monomial coefficients and constant terms of each polynomial $P_G(x)_i$ is at most 1, because $\sum_{r \in R_{A_i}} p(r) \leq 1$ for every $A_i \in V$. An MPS that satisfies this extra condition is called a **probabilistic polynomial system of equations** (PPS).

Consider any MPS, $x = P(x)$, with n variables, $x = (x_1, \dots, x_n)$. Let $\mathbb{R}_{\geq 0}^n$ denote the non-negative real numbers. Then $P(x)$ defines a monotone operator on the non-negative orthant $\mathbb{R}_{\geq 0}^n$. In general, an MPS need not have any real-valued solution: consider $x = x + 1$. However, by monotonicity of $P(x)$, if there exists $a \in \mathbb{R}_{\geq 0}^n$ such that $a = P(a)$, then there is a *least fixed point* (LFP) solution $q^* \in \mathbb{R}_{\geq 0}^n$ such that $q^* = P(q^*)$, and such that $q^* \leq a$ for all solutions $a \in \mathbb{R}_{\geq 0}^n$.

Proposition 1. (cf. [8] or see [17]) *For any SCFG (or convergent WCFG), G , with n nonterminals A_1, \dots, A_n , the LFP solution of $x = P_G(x)$ is the n -vector $q^G = (q_{A_1}^G, \dots, q_{A_n}^G)$ of termination probabilities (termination weights) of G .*

For computation purposes, we assume that the input probabilities (weights) associated with rules of input SCFGs or WCFGs are positive rationals encoded by giving their numerator and denominator in binary. We use $|G|$ to denote the encoding size (i.e., number of bits) of an input WCFG G .

Given any WCFG (SCFG) $G = (V, \Sigma, R, p)$ we can compute in linear time an SNF form WCFG (resp. SCFG) $G' = (V' \Sigma, R', p')$ of size $|G'| = O(|G|)$ with $V' \supseteq V$ such that $q_A^{G,w} = q_A^{G',w}$ for all $A \in V$, $w \in \Sigma^*$ (cf. [8] and Proposition 2.1 of [7]). Thus, for the problems studied in this paper, we may assume wlog that a given input WCFG or SCFG is in SNF form.

A DFA, $D = (Q, \Sigma, \Delta, s_0, F)$, has states Q , alphabet Σ , transition function $\Delta : Q \times \Sigma \rightarrow Q$, start state $s_0 \in Q$ and final states $F \subseteq Q$. We extend Δ to strings: $\Delta^* : Q \times \Sigma^* \rightarrow Q$ is defined by induction on the length $|w| \geq 0$ of

$w \in \Sigma^*$: for $s \in Q$, $\Delta^*(s, \epsilon) := s$. Inductively, if $w = aw'$, with $a \in \Sigma$, then $\Delta^*(s, w) := \Delta^*(\Delta(s, a), w')$. We define $L(D) = \{w \in \Sigma^* \mid \Delta^*(s_0, w) \in F\}$.

Given a WCFG G and a DFA D over the same terminal alphabet, for any nonterminal A of G , we define $q_A^{G,D} = \sum_{w \in L(D)} q_A^{G,w}$. If G is a SCFG, $q_A^{G,D}$ simply denotes the probability that G , starting at A , generates a string in $L(D)$. Our goal is to compute $q_A^{G,D}$, given SCFG G and DFA D . In general, $q_A^{G,D}$ may be an irrational probability, even when all of the rule probabilities of G are rational values. So one natural goal is to approximate $q_A^{G,D}$ to within desired precision. More precisely, the approximation problem is this: given as input an SCFG, G , with a specified nonterminal A , a DFA, D , over the same terminal alphabet Σ , and a rational error threshold $\delta > 0$, output a rational value $v \in [0, 1]$ such that $|v - q_A^{G,D}| < \delta$. We would like to do this as efficiently as possible as a function of the input size: $|G|$, $|D|$, and $\log(1/\delta)$.

To compute $q_A^{G,D}$, it will be useful to define a WCFG obtained as the *product* of a SCFG and a DFA. We assume, wlog, that the input SCFG is in SNF form. The **product** (or **intersection**) of a SCFG $G = (V, \Sigma, R, p)$ in SNF form, and DFA, $D = (Q, \Sigma, \Delta, s_0, F)$, is defined to be a new WCFG, $G \otimes D = (V', \Sigma, R', p')$, where the set of nonterminals is $V' = Q \times V \times Q$. Assuming $n = |V|$ and $d = |Q|$, then $|V'| = d^2n$. The rules R' and rule probabilities p' of the product $G \otimes D$ are defined as follows (recall G is assumed to be in SNF):

- Rules of form L: For every rule of the form $(A \xrightarrow{p} B) \in R$, and every pair of states $s, t \in Q$, there is a rule $(sAt) \xrightarrow{p} (sBt)$ in R' .
 - Rules of form Q: for every rule $(A \xrightarrow{1} BC) \in R$, and for all states $s, t, u \in Q$, there is a rule $(sAu) \xrightarrow{1} (sBt)(tCu)$ in R' .
 - Rules of form T: for every rule $(A \xrightarrow{1} a) \in R$, where $a \in \Sigma$, and for every state $s \in Q$, if $\Delta(s, a) = t$, then there is a rule $(sAt) \xrightarrow{1} a$ in R' .
- For every rule $(A \xrightarrow{1} \epsilon) \in R$, and every $s \in Q$, there is a rule $(sAs) \xrightarrow{1} \epsilon$

Associated with the WCFG, $G \otimes D$, is the MPS $y = P_{G \otimes D}(y)$, where y is now a d^2n -vector of variables, where $n = |V|$ and $d = |Q|$. The LFP solution of this MPS captures the probabilities $q_A^{G,D}$ in the following sense:

Proposition 2. (cf. [18], or [9] for a variant of this) *For any SCFG, $G = (V, \Sigma, R, p)$, and DFA, $D = (Q, \Sigma, \Delta, s_0, F)$, the LFP solution $q^{G \otimes D}$ of the MPS $x = P_{G \otimes D}(x)$, satisfies $\mathbf{0} \leq q^{G \otimes D} \leq \mathbf{1}$. Furthermore, for any $A \in V$ and $s, t \in Q$, $q_{(sAt)}^{G \otimes D} = \sum_{\{w \mid \Delta^*(s, w) = t\}} q_A^{G,w}$. Thus, for every $A \in V$, $q_A^{G,D} = \sum_{t \in F} q_{(s_0At)}^{G \otimes D}$.*

Newton's method (NM). For an MPS (or PPS), $x = P(x)$, in n variables, let $B(x) := P'(x)$ denote the Jacobian matrix of $P(x)$. In other words, $B(x)$ is an $n \times n$ matrix such that $B(x)_{i,j} = \frac{\partial P_i(x)}{\partial x_j}$. For a vector $z \in \mathbb{R}^n$, assuming that matrix $(I - B(z))$ is non-singular, we define a single iteration of Newton's method (NM) for $x = P(x)$ on z via the following operator:

$$\mathcal{N}(z) := z + (I - B(z))^{-1}(P(z) - z) \quad (1)$$

Using Newton iteration, starting at n -vector $x^{(0)} := \mathbf{0}$, yields the following iteration: $x^{(k+1)} := \mathcal{N}(x^{(k)})$, for $k = 0, 1, 2, \dots$

For every MPS, we can detect in P-time all the variables x_j such that $q_j^* = 0$ [8]. We can then remove these variables and their corresponding equation $x_j = P(x)_j$, and substitute their values on the right hand sides of remaining equations. This yields a new MPS, with LFP $q' > 0$, which corresponds to the non-zero coordinates of q^* . It was shown in [8] that one can always apply a decomposed Newton's method to this MPS, to converge monotonically to the LFP solution.

Proposition 3. (cf. Theorem 6.1 of [8] and Theorem 4.1 of [4]) *Let $x = P(x)$ be a MPS, with LFP $q^* > \mathbf{0}$. Then starting at $x^{(0)} := \mathbf{0}$, the Newton iterations $x^{(k+1)} := \mathcal{N}(x^{(k)})$ are well defined and monotonically converge to q^* , i.e. $\lim_{k \rightarrow \infty} x^{(k)} = q^*$, and $x^{(k+1)} \geq x^{(k)} \geq \mathbf{0}$ for all $k \geq 0$.*

Unfortunately, it was shown in [8] that obtaining any non-trivial additive approximation to the LFP solution of a general MPS, even one whose LFP is $0 < q^* \leq 1$, is **PosSLP**-hard, so we can not compute the termination weights of general WCFGs in P-time (nor even in NP), without a major breakthrough in the complexity of numerical computation. (See [8] for more information.)

Fortunately, for the class of PPSs, we can do a lot better. First we can identify in P-time also all the variables x_j such that $q_j^* = 1$ [8] and remove them from the system. We showed recently in [7] that by then applying a suitably *rounded down* variant of Newton's method to the resulting PPS, we can approximate q^* within additive error 2^{-j} in time polynomial in the size of the PPS and j .

3 Balance, Collapse, and Newton's method

For an SCFG, $G = (V, \Sigma, R, p)$, and a DFA, $D = (Q, \Sigma, \Delta, s_0, F)$, we want to relate the behavior of Newton's method on the MPS associated with the WCFG, $G \otimes D$, to that of the PPS associated with the SCFG G . We shall show that there is indeed a tight correspondence, regardless of what the DFA D is. This holds even when G itself is a convergent WCFG, and thus $x = P_G(x)$ is an MPS. We need an abstract algebraic way to express this correspondence. A key notion will be *balance*, and the *collapse* operator defined on balanced vectors and matrices.

Consider the LFP q^G of $x = P_G(x)$, and LFP $q^{G \otimes D}$ of $y = P_{G \otimes D}(y)$. By Prop. 1 and 2, for any $A \in V$, $q_A^G = \sum_{w \in \Sigma^*} q_A^{G,w}$ is the probability (weight) that G , starting at A , generates any finite string. Likewise $q_{(sAt)}^{G \otimes D} = \sum_{\{w \mid \Delta^*(s,w)=t\}} q_A^{G,w}$ is the probability (weight) that, starting at A , G generates a finite string w such that $\Delta^*(s, w) = t$. Thus, for any $A \in V$ and $s \in Q$, $q_A^G = \sum_{t \in Q} q_{(sAt)}^{G \otimes D}$.

It turns out that analogous relationships hold between many other vectors associated with G and $G \otimes D$, including between the Newton iterates obtained by applying Newton's method to their respective PPS (or MPS) and the product MPS. Furthermore, associated relationships also hold between the Jacobian matrices $B_G(x)$ and $B_{G \otimes D}(y)$ of $P_G(x)$ and $P_{G \otimes D}(y)$, respectively.

Let $n = |V|$ and let $d = |Q|$. A vector $y \in \mathbb{R}^{d^2 n}$, whose coordinates are indexed by triples $(sAt) \in Q \times V \times Q$, is called **balanced** if for any non-terminal

A , and any pair of states $s, s' \in Q$, $\sum_{t \in Q} y_{(sAt)} = \sum_{t \in Q} y_{(s'At)}$. In other words, y is balanced if the value of the sum $\sum_{t \in Q} y_{(sAt)}$ is independent of the state s . As already observed, $q^{G \otimes D} \in \mathbb{R}_{\geq 0}^{d^2 n}$ is balanced. Let $\mathfrak{B} \subseteq \mathbb{R}^{d^2 n}$ denote the set of balanced vectors. Let us define the **collapse** mapping $\mathfrak{C} : \mathfrak{B} \rightarrow \mathbb{R}^n$. For any $A \in V$, $\mathfrak{C}(y)_A := \sum_t y_{(sAt)}$. Note: $\mathfrak{C}(y)$ is well-defined, because for $y \in \mathfrak{B}$, and any $A \in V$, the sum $\sum_t y_{(sAt)}$ is by definition independent of the state s .

We next extend the definition of balance to matrices. A matrix $M \in \mathbb{R}^{d^2 n \times d^2 n}$ is called **balanced** if, for any non-terminals $B, C \in V$ and states $s, u \in Q$, and for any pair of states $v, v' \in Q$, $\sum_t M_{(sBt), (uCv)} = \sum_t M_{(sBt), (uCv')}$, and for any $s, v \in Q$ and $s', v' \in Q$, $\sum_{t,u} M_{(sBt), (uCv)} = \sum_{t,u} M_{(s'Bt), (uCv')}$. Let $\mathfrak{B}^\times \subseteq \mathbb{R}^{d^2 n \times d^2 n}$ denote the set of balanced matrices. We extend the **collapse** map \mathfrak{C} to matrices. $\mathfrak{C} : \mathfrak{B}^\times \rightarrow \mathbb{R}^{n \times n}$ is defined as follows. For any $M \in \mathfrak{B}^\times$, and any $B, C \in V$, $\mathfrak{C}(M)_{BC} := \sum_{t,u} M_{(sBt), (uCv)}$. Note, again, $\mathfrak{C}(M)$ is well-defined.

We denote the Newton operator, \mathcal{N} , applied to a vector $x' \in \mathbb{R}^n$ for the PPS $x = P_G(x)$ associated with G by $\mathcal{N}_G(x')$. Likewise, we denote the Newton operator applied to a vector $y' \in \mathbb{R}^{d^2 n}$ for the MPS $y = P_{G \otimes D}(y)$ associated with $G \otimes D$ by $\mathcal{N}_{G \otimes D}(y')$. For a real square matrix M , let $\rho(M)$ denote the spectral radius of M . The main result of this section is the following:

Theorem 1. *Let $x = P_G(x)$ be any PPS (or MPS), with n variables, associated with a SCFG (or WCFG) G , and let $y = P_{G \otimes D}(y)$ be the corresponding product MPS, for any DFA D , with d states. For any balanced vector $y \in \mathfrak{B} \subseteq \mathbb{R}^{d^2 n}$, with $y \geq 0$, $\rho(B_{G \otimes D}(y)) = \rho(B_G(\mathfrak{C}(y)))$. Furthermore, if $\rho(B_{G \otimes D}(y)) < 1$, then $\mathcal{N}_{G \otimes D}(y)$ is defined and balanced, $\mathcal{N}_G(\mathfrak{C}(y))$ is defined, and $\mathfrak{C}(\mathcal{N}_{G \otimes D}(y)) = \mathcal{N}_G(\mathfrak{C}(y))$. Thus, $\mathcal{N}_{G \otimes D}$ preserves balance, and the collapse map \mathfrak{C} “commutes” with \mathcal{N} over non-negative balanced vectors, irrespective of what the DFA D is.*

We prove this in the appendix via a series of lemmas that reveal many algebraic/analytic properties of balance, collapse, and Newton’s method. Key is:

Lemma 1. *Let $\mathfrak{B}_{\geq 0} = \mathfrak{B} \cap \mathbb{R}_{\geq 0}^{d^2 n}$ and $\mathfrak{B}_{\geq 0}^\times = \mathfrak{B} \cap \mathbb{R}_{\geq 0}^{d^2 n \times d^2 n}$.*

We have $q^{G \otimes D} \in \mathfrak{B}_{\geq 0}$ and $\mathfrak{C}(q^{G \otimes D}) = q^G$, and:

- (i) *If $y \in \mathfrak{B}_{\geq 0} \subseteq \mathbb{R}_{\geq 0}^{d^2 n}$ then $B_{G \otimes D}(y) \in \mathfrak{B}_{\geq 0}^\times$, and $\mathfrak{C}(B_{G \otimes D}(y)) = B_G(\mathfrak{C}(y))$.*
- (ii) *If $y \in \mathfrak{B}_{\geq 0}$, then $P_{G \otimes D}(y) \in \mathfrak{B}_{\geq 0}$, and $\mathfrak{C}(P_{G \otimes D}(y)) = P_G(\mathfrak{C}(y))$.*
- (iii) *If $y \in \mathfrak{B}_{\geq 0}$ and $\rho(B_G(\mathfrak{C}(y))) < 1$, then $I - B_{G \otimes D}(y)$ is non-singular, $(I - B_{G \otimes D}(y))^{-1} \in \mathfrak{B}_{\geq 0}^\times$, and $\mathfrak{C}((I - B_{G \otimes D}(y))^{-1}) = (I - B_G(\mathfrak{C}(y)))^{-1}$.*
- (iv) *If $y \in \mathfrak{B}_{\geq 0}$ and $\rho(B_G(\mathfrak{C}(y))) < 1$, then $\mathcal{N}_{G \otimes D}(y) \in \mathfrak{B}^\times$ and $\mathfrak{C}(\mathcal{N}_{G \otimes D}(y)) = \mathcal{N}_G(\mathfrak{C}(y))$.*

An easy consequence of Thm. 1 (and Prop. 3) is that if we use NM with exact arithmetic on the PPS or MPS, $x = P_G(x)$, and on the product MPS, $y = P_{G \otimes D}(y)$, they converge at the same rate:

Corollary 1. *For any PPS or MPS, $x = P_G(x)$, with LFP $q^G > 0$, and corresponding product MPS, $y = P_{G \otimes D}(y)$, if we use Newton’s method with exact arithmetic, starting at $x^{(0)} := 0$, and $y^{(0)} := 0$, then all the Newton iterates $x^{(k)}$ and $y^{(k)}$ are well-defined, and for all k : $x^{(k)} = \mathfrak{C}(y^{(k)})$.*

4 Rounded Newton on PPSs and product MPSs

To work in the Turing model of computation (as opposed to the unit-cost RAM model) we have to consider *rounding* between iterations of NM, as in [7].

Definition 1. (Rounded-down Newton’s method (R-NM), with parameter h .) *Given an MPS, $x = P(x)$, with LFP q^* , where $q^* > 0$, in R-NM with integer rounding parameter $h > 0$, we compute a sequence of iteration vectors $x^{[k]}$. Starting with $x^{[0]} := \mathbf{0}$, $\forall k \geq 0$ we compute $x^{[k+1]}$ as follows:*

1. *Compute $x^{\{k+1\}} := \mathcal{N}_P(x^{[k]})$, where $\mathcal{N}_P(x)$ is the Newton op. defined in (1).*
2. *For each coordinate $i = 1, \dots, n$, set $x_i^{\{k+1\}}$ to be equal to the maximum multiple of 2^{-h} which is $\leq \max(x_i^{\{k+1\}}, 0)$. (In other words, round down $x^{\{k+1\}}$ to the nearest multiple of 2^{-h} , while ensuring the result is non-negative.)*

Unfortunately, rounding can cause iterates $x^{[k]}$ to become unbalanced. Nevertheless, we can handle this. For any PPS, $x = P(x)$, with Jacobian matrix $B(x)$, and LFP q^* , $\rho(B(q^*)) \leq 1$ ([8, 7]). If $\rho(B(q^*)) < 1$, we call the PPS **non-critical**. Otherwise, if $\rho(B(q^*)) = 1$, we call the PPS **critical**. For SCFGs whose PPS $x = P_G(x)$ is non-critical, we get good bounds, even though R-NM iterates can become unbalanced:

Theorem 2. *For any $\epsilon > 0$, and for an SCFG, G , if the PPS $x = P_G(x)$ has LFP $0 < q^G \leq 1$ and $\rho(B_G(q^G)) < 1$, then if we use R-NM with parameter $h + 2$ to approximate the LFP solution of the MPS $y = P_{G \otimes D}(y)$, then $\|q^{G \otimes D} - y^{[h+1]}\|_\infty \leq \epsilon$ where $h := 14|G| + 3 + \lceil \log(1/\epsilon) + \log d \rceil$.*

Thus we can compute the probability $q_A^{G,D} = \sum_{t \in F} q_{s_0 A t}^{G \otimes D}$ within additive error $\delta > 0$ in time polynomial in the input size: $|G|$, $|D|$ and $\log(1/\delta)$, in the standard Turing model of computation.

We in fact obtain a much more general result. For any SCFG, G , and corresponding PPS, $x = P_G(x)$, with LFP $q^* > 0$, the *dependency graph*, $H_G = (V, E)$, has the variables (or the nonterminals of G) as nodes and has the following edges: $(x_i, x_j) \in E$ iff x_j appears in some monomial in $P_G(x)_i$ with a positive coefficient. We can decompose the dependency graph H_G into its SCCs, and form the DAG of SCCs, H'_G . For each SCC, \mathcal{S} , suppose its corresponding equations are $x_{\mathcal{S}} = P_G(x_{\mathcal{S}}, x_{D(\mathcal{S})})_{\mathcal{S}}$, where $D(\mathcal{S})$ is the set of variables $x_j \notin \mathcal{S}$ such that there is a path in H_G from some variable $x_i \in \mathcal{S}$ to x_j . We call a SCC, \mathcal{S} , of H_G , a **critical SCC** if the PPS $x_{\mathcal{S}} = P_G(x_{\mathcal{S}}, q_{D(\mathcal{S})}^G)_{\mathcal{S}}$ is critical. In other words, the SCC \mathcal{S} is critical if we plug in the LFP values q^G into variables that are in lower SCCs, $D(\mathcal{S})$, then the resulting PPS is critical. We note that an arbitrary PPS, $x = P_G(x)$ is non-critical if and only if it has no critical SCC. We define the **critical depth**, $\mathbf{c}(G)$, of $x = P_G(x)$ as follows: it is the maximum length, k , of any sequence $\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_k$, of SCCs of H_G , such that for all $i \in \{1, \dots, k-1\}$, $\mathcal{S}_{i+1} \subseteq D(\mathcal{S}_i)$, and furthermore, such that for all $j \in \{1, \dots, k\}$, \mathcal{S}_j is critical. Let us call a critical SCC, \mathcal{S} , of H_G a **bottom-critical SCC**, if $D(\mathcal{S})$ does not contain any critical SCCs. By using earlier results ([8, 3]) we can compute in P-time the critical SCCs of a PPS, and its critical depth (see the appendix).

PPSs with nested critical SCCs are hard to analyze directly. It turns out we can circumvent this by “tweaking” the probabilities in the SCFG G to obtain an SCFG G' with no critical SCCs, and showing that the “tweaks” are small enough so that they do not change the probabilities of interest by much. Concretely:

Theorem 3. *For any $\epsilon > 0$, and for any SCFG, G , in SNF form, with $q^G > 0$, with critical depth $\mathfrak{c}(G)$, consider the new SCFG, G' , obtained from G by the following process: for each bottom-critical SCC, \mathcal{S} , of $x = P_G(x)$, find any rule $r = A \xrightarrow{p} B$ of G , such that A and B are both in \mathcal{S} (since G is in SNF, such a rule must exist in every critical SCC). Reduce the probability p , by setting it to $p' = p(1 - 2^{-(14|G|+3)2^{\mathfrak{c}(G)}}\epsilon^{2^{\mathfrak{c}(G)}})$. Do this for all bottom-critical SCCs. This defines G' , which is non-critical. Using G' instead of G , if we apply R-NM, with parameter $h + 2$ to approximate the LFP $q^{G' \otimes D}$ of MPS $y = P_{G' \otimes D}(y)$, then $\|q^{G \otimes D} - x^{[h+1]}\|_\infty \leq \epsilon$ where $h := \lceil \log d + (3 \cdot 2^{\mathfrak{c}(G)} + 1)(\log(1/\epsilon) + 14|G| + 3) \rceil$. Thus we can compute $q_A^{G,D} = \sum_{t \in F} q_{s_0 A t}^{G \otimes D}$ within additive error $\delta > 0$ in time polynomial in: $|G|$, $|D|$, $\log(1/\delta)$, and $2^{\mathfrak{c}(G)}$, in the Turing model of computation.*

The proof is very involved, and is in the appendix. There, we also give a family of SCFGs, and a 3-state DFA that checks the infix probability of string aa , and we explain why these examples indicate it will likely be difficult to overcome the exponential dependence on the critical-depth $\mathfrak{c}(G)$ in the above bounds.

5 Non-criticality of SCFGs obtained by EM

In doing parameter estimation for SCFGs, in either the supervised or unsupervised (EM) settings (see, e.g., [17]), we are given a CFG, \mathcal{H} , with start nonterminal S , and we wish to extend it to an SCFG, G , by giving probabilities to the rules of \mathcal{H} . We also have some probability distribution, $\mathcal{P}(\pi)$, over the complete derivations, π , of \mathcal{H} that start at start non-terminal S . (In the unsupervised case, we begin with an SCFG, and the distribution \mathcal{P} arises from the prior rule probabilities, and from the training corpus of strings.) We then assign each rule of \mathcal{H} a (new) probability as follows to obtain (or update) G :

$$p(A \rightarrow \gamma) := \frac{\sum_{\pi} \mathcal{P}(\pi) C(A \rightarrow \gamma, \pi)}{\sum_{\pi} \mathcal{P}(\pi) C(A, \pi)} \quad (2)$$

where $C(r, \pi)$ is the number of times the rule r is used in the complete derivation π , and $C(A, \pi) = \sum_{r \in R_A} C(r, \pi)$. Equation (2) only makes sense when the sums $\sum_{\pi} \mathcal{P}(\pi) C(A, \pi)$ are finite and nonzero, which we assume; we also assume every non-terminal and rule of \mathcal{H} appears in some complete derivation π with $\mathcal{P}(\pi) > 0$.

Proposition 4. *If we use parameter estimation to obtain SCFG G using equation (2), under the stated assumptions, then G is consistent³, i.e. $q^G = \mathbf{1}$, and furthermore the PPS $x = P_G(x)$ is non-critical, i.e., $\rho(B_G(\mathbf{1})) < 1$.*

It follows from Prop. 4 and Thm. 2, that for SCFGs obtained by parameter estimation and EM, we can compute the probability $q_A^{G,D}$ of generating a string in $L(D)$ to within any desired precision in P-time, for any DFA D .

³ Consistency of the obtained SCFGs is well-known; see, e.g., [15, 17] & references therein; also [19] has results related to Prop. 4 for restricted grammars.

References

- [1] A. Corazza, R. De Mori, D. Gretter, and G. Satta. Computation of probabilities for an island-driven parser. *IEEE Trans. PAMI*, 13(9):936–950, 1991.
- [2] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probabilistic models of Proteins and Nucleic Acids*. Cambridge U. Press, 1999.
- [3] J. Esparza, A. Gaiser, and S. Kiefer. Computing least fixed points of probabilistic systems of polynomials. In *Proc. 27th STACS*, pages 359–370, 2010.
- [4] J. Esparza, S. Kiefer, and M. Luttenberger. Computing the least fixed point of positive polynomial systems. *SIAM J. on Computing*, 39(6):2282–2355, 2010.
- [5] J. Esparza, A. Kučera, and R. Mayr. Model checking probabilistic pushdown automata. *Logical Methods in Computer Science*, 2(1):1 – 31, 2006.
- [6] K. Etessami, A. Stewart, and M. Yannakakis. Polynomial-time algorithms for branching Markov decision processes and probabilistic min(max) polynomial Bellman equations. In *ICALP*, 2012. See full version at arXiv:1202.4798.
- [7] K. Etessami, A. Stewart, and M. Yannakakis. Polynomial-time algorithms for multi-type branching processes and stochastic context-free grammars. In *Proc. 44th ACM STOC*, 2012. Full version is available at arXiv:1201.2374.
- [8] K. Etessami and M. Yannakakis. Recursive Markov chains, stochastic grammars, and monotone systems of nonlinear equations. *Journal of the ACM*, 56(1), 2009.
- [9] K. Etessami and M. Yannakakis. Model checking of recursive probabilistic systems. *ACM Trans. Comput. Log.*, 13(2):12, 2012.
- [10] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge U. Press, 1985.
- [11] F. Jelinek and J. D. Lafferty. Computation of the probability of initial substring generation by stochastic context-free grammars. *Computational Linguistics*, 17(3):315–323, 1991.
- [12] B. Knudsen and J. Hein. Pfold: RNA secondary structure prediction using stochastic context-free grammars. *Nucleic Acids Res.*, 31:3423–3428, 2003.
- [13] P. Lancaster and M. Tismenetsky. *The Theory of Matrices*. Academic Press, 2nd edition, 1985.
- [14] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.
- [15] M.-J. Nederhof and G. Satta. Estimation of consistent probabilistic context-free grammars. In *HLT-NAACL*, 2006.
- [16] M.-J. Nederhof and G. Satta. Computing partition functions of PCFGs. *Research on Language and Computation*, 6(2):139–162, 2008.
- [17] M.-J. Nederhof and G. Satta. Probabilistic parsing. *New Developments in Formal Languages and Applications*, 113:229–258, 2008.
- [18] M.-J. Nederhof and G. Satta. Computation of infix probabilities for probabilistic context-free grammars. In *EMNLP*, pages 1213–1221, 2011.
- [19] J. Sánchez and J.-M. Benedí. Consistency of stochastic context-free grammars from probabilistic estimation based on growth transformations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(9):1052–1055, 1997.
- [20] A. Stewart, K. Etessami, and M. Yannakakis. Upper bounds for Newton’s method on monotone polynomial systems, and P-time model checking of probabilistic one-counter automata. arXiv:1302.3741 (submitted for publication), 2013.
- [21] A. Stolcke. An efficient probabilistic context-free parsing algorithm that computes prefix probabilities. *Computational Linguistics*, 21(2):167–201, 1995.
- [22] D. Wojtczak and K. Etessami. Premo: an analyzer for probabilistic recursive models. In *Proc. 13th TACAS*, pages 66–71, 2007.

A Proof of Theorem 1 (and of Lemma 1).

Theorem 1. *Let $x = P_G(x)$ be any PPS (or MPS), with n variables, associated with a SCFG (or WCFG) G , and let $y = P_{G \otimes D}(y)$ be the corresponding product MPS, for any DFA D , with d states. For any balanced vector $y \in \mathfrak{B} \subseteq \mathbb{R}^{d^2 n}$, with $y \geq 0$, $\rho(B_{G \otimes D}(y)) = \rho(B_G(\mathfrak{C}(y)))$. Furthermore, if $\rho(B_{G \otimes D}(y)) < 1$, then $\mathcal{N}_{G \otimes D}(y)$ is defined and balanced, $\mathcal{N}_G(\mathfrak{C}(y))$ is defined, and $\mathfrak{C}(\mathcal{N}_{G \otimes D}(y)) = \mathcal{N}_G(\mathfrak{C}(y))$. Thus, $\mathcal{N}_{G \otimes D}$ preserves balance, and the collapse map \mathfrak{C} “commutes” with \mathcal{N} over non-negative balanced vectors, irrespective of what the DFA D is.*

We establish this via a series of lemmas that reveal many algebraic and analytic properties of balance, collapse, and their interplay with Newton’s method. Lemma 2 first establishes a series of algebraic and analytic properties of arbitrary balanced vectors and matrices. Lemma 1 then uses these to establish properties of the specific balanced matrices and vectors arising during iterations of Newton’s method on PPSs (and MPSs), and on corresponding product MPSs. Theorem 1 is an immediate consequence of Lemma 1, parts (i)&(iv), below.

Lemma 2. *Consider the set $\mathfrak{B} \subseteq \mathbb{R}^{d^2 n}$ of balanced vectors, and the set $\mathfrak{B}^\times \subseteq \mathbb{R}^{d^2 n \times d^2 n}$ of balanced matrices. Let $\mathfrak{B}_{\geq 0} = \mathfrak{B} \cap \mathbb{R}_{\geq 0}^{d^2 n}$ and $\mathfrak{B}_{\geq 0}^\times = \mathfrak{B} \cap \mathbb{R}_{\geq 0}^{d^2 n \times d^2 n}$.*

- (i) *\mathfrak{B} and \mathfrak{B}^\times are both closed under linear combinations. In other words: $\sum_i \alpha_i v^{(i)} \in \mathfrak{B}$ and $\sum_i \alpha_i M^{(i)} \in \mathfrak{B}^\times$, if, $\forall i$, $v^{(i)} \in \mathfrak{B}$ and $M^{(i)} \in \mathfrak{B}^\times$. Furthermore, \mathfrak{C} is a linear map on both \mathfrak{B} and \mathfrak{B}^\times . In other words: $\mathfrak{C}(\sum_i \alpha_i v^{(i)}) = \sum_i \alpha_i \mathfrak{C}(v^{(i)})$ and $\mathfrak{C}(\sum_i \alpha_i M^{(i)}) = \sum_i \alpha_i \mathfrak{C}(M^{(i)})$, whenever, $\forall i$, $\alpha_i \in \mathbb{R}$, $v^{(i)} \in \mathfrak{B}$, and $M^{(i)} \in \mathfrak{B}^\times$.*
- (ii) *If $M \in \mathfrak{B}^\times$ and $v \in \mathfrak{B}$, then $Mv \in \mathfrak{B}$ and $\mathfrak{C}(Mv) = \mathfrak{C}(M)\mathfrak{C}(v)$.*
- (iii) *If $M, M' \in \mathfrak{B}^\times$, then $MM' \in \mathfrak{B}^\times$ and $\mathfrak{C}(MM') = \mathfrak{C}(M)\mathfrak{C}(M')$.*
- (iv) *If $M \in \mathfrak{B}_{\geq 0}^\times$, and $v \in \mathbb{R}^{d^2 n}$ is any vector, then $\mathfrak{C}(Mv) \geq \mathfrak{C}(M)\mathfrak{C}(v)$, where we extend the map \mathfrak{C} to arbitrary $v' \in \mathbb{R}^{d^2 n}$ by letting $\mathfrak{C}(v')_A := \min_s \sum_t v'_{(sAt)}$.*
- (v) *If $M \in \mathfrak{B}_{\geq 0}^\times$, then $\rho(M) = \rho(\mathfrak{C}(M))$. In other words, the collapse operator \mathfrak{C} preserves the spectral radius of balanced non-negative matrices.*
- (vi) *If $v \in \mathfrak{B}_{\geq 0}$, then $\|v\|_\infty \leq \|\mathfrak{C}(v)\|_\infty$. If $M \in \mathfrak{B}_{\geq 0}^\times$ then $\|M\|_\infty \leq d\|\mathfrak{C}(M)\|_\infty$.*

Proof.

(i): This can be verified directly from the definitions of balance and collapse. In particular, for any nonterminal $A \in V$, and any states $s, s' \in Q$:

$$\begin{aligned}
 \sum_t \left(\sum_i \alpha_i v^{(i)} \right)_{(sAt)} &= \sum_i \alpha_i \sum_t v_{(sAt)}^{(i)} \\
 &= \sum_i \alpha_i \mathfrak{C}(v^{(i)})_A \quad (\text{because every } v^{(i)} \text{ is balanced}) \\
 &= \sum_i \alpha_i \sum_t v_{(s'At)}^{(i)} \\
 &= \sum_t \left(\sum_i \alpha_i v^{(i)} \right)_{(s'At)}
 \end{aligned}$$

Also, we have $\mathfrak{C}(\sum_i \alpha_i v^{(i)})_A := \sum_t (\sum_i \alpha_i v^{(i)})_{(sAt)} = \sum_i \alpha_i \mathfrak{C}(v^{(i)})_A$.
Likewise, for any nonterminals $B, C \in V$, and any states $s, u \in Q$ and $v, v' \in Q$:

$$\begin{aligned} \sum_t (\sum_i \alpha_i M^{(i)})_{(sBt), (uCv)} &= \sum_i \alpha_i \sum_t M_{(sBt), (uCv)}^{(i)} \\ &= \sum_i \alpha_i \sum_t M_{(sBt), (uCv')}^{(i)} \quad (\text{because every } M^{(i)} \text{ is balanced}) \\ &= \sum_t (\sum_i \alpha_i M^{(i)})_{(sBt), (uCv')} \end{aligned}$$

Similarly, for any nonterminals B, C , and any states $s, v, s', v' \in Q$:

$$\begin{aligned} \sum_{t,u} (\sum_i \alpha_i M^{(i)})_{(sBt), (uCv)} &= \sum_i \alpha_i \sum_{t,u} M_{(sBt), (uCv)}^{(i)} \\ &= \sum_i \alpha_i \sum_{t,u} M_{(s'Bt), (uCv')}^{(i)} \quad (\text{because every } M^{(i)} \text{ is balanced}) \\ &= \sum_{t,u} (\sum_i \alpha_i M^{(i)})_{(s'Bt), (uCv')} \end{aligned}$$

Now, $\mathfrak{C}(\sum_i \alpha_i M^{(i)})_{B,C} := \sum_{t,u} (\sum_i \alpha_i M^{(i)})_{(sBt), (uCv)} = \sum_i \alpha_i \sum_{t,u} M_{(sBt), (uCv)}^{(i)} = \sum_i \alpha_i \mathfrak{C}(M^{(i)})_{B,C}$.

(ii): For any non-terminal B and state s :

$$\begin{aligned} \sum_t (Mv)_{(sBt)} &= \sum_{t,u,C,z} M_{(sBt), (uCz)} v_{uCz} \\ &= \sum_{u,C,z} (\sum_t M_{(sBt), (uCz)}) v_{uCz} \\ &= \sum_{C,u} (\sum_t M_{(sBt), (uCz)}) \sum_z v_{uCz} \quad (\text{since } M \text{ is balanced}) \\ &= \sum_{C,u} (\sum_t M_{(sBt), (uCz)}) \mathfrak{C}(v)_C \quad (\text{since } v \text{ is balanced}) \\ &= \sum_C (\sum_{t,u} M_{(sBt), (uCz)}) \mathfrak{C}(v)_C \\ &= \sum_C \mathfrak{C}(M)_{B,C} \mathfrak{C}(v)_C \quad (\text{since } M \text{ is balanced}) \\ &= (\mathfrak{C}(M) \mathfrak{C}(v))_B \end{aligned}$$

which is independent of s . So $\mathfrak{C}(Mv)_B = \sum_t (Mv)_{(sBt)} = (\mathfrak{C}(M) \mathfrak{C}(v))_B$.

(iii): For any non-terminal D, E , and states $s, w, x \in Q$:

$$\begin{aligned}
\sum_t (MM')_{(sDt), (wEx)} &= \sum_{t, u, C, v} M_{(sDt), (uCv)} M'_{(uCv), (wEx)} \\
&= \sum_{u, C, v} \left(\sum_t M_{(sDt), (uCv)} \right) M'_{(uCv), (wEx)} \\
&= \sum_{C, u} \left(\sum_t M_{(sDt), (uCv)} \right) \sum_v M'_{(uCv), (wEx)} \quad (\text{since } M \text{ is balanced})
\end{aligned}$$

Since $M' \in \mathfrak{B}^\times$, the last sum is independent of x , which is what we aimed to show. Next consider:

$$\begin{aligned}
\sum_{t, w} (MM')_{(sDt), (wEx)} &= \sum_{t, w, u, C, v} M_{(sDt), (uCv)} M'_{(uCv), (wEx)} \\
&= \sum_{u, w, C, v} \left(\sum_t M_{(sDt), (uCv)} \right) M'_{(uCv), (wEx)} \\
&= \sum_{C, u, w} \left(\sum_t M_{(sDt), (uCv)} \right) \sum_v M'_{(uCv), (wEx)} \quad (\text{since } M \text{ is balanced}) \\
&= \sum_{C, u} \left(\sum_t M_{(sDt), (uCv)} \right) \sum_{v, w} M'_{(uCv), (wEx)} \\
&= \sum_{C, u} \left(\sum_t M_{(sDt), (uCv)} \right) \mathfrak{C}(M')_{C, E} \quad (\text{since } M' \text{ is balanced}) \\
&= \sum_C \mathfrak{C}(M)_{D, C} \mathfrak{C}(M')_{C, E} \quad (\text{since } B \text{ is balanced}) \\
&= (\mathfrak{C}(M) \mathfrak{C}(M'))_{D, E}
\end{aligned}$$

So, $\sum_{t, w} (MM')_{(sDt), (wEx)}$ is independent of s, x and $\mathfrak{C}(MM')_{D, E} = \sum_{t, w} (MM')_{(sDt), (wEx)} = (\mathfrak{C}(M) \mathfrak{C}(M'))_{D, E}$, for any $D, E \in V$.

(iv): For any non-terminal B and state s :

$$\begin{aligned}
\sum_t (Mv)_{(sBt)} &= \sum_{t, u, C, z} M_{(sBt), (uCz)} v_{uCz} \\
&= \sum_{u, C, z} \left(\sum_t M_{(sBt), (uCz)} \right) v_{uCz} \\
&= \sum_{C, u} \left(\sum_t M_{(sBt), (uCz)} \right) \sum_z v_{uCz} \quad (\text{since } M \text{ is balanced}) \\
&\geq \sum_{C, u} \left(\sum_t M_{(sBt), (uCz)} \right) \min_u \sum_z v_{uCz} \quad (\text{since } (\sum_t M_{(sBt), (uCz)}) \geq 0 \text{ for any } C, u) \\
&= \sum_C \mathfrak{C}(M)_{B, C} \mathfrak{C}(v)_C = (\mathfrak{C}(M) \mathfrak{C}(v))_B
\end{aligned}$$

Since this holds for any B and any s , $\mathfrak{C}(Mv)_B = \min_s \sum_t (Mv)_{(sBt)} \geq (\mathfrak{C}(M) \mathfrak{C}(v))_B$.

(vi): (we will prove part (v) below) Since $v \in \mathfrak{B}_{\geq 0}$, $v_{(sAt)} \leq \sum_{t'} v_{(sAt')} = \mathfrak{C}(v)_A$ so $\|v\|_\infty \leq \|\mathfrak{C}(v)\|_\infty$. For $M \in \mathfrak{B}_{\geq 0}^\times$:

$$\begin{aligned}
\|M\|_\infty &= \max_{s,B,t} \sum_{u,C,v} M_{(sBt),(uCv)} \\
&\leq \max_{s,B} \sum_{u,C,v,t} M_{(sBt),(uCv)} \\
&= \max_{s,B} \sum_{C,v} \mathfrak{C}(M)_{B,C} \\
&= \max_B d \sum_C \mathfrak{C}(M)_{B,C} \\
&= d \|\mathfrak{C}(M)\|_\infty
\end{aligned}$$

(v): By standard facts from Perron-Frobenius theory (see e.g. Theorem 8.3.1 of [10]), the non-negative matrix $\mathfrak{C}(M)$, has as an eigenvalue $\rho(\mathfrak{C}(M))$ associated with which is a non-negative eigenvector $v_G \neq 0$. That is $\mathfrak{C}(M)v_G = \rho(\mathfrak{C}(M))v_G$ for some non-zero $v_G \geq 0$. Now consider any non-negative balanced vector u with $\mathfrak{C}(u) = v_G$. (Such a u obviously exists.) Let $f(u) = \frac{1}{\rho(\mathfrak{C}(M))}Mu$. By part (ii), Mu is balanced and $\mathfrak{C}(Mu) = \mathfrak{C}(M)v_G = \rho(\mathfrak{C}(M))v_G$. So, $f(u)$ is non-negative and balanced and has $\mathfrak{C}(f(u)) = v_G$. The set of non-negative balanced vector u with $\mathfrak{C}(u) = v_G$ is compact (it is a product of simplices) and the continuous function f maps this set into itself. So by *Brouwer's fixed point theorem*, f has a fixed point, that is a u^* with $u^* = \frac{1}{\rho(\mathfrak{C}(M))}Mu^*$. That is, u^* is an eigenvector of M with eigenvalue $\rho(\mathfrak{C}(M))$. So $\rho(M) \geq \rho(\mathfrak{C}(M))$.

In the other direction, we use the fact (see, e.g., Theorem 5.6.12 of [10]) that for any square matrix N , $\lim_{k \rightarrow \infty} \|N^k\|_\infty = 0$ if and only if $\rho(N) < 1$.

Now for $M \in \mathfrak{B}_{\geq 0}^\times$ assume, for contradiction, that $\rho(M) > \rho(\mathfrak{C}(M))$. Then $\rho(\frac{1}{\rho(M)}M) = \frac{1}{\rho(M)}\rho(M) = 1 > \frac{1}{\rho(M)}\rho(\mathfrak{C}(M)) = \rho(\frac{1}{\rho(M)}\mathfrak{C}(M))$. Thus, by the above fact from matrix theory, we have that $\lim_{k \rightarrow \infty} \|(\frac{1}{\rho(M)}\mathfrak{C}(M))^k\|_\infty = 0$.

But for any $k \geq 1$,

$$\begin{aligned}
0 &\leq \|(\frac{1}{\rho(M)}M)^k\|_\infty \leq d \|\mathfrak{C}((\frac{1}{\rho(M)}M)^k)\|_\infty \quad (\text{by part (vi)}) \\
&= d \|\mathfrak{C}(\frac{1}{\rho(M)}M)^k\|_\infty \quad (\text{by part (iii)}) \\
&= d \|(\frac{1}{\rho(M)}\mathfrak{C}(M))^k\|_\infty \quad (\text{by part (i)})
\end{aligned}$$

And thus, since the right hand side goes to 0 as $k \rightarrow \infty$, we must also have $\lim_{k \rightarrow \infty} \|(\frac{1}{\rho(M)}M)^k\|_\infty = 0$, but this is a contradiction, because $\rho(\frac{1}{\rho(M)}M) = 1$. So, our assumption $\rho(M) > \rho(\mathfrak{C}(M))$ must be false.

Having established both directions, we conclude that $\rho(M) = \rho(\mathfrak{C}(M))$. \square

Lemma 1. Let $\mathfrak{B}_{\geq 0} = \mathfrak{B} \cap \mathbb{R}_{\geq 0}^{d^2 n}$ and $\mathfrak{B}_{\geq 0}^\times = \mathfrak{B} \cap \mathbb{R}_{\geq 0}^{d^2 n \times d^2 n}$.

Let $B_G(x)$ denote the Jacobian of the PPS (or MPS) $x = P_G(x)$, and let $B_{G \otimes D}(y)$ be the Jacobian of MPS $y = P_{G \otimes D}(y)$.

Then $q^{G \otimes D} \in \mathfrak{B}_{\geq 0}$ and $\mathfrak{C}(q^{G \otimes D}) = q^G$, and:

- (i) If $y \in \mathfrak{B}_{\geq 0} \subseteq \mathbb{R}_{\geq 0}^{d^2 n}$ then $B_{G \otimes D}(y) \in \mathfrak{B}_{\geq 0}^\times$, and $\mathfrak{C}(B_{G \otimes D}(y)) = B_G(\mathfrak{C}(y))$.
- (ii) If $y \in \mathfrak{B}_{\geq 0}$, then $P_{G \otimes D}(y) \in \mathfrak{B}_{\geq 0}$, and $\mathfrak{C}(P_{G \otimes D}(y)) = P_G(\mathfrak{C}(y))$.
- (iii) If $y \in \mathfrak{B}_{\geq 0}$ and $\rho(B_G(\mathfrak{C}(y))) < 1$, then $I - B_{G \otimes D}(y)$ is non-singular, $(I - B_{G \otimes D}(y))^{-1} \in \mathfrak{B}_{\geq 0}^\times$, and $\mathfrak{C}((I - B_{G \otimes D}(y))^{-1}) = (I - B_G(\mathfrak{C}(y)))^{-1}$.
- (iv) If $y \in \mathfrak{B}_{\geq 0}$ and $\rho(B_G(\mathfrak{C}(y))) < 1$, then $\mathcal{N}_{G \otimes D}(y) \in \mathfrak{B}^\times$ and $\mathfrak{C}(\mathcal{N}_{G \otimes D}(y)) = \mathcal{N}_G(\mathfrak{C}(y))$.

Proof.

Firstly, let us recall why $q^{G \otimes D} \in \mathfrak{B}_{\geq 0}$ and $\mathfrak{C}(q^{G \otimes D}) = q^G$. Recall these are the LFP q^G of $x = P_G(x)$, and the LFP $q^{G \otimes D}$ of $y = P_{G \otimes D}(y)$. By Propositions 1 and 2, for any nonterminal $A \in V$, $q_A^G = \sum_{w \in \Sigma^*} q_A^{G,w}$ is the probability (weight) that G generates any finite string w . Likewise $q_{(sAt)}^{G \otimes D} = \sum_{\{w \mid \Delta^*(s,w)=t\}} q_A^{G,w}$ is the probability (weight) that, starting at A , G generates a finite string w such that $\Delta^*(s,w) = t$. Thus, clearly, for any $A \in V$, and any $s \in Q$, $q_A^G = \sum_{t \in Q} q_{(sAt)}^{G \otimes D} = \mathfrak{C}(q^{G \otimes D})_A$. Now we prove the enumerated assertions one by one:

(i): We need to argue both that $B_{G \otimes D}(y) \in \mathfrak{B}_{\geq 0}^\times$, and that $\mathfrak{C}(B_{G \otimes D}(y)) = B_G(\mathfrak{C}(y))$, for $y \in \mathfrak{B}_{\geq 0}$. Again, recall that we are assuming wlog that G is in SNF form. We split the proof into cases depending on the type of non-terminal A in $B_{G \otimes D}(y)_{(sAt),(uEv)}$. Let $\delta_{\alpha,\beta}$ denote the Dirac function: $\delta_{\alpha,\beta} := 1$ if $\alpha = \beta$, and $\delta_{\alpha,\beta} := 0$ if $\alpha \neq \beta$.

Type Q: For any non-terminal A of type Q, the only rule in R_A has the form $A \xrightarrow{1} BC$, and $P_G(x)_A \equiv x_B x_C$. And, for any states $s, t \in Q$, $P_{G \otimes D}(y)_{(sAt)} \equiv \sum_{w \in Q} y_{(sBw)} y_{(wCt)}$. Thus

$$B_{G \otimes D}(y)_{(sAt),(uEv)} \doteq \frac{\partial P_{G \otimes D}(y)_{(sAt)}}{\partial y_{(uEv)}} = \delta_{t,v} \cdot \delta_{E,C} \cdot y_{(sBu)} + \delta_{s,u} \cdot \delta_{E,B} \cdot y_{(vCt)}$$

Thus

$$\sum_t B_{G \otimes D}(y)_{(sAt),(uEv)} = \delta_{E,C} \cdot y_{(sBu)} + \delta_{s,u} \cdot \delta_{E,B} \cdot \sum_t y_{(vCt)}$$

Since y is balanced, $\sum_t y_{(vCt)}$ is independent of v , so $\sum_t B_{(sAt),(uEv)}$ is independent of v . Next we note that:

$$\sum_{t,u} B_{G \otimes D}(y)_{(sAt),(uEv)} = \delta_{E,C} \sum_u y_{(sBu)} + \delta_{E,B} \sum_t y_{(vCt)}$$

Thus

$$\sum_{t,u} B_{G \otimes D}(y)_{(sAt),(uEv)} = \delta_{E,C} \mathfrak{C}(y)_B + \delta_{E,B} \mathfrak{C}(y)_C = B_G(\mathfrak{C}(y))$$

Type T: For any non-terminal A of type T, $P_G(x)_A$ does not depend on x , and $P_{G \otimes D}(y)_{sAt}$ does not depend on y , for any $s, t \in Q$. Thus $\sum_t B_{G \otimes D}(y)_{(sAt), (uCv)} = 0$, and $\sum_{t,u} B_{G \otimes D}(y)_{(sAt), (uCv)} = 0 = B_G(\mathfrak{C}(y))_{A,C}$.

Type L: For any non-terminal A of type L, recall that $P_G(x)_A = \sum_{r \in R_A} p_r x B_r$. And for any states s, t , $P_{G \otimes D}(y)_{(sAt)} = \sum_{r \in R_A} p_r y_{(sB_r t)}$.

Thus, all the entries of $B_G(x)_{A,C}$ and $B_{G \otimes D}(y)_{(sAt), (uCv)}$ are independent of x and y , respectively. And

$$B_{G \otimes D}(y)_{(sAt), (uCv)} = \frac{\partial P_{G \otimes D}(y)_{(sAt)}}{\partial y_{(uCv)}} = \delta_{s,u} \cdot \delta_{t,v} \cdot B_G(x)_{A,C}$$

Consequently $\sum_t B_{G \otimes D}(y)_{(sAt), (uCv)} = \delta_{s,u} B_G(x)_{A,C}$, which is independent of v . And, $\sum_{t,u} B_{G \otimes D}(y)_{(sAt), (uCv)} = B_G(x)_{A,C}$, which is independent of s and v , and $B_G(x)_{A,C} = B_G(\mathfrak{C}(y))_{A,C}$, because $B_G(x)_{A,C}$ is independent of x .

Having shown that for all nonterminals A and C , and all nonterminals $s, u \in Q$, the sum $\sum_t B_{G \otimes D}(y)_{(sAt), (uCv)}$ is independent of v . And we have also shown that for all nonterminals A and C , the sum $\sum_{t,u} B_{G \otimes D}(y)_{(sAt), (uCv)}$ is independent of s and v , and furthermore, that the latter sum (which is by definition $\mathfrak{C}(B_{G \otimes D}(y))_{A,C}$), is equal to $B_G(\mathfrak{C}(y))$. Thus our proof for part (i) is complete.

(ii): Part (ii) could be proved using a case-by-case analysis similar to part (i). Instead, we shall use part (i). Recall that $P_G(x)$ and $P_{D \otimes G}(y)$ have no polynomials of degree more than 2. Furthermore:

$$P_G(x) = P_G(0) + B_G\left(\frac{1}{2}x\right)x$$

And

$$P_{G \otimes D}(y) = P_{G \otimes D}(0) + B_{G \otimes D}\left(\frac{1}{2}y\right)y$$

By the previous parts of this Lemma, and by Lemma 2, we know that $B_{G \otimes D}(\frac{1}{2}y)y$ is balanced, and $\mathfrak{C}(B_{G \otimes D}(\frac{1}{2}y)y) = B_G(\frac{1}{2}\mathfrak{C}(y))\mathfrak{C}(y)$. All that remains is to show that $P_{G \otimes D}(0)$ is balanced and that $\mathfrak{C}(P_{G \otimes D}(0)) = P_G(0)$, and again use the properties established in Lemma 2.

Now, unless a non-terminal A has type T, $P_G(0)_A = 0$, and for any states $s, t \in Q$, $P_{G \otimes D}(0)_{(sAt)} = 0$. So, in these cases, there is nothing to prove. If the nonterminal A does have type T, then $P_G(x)_A = 1$. If there is a rule $A \xrightarrow{1} a$, for some $a \in \Sigma$, then for any state $s \in Q$, there is a unique state $t' \in Q$ with $\Delta(s, a) = t'$. If instead there is a rule $A \xrightarrow{1} \epsilon$, then let $t' := s$. In both cases, note that $\sum_t P_{G \otimes D}(y)_{(sAt)} = 1 = P_G(\mathfrak{C}(y))_A$, since $P_{G \otimes D}(y)_{(sAt)} = 1$ when $t = t'$ and $P_{G \otimes D}(y)_{(sAt)} = 0$ otherwise. Thus also $\mathfrak{C}(P_{G \otimes D}(y)) = P_G(\mathfrak{C}(y))$ in all cases.

(iii): By assumption, $\rho(B_G(\mathfrak{C}(y))) < 1$, so by Lemma 2 (iv), $\rho(B_{G \otimes D}(y)) < 1$. It is a basic fact that for any square $M \geq 0$ if $\rho(M) < 1$ then $(I - M)$ is non-singular

and $(I - M)^{-1} = \sum_{i=0}^{\infty} M^i$. (See, e.g., [13], Theorem 15.2.2, page 531). Thus $I - B_{G \otimes D}(y)$ is non-singular, and $(I - B_{G \otimes D}(y))^{-1} = \sum_{i=0}^{\infty} (B_{G \otimes D}(y))^i$. Note that each $(B_{G \otimes D}(y))^i$, for $i \geq 0$, is balanced, by using the previous parts of this Lemma and Lemma 2 (iii), and thus so are the partial sums $\sum_{i=0}^k (B_{G \otimes D}(y))^i$, for any $k \geq 0$. Therefore $(I - B_{G \otimes D}(y))^{-1} = \lim_{k \rightarrow \infty} \sum_{i=1}^k (B_{G \otimes D}(y))^i$ is a limit of balanced non-negative matrices. But then $(I - B_{G \otimes D}(y))^{-1}$ must be balanced, because the definition of balance for a matrix M requires equalities between continuous (in fact, linear) functions of the entries, and thus if all the matrices $\sum_{i=1}^k (B_{G \otimes D}(y))^i$ satisfy these conditions, then so does their limit.

Furthermore \mathfrak{C} is a linear and continuous function on matrices, so $\mathfrak{C}((I - B_{G \otimes D}(y))^{-1}) = \sum_{i=1}^{\infty} \mathfrak{C}(B_{G \otimes D}(y))^i = \sum_{i=1}^{\infty} \mathfrak{C}(B_{G \otimes D}(y))^i = (I - \mathfrak{C}(B_{G \otimes D}(y)))^{-1}$. By part (i) of this Lemma, this is equal to $(I - B_G(\mathfrak{C}(y)))^{-1}$. Done.

(iv): By part (ii) of this Lemma, $P_{G \otimes D}(y)$ is balanced and $\mathfrak{C}(P_{G \otimes D}(y)) = P_G(\mathfrak{C}(y))$. Part (iii) of this lemma says that $(I - B_{G \otimes D}(y))^{-1}$ is balanced and $\mathfrak{C}((I - B_{G \otimes D}(y))^{-1}) = (I - \mathfrak{C}(B_{G \otimes D}(y)))^{-1}$. Now we can apply the various algebraic properties of balanced vectors and matrices from Lemma 2 to conclude that

$$\mathcal{N}_{G \otimes D}(y) := y + (I - B_{G \otimes D}(y))^{-1}(P_{G \otimes D}(y) - y)$$

is balanced and that $\mathfrak{C}(\mathcal{N}_{G \otimes D}(y)) = \mathfrak{C}(y) + (I - B_G(\mathfrak{C}(y)))^{-1}(P_G(\mathfrak{C}(y)) - \mathfrak{C}(y)) = \mathcal{N}_G(\mathfrak{C}(y))$. \square

As mentioned already, Theorem 1 follows immediately from Lemma 1, parts (i)&(iv).

B Proofs for Section 4

We will first show how to compute in P-time the critical SCCs and the critical depth of a PPS. We then proceed to prove the main theorems of the section: Theorems 2 and 3.

Let $x = P(x)$ be a PPS (wlog in SNF), with LFP $q^* > 0$, let $B(x)$ be its Jacobean matrix, and let $H = (V, E)$ be its dependency graph. If B is a square matrix and I, J are subsets of indices, we will use $B_{I,J}$ to denote the submatrix with rows in I and columns in J , and we use B_I to denote the square submatrix $B_{I,I}$.

Proposition 5. *Given a PPS $x = P(x)$ with LFP $q^* > 0$, we can compute in polynomial time its critical SCCs and its critical depth.*

Proof. We know that for each SCC \mathcal{S} of H , either all the variables (nodes) of the SCC have value 1 in the LFP q^* , or they all have value < 1 ; moreover, if they have value 1, then so do all the variables that they can reach in H , i.e., $q_{\mathcal{S}}^* = 1$ implies $q_{D(\mathcal{S})}^* = 1$ [8]. Furthermore, we can determine which variables and SCCs have value 1, and which value < 1 , in polynomial time [8] (this was improved to strongly polynomial time in [3]). We also know that $\rho(B(q^*)) \leq 1$, thus a PPS

is critical iff $\rho(B(q^*)) = 1$. Furthermore, by Theorem 3.6 of [7], if $q^* < 1$, then $\rho(B(q^*)) < 1$.

Therefore, for each SCC \mathcal{S} , we can determine whether it is critical as follows. If $q_{\mathcal{S}}^* < 1$ then \mathcal{S} is not critical. If $q_{\mathcal{S}}^* = 1$, then \mathcal{S} is critical iff $\rho(B(\mathbf{1})_{\mathcal{S}}) = 1$, and it is not critical iff $\rho(B(\mathbf{1})_{\mathcal{S}}) < 1$; we can determine which of the two is the case as follows. Since the spectral radius of $B(\mathbf{1})_{\mathcal{S}}$ is at most 1, $\rho(B(\mathbf{1})_{\mathcal{S}}) = 1$ iff there is a vector $u \neq 0$ such that $(B(\mathbf{1})_{\mathcal{S}}) \cdot u = u$ (and we can take $u \geq 0$ to be an eigenvector for the eigenvalue 1 in this case since the matrix is nonnegative), or equivalently since the constraints are homogeneous in u , this is the case iff the set of linear equations $\{(B(\mathbf{1})_{\mathcal{S}}) \cdot u = u; \sum_i u_i = 1\}$ has a solution. This can be checked in (strongly) polynomial time by standard methods.

Once we have identified the critical SCCs, it is straightforward to compute the critical depth in linear time in the size of the DAG of SCCs by a traversal of the DAG in topological order. \square

Proposition 6. *A PPS $x = P(x)$ is critical if and only if at least one of its SCCs is critical.*

Proof. (Only if): Suppose first that the PPS is critical, i.e., that $\rho(B(q^*)) = 1$. Let $v \geq 0$, $v \neq 0$, be an eigenvector of $B(q^*)$ for the eigenvalue 1, i.e., $B(q^*)v = v$. Let \mathcal{S} be a lowest SCC that contains a variable with nonzero value in v , i.e. $v_{\mathcal{S}} \neq 0$ and $v_{D(\mathcal{S})} = 0$. Then $v_{\mathcal{S}} = B(q^*)_{\mathcal{S}, \mathcal{S} \cup D(\mathcal{S})} \cdot v_{\mathcal{S} \cup D(\mathcal{S})} = B(q^*)_{\mathcal{S}} \cdot v_{\mathcal{S}}$. Thus, $v_{\mathcal{S}}$ is an eigenvector of $B(q^*)_{\mathcal{S}}$ with eigenvalue 1, hence $\rho(B(q^*)_{\mathcal{S}}) \geq 1$, and since we always have $\rho(B(q^*)_{\mathcal{S}}) \leq 1$, it follows that \mathcal{S} is a critical SCC.

(If): Conversely, suppose that there is a critical SCC, and let \mathcal{S} be a highest critical SCC in the DAG of SCC's. Then $\rho(B(q^*)_{\mathcal{S}}) = 1$. Let $u \geq 0$ be an eigenvector of $B(q^*)_{\mathcal{S}}$ with eigenvalue 1. Let $E(\mathcal{S})$ be the (possibly empty) set of variables which depend on variables in \mathcal{S} but are not themselves in \mathcal{S} . If $E(\mathcal{S}) = \emptyset$ then let v be a vector with $v_{\mathcal{S}} = u$ and $v_i = 0$ for all variables $x_i \notin \mathcal{S}$. Then $B(q^*)v = v$, i.e., v is an eigenvector of $B(q^*)$ with eigenvalue 1, hence $\rho(B(q^*)) \geq 1$ and the PPS is critical.

Suppose that $E(\mathcal{S})$ is nonempty. Then $E(\mathcal{S})$ contains no critical SCCs by our choice of \mathcal{S} . This implies by our proof above for the (only if) direction that the PPS $x_{E(\mathcal{S})} = P(x_{E(\mathcal{S})}, x_{D(E(\mathcal{S}))})$ is not critical, i.e., $\rho(B(q^*)_{E(\mathcal{S})}) < 1$. Thus, $(I - B(q^*)_{E(\mathcal{S})})^{-1}$ exists. Let v be the vector with $v_{\mathcal{S}} = u$, $v_{E(\mathcal{S})} = (I - B(q^*)_{E(\mathcal{S})})^{-1} B(q^*)_{E(\mathcal{S}), \mathcal{S}} \cdot v_{\mathcal{S}}$ and $v_i = 0$ for all x_i not in either \mathcal{S} or $E(\mathcal{S})$.

We claim that $B(q^*)v = v$. If x_i does not depend on a variable in \mathcal{S} , then any x_j which x_i depends on also does not depend on \mathcal{S} and so has $v_j = 0$. So $(B(q^*)v)_i = 0 = v_i$. Next we consider $(B(q^*)v)_{\mathcal{S}}$. Since $D(\mathcal{S})$ is disjoint from \mathcal{S} and $E(\mathcal{S})$, $v_{D(\mathcal{S})} = 0$. So $(B(q^*)v)_{\mathcal{S}} = (B(q^*))_{\mathcal{S}} \cdot v_{\mathcal{S}} = v_{\mathcal{S}}$. Lastly consider $(B(q^*)v)_{E(\mathcal{S})}$.

$$\begin{aligned} (B(q^*)v)_{E(\mathcal{S})} &= B(q^*)_{E(\mathcal{S})} \cdot v_{E(\mathcal{S})} + B(q^*)_{E(\mathcal{S}), \mathcal{S}} \cdot v_{\mathcal{S}} \\ &= v_{E(\mathcal{S})} - (I - B(q^*)_{E(\mathcal{S})}) \cdot v_{E(\mathcal{S})} + B(q^*)_{E(\mathcal{S}), \mathcal{S}} \cdot v_{\mathcal{S}} \\ &= v_{E(\mathcal{S})} - B(q^*)_{E(\mathcal{S}), \mathcal{S}} \cdot v_{\mathcal{S}} + B(q^*)_{E(\mathcal{S}), \mathcal{S}} \cdot v_{\mathcal{S}} \\ &= v_{E(\mathcal{S})} \end{aligned}$$

So $B(q^*)v = v$. Therefore, $\rho(B(q^*)) \geq 1$ and hence the PPS is critical. \square

In the remainder of this section we will prove Theorem 3, and along the way, we will also establish Theorem 2. The proof of Theorem 3 is long and involved. We first need to recall, and establish, a series of Lemmas and Theorems.

Lemma 3. (Lemma C.3 of [6]) *If A is a non-negative matrix, and vector $u > 0$ is such that $Au \leq u$ and $\|u\|_\infty \leq 1$, and $\alpha, \beta \in (0, 1)$ are constants such that for every $i \in \{1, \dots, n\}$, one of the following two conditions holds:*

- (I) $(Au)_i \leq (1 - \beta)u_i$
- (II) *there is some k , $1 \leq k \leq n$, and some j , such that $(A^k)_{ij} \geq \alpha$ and $(Au)_j \leq (1 - \beta)u_j$.*

then $(I - A)$ is non-singular, $\rho(A) < 1$,⁴ and

$$\|(I - A)^{-1}\|_\infty \leq \frac{n}{u_{\min}^2 \alpha \beta}$$

Lemma 4. (Lemma A.4 of [7]) *Let A be a non-singular $n \times n$ matrix with rational entries. If the product of the denominators of all these entries is m , then*

$$\|A^{-1}\|_\infty \leq nm \|A\|_\infty^n$$

Lemma 5. (Lemma 5.4 from [4]; or see Lemma 3.7 from [7]) *Let $x = P(x)$ be a monotone system of polynomial equations which has a LFP q^* . For any positive vector $\mathbf{d} \in \mathbb{R}_{>0}^n$ that satisfies $B(q^*)\mathbf{d} \leq \mathbf{d}$, any positive real value $\lambda > 0$, and any nonnegative vector $x \in \mathbb{R}_{\geq 0}^n$, if $q^* - x \leq \lambda \mathbf{d}$, and $(I - B(x))^{-1}$ exists and is nonnegative, then*

$$q^* - \mathcal{N}(x) \leq \frac{\lambda}{2} \mathbf{d}$$

Theorem 4. (Theorem 3.12 of [7]) *For a PPS, $x = P(x)$ in n variables, in SNF form, with LFP q^* , such that $0 < q^* < 1$, for all $i = 1, \dots, n$: $1 - q_i^* \geq 2^{-4|P|}$. In other words, $\|q^*\|_\infty \leq 1 - 2^{-4|P|}$.*

Theorem 5. (Theorem 4.6 of [6])

- (i) *if $x = P(x)$ is a PPS with $q^* < 1$ and $0 \leq y < 1$ then*

$$\|(I - B(\frac{1}{2}(y + q^*)))^{-1}\|_\infty \leq 2^{10|P|} \max\{2(\mathbf{1} - y)_{\min}^{-1}, 2^{|P|}\}$$

- (ii) *if $x = P(x)$ is a strongly connected PPS with $q^* = \mathbf{1}$ and $0 \leq y < 1$, then*

$$\|(I - B(y))^{-1}\|_\infty \leq 2^{4|P|} \frac{1}{(\mathbf{1} - y)_{\min}}$$

⁴ Although the fact that the conditions imply also that $\rho(A) < 1$ is not stated explicitly in Lemma C.3 of [6], it is indeed established in the proof in [6].

Lemma 6. *If $x = P(x)$ is a strongly connected PPS (in SNF form), with Jacobian $B(x)$, and if $B(\frac{1}{2}\mathbf{1})v \leq v$ for some vector $v > 0$, then $\frac{\|v\|_\infty}{v_{\min}} \leq 2^{|P|}$*

Proof. (This proof is a variant of that of Lemma 3.10 in [7].) Let $l = \arg \max_i v_i$, and let $k = \arg \min_j v_j$. Since $x = P(x)$ is in SNF form, every non-zero entry of the matrix $B(\frac{1}{2}\mathbf{1})$ is either $1/2$ or is a coefficient of some monomial in some polynomial $P(x)_i$ of $P(x)$. Moreover, $B(\frac{1}{2}\mathbf{1})$ is irreducible. Calling the entries of $B(\frac{1}{2}\mathbf{1})$, $b_{i,j}$, we have a sequence of *distinct* indices, i_1, i_2, \dots, i_m , with $l = i_1$, $k = i_m$, $m \leq n$, where each $b_{i_j i_{j+1}} > 0$. (Just take the “shortest positive path” from l to k .) For any j :

$$(B(\frac{1}{2}\mathbf{1})v)_{i_{j+1}} \geq b_{i_j i_{j+1}} v_{i_j}$$

By simple induction: $v_k \geq (\prod_{j=1}^{m-1} b_{i_j i_{j+1}}) v_l$. Note that $|P|$ includes the encoding size of each positive coefficient of every polynomial $P(x)_i$. We argued before that each $b_{i_j i_{j+1}}$ is either a coefficient of $x = P(x)$, or is equal to $1/2$. Furthermore, if we consider the equation $x_{i_j} = P(x)_{i_j}$, and denote its encoding size as $|P_{i_j}|$, then it is easy to see $b_{i_j i_{j+1}} \geq 2^{-|P_{i_j}|}$, because either $b_{i_j i_{j+1}}$ appears in $P(x)_{i_j}$, or else $b_{i_j i_{j+1}} = 1/2$, but it is always the case that $|P_{i_j}| \geq 1$. Now, the i_j ’s are distinct (because we are using a shortest path). Therefore, since $|P| = \sum_{i=1}^n |P_i|$, we must have $\prod_{j=1}^{m-1} b_{i_j i_{j+1}} \geq 2^{-|P|}$, and thus we have: $v_k \geq 2^{-|P|} v_l$. \square

Theorem 6. *If $x = P(x)$ is an MPS with n variables, with LFP $q^* \leq 1$, and $\rho(B(q^*)) < 1$, and if we use any rounded-down Newton iteration method defined by $x^{[0]} := 0$, and for all $k \geq 0$, and $x^{[k+1]} := \max(0, \mathcal{N}(x^{[k]}) - e_k)$, where e_k is some error vector such that $0 \leq (e_k)_i \leq 2^{-(h+2)}$ for all $i \in \{1, \dots, n\}$, then for any $0 < \epsilon \leq 1$, $\|q^* - x^{[h+1]}\|_\infty \leq \epsilon$, whenever the chosen parameter h satisfies $h \geq \lceil \log \|(I - B(q^*))^{-1}\|_\infty + \log \frac{1}{\epsilon} \rceil$.*

Proof. We shall use Lemma 5 to prove this. We need to find a vector v , with $B(q^*)v \leq v$ and $v > 0$, called a *cone vector*, such that we can bound the ratio $\frac{v_{\max}}{v_{\min}}$. Here $v_{\max} = \max_i v_i$, and $v_{\min} = \min_i v_i$.

Since we know that $\rho(B(q^*)) < 1$, we have that $(I - B(q^*))$ is nonsingular, and $(I - B(q^*))^{-1} = \sum_{i=0}^{\infty} B(q^*)^i$. We simply take $v := \frac{1}{\|(I - B(q^*))^{-1}\|_\infty} (I - B(q^*))^{-1} \mathbf{1}$ as our cone vector.

Then $B(q^*)v = v - \frac{1}{\|(I - B(q^*))^{-1}\|_\infty} \mathbf{1} \leq v$ and $v = \frac{1}{\|(I - B(q^*))^{-1}\|_\infty} (\mathbf{1} + B(q^*)\mathbf{1} + B(q^*)^2\mathbf{1} + \dots) \geq \frac{1}{\|(I - B(q^*))^{-1}\|_\infty} \mathbf{1}$. The latter not only shows that $v > 0$, but also that $v_{\min} \geq \frac{1}{\|(I - B(q^*))^{-1}\|_\infty}$. Recall that by definition, since $(I - B(q^*))^{-1}$ is non-negative, $\|(I - B(q^*))^{-1}\|_\infty$ is the maximum row sum of any row of $(I - B(q^*))^{-1} = \sum_{i=0}^{\infty} B(q^*)^i$. It follows that $v_{\max} \leq 1$, since $B(q^*)^0 = I$.

Now, $x^{[0]} := 0$, and $q^* \leq 1$, so we know that $q^* - x^{[0]} \leq \mathbf{1} \leq \|(I - B(q^*))^{-1}\|_\infty v \leq 2^h \epsilon v$ (by definition of h). Now, for all $k > 0$, $e_k \leq 2^{-(h+2)} \mathbf{1} \leq \frac{1}{4} \epsilon \frac{1}{\|(I - B(q^*))^{-1}\|_\infty} \mathbf{1} \leq \frac{1}{4} \epsilon v$.

Applying Lemma 5, if $q^* - x^{[k]} \leq \lambda v$, then $q^* - x^{[k+1]} \leq q^* - \mathcal{N}(x^{[k]}) + e_k \leq (\frac{\lambda}{2} + \frac{1}{4}) \epsilon v$. It follows by induction that, for all $k \geq 1$, $q^* - x^{[k]} \leq (2^{h-k} + \frac{1}{2}) \epsilon v$

When $k = h + 1$, this gives $q^* - x^{[h+1]} \leq \epsilon v$. Since $v_{\max} = \|v\|_{\infty} \leq 1$, this means that $\|q^* - x^{[h+1]}\|_{\infty} \leq \epsilon$ as required. \square

Theorem 7. *If the PPS $x = P(x)$ with LFP solution q^* has $\rho(B(q^*)) < 1$ and we use any rounded-down Newton iteration, starting at $x^{[0]} = 0$, defined by $x^{[k+1]} = \max(0, x^{[k]} + (I - B(x^{[k]}))^{-1}(P(x^{[k]}) - x^{[k]}) - e_k)$, for any error vectors e_k where $0 \leq (e_k)_i \leq 2^{-(h+2)}$ for all $i \in \{1, \dots, n\}$, then for any given $0 < \epsilon \leq 1$, $\|q^* - x^{[h+1]}\|_{\infty} \leq \epsilon$, where $h = 14|P| + 3 + \lceil \log(1/\epsilon) \rceil$.*

Theorem 7 follows from Theorem 6 and an upper bound on $\|(I - B(q^*))^{-1}\|_{\infty}$. The following Lemma gives us this, from which Theorem 7 follows immediately:

Lemma 7. *If the PPS $x = P(x)$ with LFP solution q^* has $\rho(B(q^*)) < 1$ then*

$$\|(I - B(q^*))^{-1}\|_{\infty} \leq 2^{14|P|+3}$$

Proof. We split into several cases, based on q^* .

Case 1: $q^ < \mathbf{1}$.* In this case we just need to use Theorem 5 (i), in which we set $y := q^*$, combined with Theorem 4, to conclude that:

$$\|(I - B(q^*))^{-1}\|_{\infty} \leq 2^{14|P|+1}$$

Case 2: $q^ = \mathbf{1}$.* In this case we can instead use the following result from [7]:

Lemma 8. *For a PPS $x = P(x)$, if $(I - B(\mathbf{1}))$ is non-singular then*

$$\|(I - B(\mathbf{1}))^{-1}\|_{\infty} \leq 3^n n 2^{|P|} \leq 2^{3|P|}$$

Proof. The proof of this is basically identical to a proof in [7] for a closely related fact, which was based on more assumptions (but not all of the assumptions were needed).

If we take $(I - B(\mathbf{1}))$ to be the matrix A of Lemma 4, then noting that the product of all the denominators in $(I - B(\mathbf{1}))$ is at most $2^{|P|}$, this yields:

$$\|(I - B(\mathbf{1}))^{-1}\|_{\infty} \leq n 2^{|P|} \|(I - B(\mathbf{1}))\|_{\infty}^n$$

Of course $\|(I - B(\mathbf{1}))\|_{\infty} \leq 1 + \|B(\mathbf{1})\|_{\infty} \leq 3$ (note that here we are using the fact that the system is in SNF normal form). Thus

$$\|(I - B(\mathbf{1}))^{-1}\|_{\infty} \leq 3^n n 2^{|P|}$$

Furthermore, as discussed in [7] (see section A.6, first paragraph), for any PPS $x = P(x)$ we can assume wlog that the equation for every variable requires at least 3 bits, and thus that $|P| \geq 3n \geq n \log 3 + \log n$. Therefore $3^n n 2^{|P|} \leq 2^{3|P|}$. \square

Case 3: Neither $q^* < \mathbf{1}$ nor $q^* = \mathbf{1}$. To finish the proof of Lemma 7, we will combine the above two results for the first two cases to deal with the case when neither $q^* < \mathbf{1}$ nor $q^* = \mathbf{1}$, but that nevertheless $\rho(B(q^*)) < 1$. (It is indeed possible for all three of these conditions to hold, when some coordinates of q^* are 1, and others less than 1.)

Let A (for “always”) denote the set of variables x_i for which $q_i^* = 1$, and let M (for “maybe”) denote the set of variables x_i for which $0 < q_i^* < 1$. We can obviously assume that both A and M are non-empty; otherwise one of the two above theorems gives the result. Furthermore, variables in A obviously cannot depend on those in M (neither directly nor indirectly). Thus we can describe $B(q^*)$ by the following block decomposition

$$B(q^*) = \begin{pmatrix} B(q^*)_M & B(q^*)_{M,A} \\ 0 & B(q^*)_A \end{pmatrix}$$

We need a lemma:

Lemma 9. *For any matrix M satisfying the block decomposition given by $M = \begin{pmatrix} A & B \\ 0 & D \end{pmatrix}$, if both A and D are square and non-singular matrices, then M is also non-singular, and:*

$$\|M^{-1}\|_\infty \leq \max\{\|A^{-1}\|_\infty + \|A^{-1}\|_\infty \|B\|_\infty \|D^{-1}\|_\infty, \|D^{-1}\|_\infty\}$$

Proof. The standard formula for the blockwise inverse of a matrix gives

$$\begin{pmatrix} A & B \\ 0 & D \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} & -A^{-1}BD^{-1} \\ 0 & D^{-1} \end{pmatrix}, \text{ provided that } A \text{ and } D \text{ are non-singular.}$$

(The formula can easily be verified directly by multiplying by $\begin{pmatrix} A & B \\ 0 & D \end{pmatrix}$.)

Now recall that the l_∞ norm for a matrix C is $\|C\|_\infty = \max_i \sum_j |C_{ij}|$, i.e., it is the maximum sum across any row of the absolute value of the entries of the row. So

$$\|M^{-1}\|_\infty \leq \max\{\|A^{-1}\|_\infty + \|A^{-1}\|_\infty \|B\|_\infty \|D^{-1}\|_\infty, \|D^{-1}\|_\infty\}$$

□

Now, $(I - B(q^*)) = \begin{pmatrix} I - B(q^*)_M & -B(q^*)_{M,A} \\ 0 & I - B(q^*)_A \end{pmatrix}$, so $\|(I - B(q^*))^{-1}\|_\infty \leq \max\{\|(I - B(q^*)_M)^{-1}\|_\infty + \|(I - B(q^*)_M)^{-1}\|_\infty \|B(q^*)_{M,A}\|_\infty \|(I - B(q^*)_A)^{-1}\|_\infty, \|(I - B(q^*)_A)^{-1}\|_\infty\}$.

Since we always wlog assume that $x = P(x)$ is a PPS is SNF normal form, $\|B(q^*)\|_\infty \leq 2$. More specifically, $\|B(q^*)_{M,A}\|_\infty \leq 2$. By Case 1, since $\mathbf{0} < q_M^* < \mathbf{1}$, $\|(I - B(q^*)_M)^{-1}\|_\infty \leq 2^{14|P_M|+1}$, where $|P_M|$ denotes the encoding size of the system of equations $x_M = P(x_M, 1_A)_M$, restricted to the variables in M , and with 1 plugged in for all variables in A . Also, by Lemma 8, since $q_A^* = \mathbf{1}$, $\|(I - B(q^*)_A)^{-1}\|_\infty \leq 2^{3|P_A|}$, where $x_A = P(x)_A$ denotes the system of equations

restricted to variables in A (note that these do not depend on variables in M). Thus,

$$\|(I - B(q^*))^{-1}\|_\infty \leq \max\{2^{14|P_M|+1} + 2^{14|P_M|+2+3|P_A|}, 2^{3|P_A|}\}$$

This can be simplified to $\|(I - B(q^*))^{-1}\|_\infty \leq 2^{14|P|+3}$. This completes the proof of Lemma 7. \square

We now have enough to deal with the non-critical case of Theorem 2.

Theorem 2. *For any $\epsilon > 0$, and for an SCFG, G , if the PPS $x = P_G(x)$ has LFP $0 < q^G \leq 1$ and $\rho(B_G(q^G)) < 1$, then if we use R-NM with parameter $h+2$ to approximate the LFP solution of the MPS $y = P_{G \otimes D}(y)$, then $\|q^{G \otimes D} - y^{[h+1]}\|_\infty \leq \epsilon$ where $h := 14|G| + 3 + \lceil \log(1/\epsilon) + \log d \rceil$.*

Thus we can compute the probability $q_A^{G,D} = \sum_{t \in F} q_{s_0 A t}^{G \otimes D}$ within additive error $\delta > 0$ in time polynomial in the input size: $|G|$, $|D|$ and $\log(1/\delta)$, in the standard Turing model of computation.

Proof. Lemma 1 yields that $(I - B_{G \otimes D}(q^{G \otimes D}))^{-1} \in \mathfrak{B}_{\geq 0}^\times$, and that $\mathfrak{C}((I - B_{G \otimes D}(q^{G \otimes D}))^{-1}) = (I - (B_G(q^G))^{-1})$. Lemma 2(vi) relates the norms: $\|(I - B_{G \otimes D}(q^{G \otimes D}))^{-1}\|_\infty \leq d \|(I - (B_G(q^G))^{-1})\|_\infty$. We need a bound on the latter norm. Lemma 7 shows $\|(I - B_G(q^G))^{-1}\|_\infty \leq 2^{14|G|+3}$. So $\|(I - B_{G \otimes D}(q^{G \otimes D}))^{-1}\|_\infty \leq d 2^{14|G|+3}$. Plugging this bound into Theorem 6 yields the result. \square

To deal with critical SCCs, we need a way to analyse how an error in the LFP q^* inside one SCC, \mathcal{S} , where $q_{\mathcal{S}}^* = 1$, affects those SCCs that depend on it:

Theorem 8. *Given a PPS, $y = P(y)$ in SNF form, such that for a subvector x of y , whose equations are $x = P(x, y_{D(x)})$, when restricting $y = P(y)$ to the variables in x , and if we let $y_{D(x)} := z$, for a real-valued vector $0 \leq z < \mathbf{1}$, and if the resulting PPS, $x = P(x, z)$ has LFP $q_z^* > 0$, and if q_1^* is the LFP solution of $x = P(x, \mathbf{1})$ (note that $q_1^* \geq q_z^*$), then:*

(i) *If $q_1^* < \mathbf{1}$ then, $\|q_1^* - q_z^*\|_\infty \leq 2^{14|P|+2} \|\mathbf{1} - z\|_\infty$*

(ii) *If the PPS $x = P(x, \mathbf{1})$ is strongly connected and $q_1^* = \mathbf{1}$ then $\|\mathbf{1} - q_z^*\|_\infty \leq 2^{3|P|} \sqrt{\|\mathbf{1} - z\|_\infty}$*

(iii) *If the PPS, $x = P(x, \mathbf{1})$, is strongly connected and $q_1^* = \mathbf{1}$, and $\rho(B(\mathbf{1}, \mathbf{1})) < 1$ then $\|\mathbf{1} - q_z^*\|_\infty \leq 2^{3|P|} \|\mathbf{1} - z\|_\infty$*

Bad examples given in [4] (see also [20]), show that there are critical PPSs with $q_1^* = 1$, and with $\|\mathbf{1} - q_z^*\|_\infty \geq \sqrt{\|\mathbf{1} - z\|_\infty}$. Thus we cannot hope to get a bound linear in $\|\mathbf{1} - z\|_\infty$ in all cases. Cases (i) and (iii) of Theorem 8 say that we can get a linear bound *except for* critical PPSs, where we indeed need a square root in the strongly connected case (case (ii)).

Proof (of Theorem 8). We first prove the following:

Lemma 10. *For $0 \leq z \leq z' \leq 1$, and for all $0 \leq x \leq 1$, $\|P(x, z') - P(x, z)\|_\infty \leq 2\|z - z'\|_\infty$*

Proof. Consider the k 'th coordinate, $P(x, y)_k$, of the PPS polynomials $P(x, y)$, in SNF form. We distinguish cases based on the type of x_k . If x_k has type Q: then $P(x, z)_k$ and $P(x, z')_k$ both have the form $x_i x_j$, or both have form $z_i^{(\prime)} x_j$, or both the form $x_i z_j^{(\prime)}$, or both the form $z_i^{(\prime)} z_j^{(\prime)}$. Thus, since $0 \leq z \leq z' \leq 1$, and $0 \leq x \leq 1$, we have $0 \leq P(x, z')_k - P(x, z)_k \leq z'_i z'_j - z_i z_j \leq 2\|z - z'\|_\infty$.

In the case where x_k has type L, we have $0 \leq P(x, z')_k - P(x, z)_k \leq \sum_j p_{k,j}(z'_j - z_j) \leq \|z - z'\|_\infty$, because the coefficients $p_{k,j}$ of the type L equation must sum to ≤ 1 .

Finally, if x_k has type T, $P(x, z)_k$ and $P(x, z')_k$ are equal constants, so their difference is 0. \square

Lemma 11. *If $x = P(x, z)$ is a PPS with LFP $q_z^* > 0$ and $x = P(x, z')$ has LFP $q_{z'}^* > 0$ for some $0 \leq z \leq z' \leq 1$, and $(I - B(\frac{1}{2}(q_{z'}^* + q_z^*), z'))$ is non-singular then*

$$\|q_{z'}^* - q_z^*\|_\infty \leq 2\|(I - B(\frac{1}{2}(q_{z'}^* + q_z^*), z'))^{-1}\|_\infty \|z' - z\|_\infty$$

Proof. From Lemma 4.3 of [6], applied to the PPS $x = P(x, z')$, (where we let $y := q_z^*$), we have:

$$(q_{z'}^* - q_z^*) = (I - B(\frac{1}{2}(q_{z'}^* + q_z^*), z'))^{-1}(P(q_z^*, z') - q_z^*)$$

We can take norms:

$$\|q_{z'}^* - q_z^*\|_\infty = \|(I - B(\frac{1}{2}(q_{z'}^* + q_z^*), z'))^{-1}\|_\infty \|P(q_z^*, z') - q_z^*\|_\infty$$

Now we just apply Lemma 10, to obtain that $\|P(q_z^*, z') - q_z^*\|_\infty \leq 2\|z' - z\|_\infty$. \square

To get parts (i) and (ii) of Theorem 8, we apply Theorem 5. For establishing (i) of Theorem 8, we need to apply (i) of Theorem 5 to the PPS, $x = P(x, \mathbf{1})$, with $y := q_z^*$. This gives

$$\|(I - B(\frac{1}{2}(q_z^* + q_1^*), \mathbf{1}))^{-1}\|_\infty \leq 2^{10|P|} \max\{2(\mathbf{1} - q_z^*)_{\min}^{-1}, 2^{|P|}\}$$

Now, since in part (i) of Theorem 8, we are given that $q_1^* < 1$, we know that $q_z^* \leq q_1^* \leq \mathbf{1} - 2^{-4|P|}\mathbf{1}$, by Theorem 3.12 of [7]. So we have

$$\|(I - B(\frac{1}{2}(q_z^* + q_1^*), \mathbf{1}))^{-1}\|_\infty \leq 2^{14|P|+1}$$

Lemma 11 now tells us that:

$$\|q_1^* - q_z^*\|_\infty \leq 2^{14|P|+2} \|\mathbf{1} - z\|_\infty$$

This finishes the proof of part (i) of Theorem 8.

To prove part (ii) of Theorem 8, first remember that we assume $x = P(x, \mathbf{1})$ is strongly connected. We use part (ii) of Theorem 5.

By assumption, $q_1^* = \mathbf{1}$. We take $z = \frac{1}{2}(\mathbf{1} + q_y^*)$, giving:

$$\|(I - B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1}))^{-1}\|_\infty \leq 2^{4|P|} \frac{2}{(\mathbf{1} - q_z^*)_{\min}} \quad (3)$$

Now

$$\begin{aligned} B(\frac{1}{2}\mathbf{1}, \mathbf{1})(\mathbf{1} - q_z^*) &\leq B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1})(\mathbf{1} - q_z^*) \\ &= P(\mathbf{1}, \mathbf{1}) - P(q_z^*, \mathbf{1}) \quad (\text{by Lemma 3.3 of [7]}) \\ &\leq P(\mathbf{1}, \mathbf{1}) - P(q_z^*, z) = \mathbf{1} - q_z^* \end{aligned}$$

Now we apply Lemma 6, letting v be $\mathbf{1} - q_z^*$ in the statement of that Lemma, and considering $B(\frac{1}{2}\mathbf{1}, \mathbf{1})$ in place of the $B(\frac{1}{2}\mathbf{1})$ in the statement of the Lemma. This tells us that $\frac{\|\mathbf{1} - q_z^*\|_\infty}{(\mathbf{1} - q_z^*)_{\min}} \leq 2^{|P|}$.

Now, if we substitute this into the equation (3), we get

$$\|(I - B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1}))^{-1}\|_\infty \leq 2^{5|P|+1} \frac{1}{\|\mathbf{1} - q_z^*\|_\infty}$$

Lemma 11 now gives:

$$\|\mathbf{1} - q_z^*\|_\infty \leq 2\|(I - B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1}))^{-1}\|_\infty \|1 - z\|_\infty$$

Inserting our bound for the norm of $(I - B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1}))^{-1}$ gives:

$$\|\mathbf{1} - q_z^*\|_\infty \leq 2^{5|P|+2} \frac{1}{\|\mathbf{1} - q_z^*\|_\infty} \|1 - z\|_\infty$$

re-arranging and taking the square root gives:

$$\|\mathbf{1} - q_z^*\|_\infty \leq \sqrt{2^{5|P|+2} \|1 - z\|_\infty}$$

As long as the encoding size is $|P| \geq 2$, which we can clearly assume, we have:

$$\|\mathbf{1} - q_z^*\|_\infty \leq 2^{3|P|} \sqrt{\|\mathbf{1} - z\|_\infty}$$

For part (iii), the significance of the condition that $\rho(B(\mathbf{1}, \mathbf{1})) < 1$ is that it implies $(I - B(\mathbf{1}, \mathbf{1}))^{-1}$ exists, and $(I - B(\mathbf{1}, \mathbf{1}))^{-1} \geq (I - B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1}))^{-1}$. So, we use a bound on $\|(I - B(\mathbf{1}, \mathbf{1}))^{-1}\|_\infty$:

Lemma 11 gives:

$$\|\mathbf{1} - q_z^*\|_\infty \leq 2\|(I - B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1}))^{-1}\|_\infty \|1 - z\|_\infty$$

Now $\|(I - B(\frac{1}{2}(\mathbf{1} + q_z^*), \mathbf{1}))^{-1}\|_\infty \leq \|(I - B(\mathbf{1}, \mathbf{1}))^{-1}\|_\infty$. We can apply Lemma 8 on the PPS $x = P(x, \mathbf{1})$, which yields $\|(I - B(\mathbf{1}, \mathbf{1}))^{-1}\|_\infty \leq 2^{3|P|}$. Now we have

$$\|\mathbf{1} - q_z^*\|_\infty \leq 2^{3|P|} \|\mathbf{1} - z\|_\infty$$

as required. \square

Theorem 9. *Suppose $x = P(x)$ is a PPS in SNF form that has critical depth at most \mathfrak{c} . Let $\delta \in \mathbb{R}$, such that $0 \leq \delta \leq 2^{-3|P|-1}$. Suppose that in every bottom-critical SCC of $x = P(x)$ we reduce a single positive coefficient, p , by setting it to $p' = p(1 - \delta)$, resulting in the PPS $x = P_\delta(x)$. Then $\|q^* - q_\delta^*\|_\infty \leq 2^{14|P|+2}\delta^{(1/2^\mathfrak{c})}$ where q^* and q_δ^* are the LFP solutions of $x = P(x)$ and $x = P_\delta(x)$, respectively. Furthermore, $\|(I - B_\delta(q_\delta^*))^{-1}\|_\infty \leq 2^{8|P|+2}\delta^{-3}$.*

Proof. If $\mathfrak{c} = 0$, we have no critical SCCs, so we don't change any coefficients, and $q^* = q_\delta^*$, and the remaining claim about $\|(I - B_\delta(q_\delta^*))^{-1}\|_\infty$ follows directly from Lemma 7.

So, we can assume $\mathfrak{c} > 0$ in the rest of the proof. To establish that q^* and q_δ^* are close, we will use Theorem 8. For any SCC, S , of a PPS $x = P(x)$, either $q_S^* = \mathbf{1}$ or $q_S^* < \mathbf{1}$, because every variable in S depends (directly or indirectly) on every other, so if any of them are < 1 , then so are all the others.

Let S be an SCC with $q_S^* = \mathbf{1}$ and with $(q_\delta^*)_S < \mathbf{1}$. The SCC S necessarily only depends on SCCs, T , with $q_T^* = 1$, because otherwise we wouldn't have $q_S^* = \mathbf{1}$. We want to show that

$$\|\mathbf{1} - (q_\delta^*)_S\|_\infty \leq \delta^{(1/2^{\mathfrak{c}_{S \cup D(S)}})} \cdot 2^{6|P_{S \cup D(S)}|}$$

where $\mathfrak{c}_{S \cup D(S)}$ is the critical depth in $x_{S \cup D(S)} = P_{S \cup D(S)}(x_{S \cup D(S)})$, and $|P_{S \cup D(S)}|$ denotes the encoding size of the latter PPS. To prove this by induction, we can assume

$$\|\mathbf{1} - (q_\delta^*)_{D(S)}\|_\infty \leq \delta^{(1/2^{\mathfrak{c}_{D(S)}})} \cdot 2^{6|P_{D(S)}|} \quad (4)$$

The base case is when S is a bottom-critical SCC, that does not depend on any other critical SCCs. Then even if $D(S)$ is non-empty, $q_{D(S)}^* = (q_\delta^*)_{D(S)}$. However, we do change a single coefficient p in S , by setting it to $p' = p(1 - \delta)$. Note that because the PPS is in SNF form, p must appear in a equation $x_i = P(x_S, \mathbf{1})_i$ where x_i is of type L, and thus the coefficient p appears in a single term px_j . We wish to consider a new PPS in SNF form, parametrized by the possible values $z \in \{(1 - \delta), 1\}$ that we multiply p by. To do this, we can simply add a new variable x_{n+1} (for this particular SCC, S), and we then replace the term px_j by px_{n+1} , and we add a new equation $x_{n+1} = zx_j$ to our system of equations. We denote this new PPS by $(x_S, x_{n+1}) = Q_S((x_S, x_{n+1}), z)$. Note that this is indeed a SNF form PPS for either $z \in \{(1 - \delta), 1\}$. Note also that in terms of encoding size, we have $|Q_S| \leq 2|P_S|$.

The LFP solution of $(x_S, x_{n+1}) = Q_S((x_S, x_{n+1}), 1)$, in the S coordinates has $q_S^* = \mathbf{1}$, and the LFP solution of $(x_S, x_{n+1}) = Q_S((x_S, x_{n+1}), (1 - \delta))$ in the S coordinates is $(q_\delta^*)_S$. Thus, by Theorem 8 (ii), we get $\|\mathbf{1} - (q_\delta^*)_S\|_\infty \leq$

$2^{3|Q_S|}\sqrt{\delta} \leq 2^{6|P_S|}\sqrt{\delta}$. In this case $\mathbf{c}_{S \cup D(S)} = 1$ so this is enough to establish the inductive claim in inequality (4).

Next, suppose that S is a critical SCC that depends on a different critical SCC. q_S^* is the LFP solution of $x_S = P_S(x_S, q_{D(S)}^*)$ and $(q_\delta^*)_S$ is the LFP solution of $x_S = P_S(x_S, (q_\delta^*)_{D(S)})$. By Theorem 8 (ii), $\|\mathbf{1} - (q_\delta^*)_S\|_\infty \leq 2^{3|P_S|}\sqrt{\|\mathbf{1} - (q_\delta^*)_{D(S)}\|_\infty}$. Substituting using the inductive assumption in inequality (4) gives:

$$\begin{aligned} \|\mathbf{1} - (q_\delta^*)_S\|_\infty &\leq 2^{3|P_S|}\sqrt{\|\mathbf{1} - (q_\delta^*)_{D(S)}\|_\infty} \\ &\leq 2^{3|P_S|}\sqrt{\delta^{(1/2^{\mathbf{c}_{D(S)}})}2^{6|P_{D(S)}|}} \\ &= 2^{3|P_S| + \frac{6}{2}|P_{D(S)}|}\delta^{(1/2^{\mathbf{c}_{D(S)}+1})} \\ &\leq \delta^{(1/2^{\mathbf{c}_{S \cup D(S)}})}2^{3|P_{S \cup D(S)}|} \end{aligned}$$

The last inequality holds because $\mathbf{c}_{S \cup D(S)} = \mathbf{c}_{D(S)} + 1$. This is because S is itself a critical SCC. Note also that $|P_{S \cup D(S)}| = |P_S| + |P_{D(S)}|$ since $x_S = P(x_S, x_{D(S)})_S$ and $x_{D(S)} = P(x_{D(S)})_{D(S)}$ are disjoint subsets of the equations in $x = P(x)$.

Finally suppose that S is not a critical SCC but does have $q_S^* = 1$ and depends on some critical SCC. Again q_S^* is the LFP solution of $x_S = P_S(x_S, q_{D(S)}^*)$ and $(q_\delta^*)_S$ is the LFP solution of $x_S = P_S(x_S, (q_\delta^*)_{D(S)})$. By Theorem 8 (iii): $\|\mathbf{1} - (q_\delta^*)_S\|_\infty \leq 2^{3|P_S|}\|\mathbf{1} - (q_\delta^*)_{D(S)}\|_\infty$. Substituting the inductive assumption (4) gives $\|\mathbf{1} - (q_\delta^*)_S\|_\infty \leq 2^{3|P_S|+6|P_{D(S)}|}\delta^{(1/2^{\mathbf{c}_{D(S)}})}$ which simplifies to $\|\mathbf{1} - (q_\delta^*)_S\|_\infty \leq \delta^{(1/2^{\mathbf{c}_{S \cup D(S)}})}2^{6|P_{S \cup D(S)}|}$. This is because S itself is non-critical, so $\mathbf{c}_{D(S)} = \mathbf{c}_{S \cup D(S)}$.

Let A (for “always”) denote the set of variables x_i for which $q_i^* = 1$, and let M (for “maybe”) denote the set of variables x_i for which $0 < q_i^* < 1$. A is non-empty as otherwise we would have no critical SCCs. Every variable x_i in A is part of some SCC S with $q_S^* = 1$. So our induction has already given that

$$\|\mathbf{1} - (q_\delta^*)_A\|_\infty \leq \delta^{1/2^{\mathbf{c}}} 2^{6|P_A|}$$

If M is empty, this bound on $\|q^* - q_\delta^*\|_\infty$ is enough. Otherwise we have to use Theorem 8 (i). This gives that $\|q_M^* - (q_\delta^*)_M\|_\infty \leq 2^{14|P_M|+2}\|\mathbf{1} - (q_\delta^*)_A\|_\infty$. Substituting gives $\|q_M^* - (q_\delta^*)_M\|_\infty \leq 2^{14|P|+2}\delta^{1/2^{\mathbf{c}}}$. We have now shown that

$$\|q^* - q_\delta^*\|_\infty \leq 2^{14|P|+2}\delta^{1/2^{\mathbf{c}}}$$

The only thing left to complete the proof of Theorem 9 is to get a bound on $\|(I - B_\delta(q_\delta^*))^{-1}\|_\infty$. For this we will use the techniques of the proof of Theorem 7. Call the set of variables for which $(q_\delta^*)_i = 1$, A_δ and the set of variables x_i for which $0 < (q_\delta^*)_i < 1$, M_δ . Since $q_\delta^* \leq q^*$, $M \subseteq M_\delta$ and $A_\delta \subseteq A$. It is worth noting that variables belonging to critical SCCs are in $A \cap M_\delta$. We will first show that if a variable x_i depends (directly or indirectly) on some variable x_j for which we have reduced a coefficient in $P_\delta(x)_j$, then $(q_\delta^*)_i \leq 1 - 2^{-|P|}\delta$. For any such x_i , consider a *shortest* sequence $x_{l_1}, x_{l_2}, \dots, x_{l_m}$, such that (1):

$l_1 = j$ and $P_\delta(x)_j$ has a reduced coefficient in it, (2): $l_m = i$, and (3): for every $0 \leq k < m$, $P_\delta(x)_{l_{k+1}}$ contains a term with x_{l_k} . There is some term $p_{j,h}x_h$ in $P(x)_j$ which has been changed to $p_{j,h}(1 - \delta)x_h$ in $P_\delta(x)_j$. Since $x = P(x)$ is a PPS, $P(\mathbf{1})_j \leq 1$, but note that $P_\delta(x)_j$ is not proper, as indeed we must have that $P_\delta(\mathbf{1})_j \leq P(\mathbf{1})_j - p_{j,h}\delta \leq 1 - p_{j,h}\delta$. Also note that $(q_\delta^*)_j = P_\delta(q_\delta^*)_j \leq P_\delta(\mathbf{1})_j \leq 1 - p_{j,h}\delta$. For any $0 \leq k < m$, if $x_{l_{k+1}}$ has type **Q**, then $(q_\delta^*)_{l_{k+1}} \leq (q_\delta^*)_{l_k}$. If $x_{l_{k+1}}$ has type **L**, then $1 - (q_\delta^*)_{l_{k+1}} \geq p_{l_{k+1},l_k}(1 - (q_\delta^*)_{l_k})$. By an easy induction $1 - (q_\delta^*)_i \geq (\prod_{\{k | x_{l_k} \text{ has Type L}\}} p_{l_{k+1},l_k})(1 - (q_\delta^*)_j)$. Thus:

$$1 - (q_\delta^*)_i \geq \left(\prod_{\{k | x_{l_k} \text{ has Type L}\}} p_{l_{k+1},l_k} \right) p_{j,h} \delta$$

Since this is the shortest sequence satisfying the stated conditions, for any $0 \leq k < m$, $P_\delta(x)_{l_k}$ has not had any coefficients reduced, and furthermore the x_{l_k} 's are all distinct variables. So all these coefficients p_{l_{k+1},l_k} and $p_{j,h}$ are distinct coefficients in $x = P(x)$. The encoding size $|P|$ is at least the number of bits describing these rationals p_{l_{k+1},l_k} and $p_{j,h}$ and thus

$$(q_\delta^*)_i \leq 1 - 2^{-|P|}\delta$$

Next we show that the PPS $x = P_\delta(x)$ is non-critical. Suppose, for a contradiction that $x = P_\delta(x)$ is critical. Then it has some critical SCC S . But then S must have also been an SCC in the PPS $x = P(x)$, because the dependency graphs of these PPSs are the same (we never reduce a positive probability to 0). For S to be a critical SCC in $x = P_\delta(x)$, we must have that $(q_\delta^*)_S = \mathbf{1}$ and $\rho(B_\delta(\mathbf{1})_S) = 1$. However, $q^* \geq q_\delta^*$ and $\rho(B(\mathbf{1})_S) \geq \rho(B_\delta(\mathbf{1})_S) = 1$. So $q_S^* = \mathbf{1}$. Lemma 6.5 of [8] shows that for any strongly connected PPS, $x = P(x)$, with Jacobian $B(x)$, and with LFP, q^* , if $x < q^*$, then $\rho(B(x)) < 1$. Thus, by continuity of eigenvalues, $\rho(B(q^*)) \leq 1$. Applying this to the strongly connected PPS $x_S = P(x_S, \mathbf{1})_S$, since $q_S^* = \mathbf{1}$, we get $\rho(B(\mathbf{1})_S) \leq 1$. Thus $\rho(B(\mathbf{1})_S) = 1$ i.e. S is a critical SCC of $x = P(x)$. Either S is a bottom-critical-SCC or it depends on some bottom-critical-SCC. So every variable x_i in S depends on some variable x_j for which we have reduced a coefficient in $P_\delta(x)_j$. So for every x_i in S , $q_i^* \leq 1 - 2^{-|P|}\delta$. But this contradicts our earlier assertion that $q_S^* = \mathbf{1}$.

$$B_\delta(q_\delta^*) \text{ has the block decomposition } B_\delta(q_\delta^*) = \begin{pmatrix} B_\delta(q_\delta^*)_{M_\delta} & B_\delta(q_\delta^*)_{M_\delta, A_\delta} \\ 0 & B_\delta(q_\delta^*)_{A_\delta} \end{pmatrix}.$$

It is possible that A_δ is empty, in which case the bound we will obtain on $\|(I - B_\delta(q_\delta^*)_{M_\delta})^{-1}\|_\infty$ will be enough to show the theorem. So we suppose here that A_δ is non-empty. M_δ is non-empty since we assumed that we have at least one critical SCC.

We need to show that both $I - B_\delta(q_\delta^*)_{M_\delta}$ and $I - B_\delta(q_\delta^*)_{A_\delta}$ are nonsingular, and we need to get upper bounds on $\|(I - B_\delta(q_\delta^*)_{M_\delta})^{-1}\|_\infty$ and $\|(I - B_\delta(q_\delta^*)_{A_\delta})^{-1}\|_\infty$. Once we do so, we can then apply Lemma 9 to get a bound on $\|(I - B(q_\delta^*))^{-1}\|_\infty$.

First, let us show that $I - B_\delta(q_\delta^*)_{A_\delta}$ is non-singular, and also bound $\|(I - B_\delta(q_\delta^*)_{A_\delta})^{-1}\|_\infty$.

We note that $P(x)_{A_\delta} = P_\delta(x)_{A_\delta}$. We have shown that any variable x_i for which we have reduced a coefficient in $P_\delta(x)_i$ has $q_i^* \leq 1 - 2^{-|P|}\delta$ and so x_i is not in A_δ . Thus the equations in $x_{A_\delta} = P_\delta(x_{A_\delta})_{A_\delta}$ are a subset of the equations $x = P(x)$ and so the encoding size of this PPS is at most $|P|$. We have also shown that the PPS $x = P_\delta(x)$ is non-critical. So we can apply Lemma 8 to the PPS $x_{A_\delta} = P_\delta(x_{A_\delta})_{A_\delta}$, which gives $\|(I - B_\delta(q_\delta^*)_{A_\delta})^{-1}\|_\infty \leq 2^{3|P|}$.

Now, let us show that $I - B_\delta(q_\delta^*)_{M_\delta}$ is non-singular, and also bound $\|(I - B_\delta(q_\delta^*)_{M_\delta})^{-1}\|_\infty$.

Consider the PPS, restricted to the variables in M_δ . Note that no variable in A_δ can depend on these. Thus, restricting the PPS $x = P_\delta(x)$ to the variables in M_δ defines a PPS $x_{M_\delta} = P_\delta(x_{M_\delta}, \mathbf{1})_{M_\delta}$. Note that the LFP of this is $(q_\delta^*)_{M_\delta} < 1$, by definition of M_δ . To simplify notation in the current argument, we shall denote this PPS by $y = R(y)$, and we shall use $r^* := (q_\delta^*)_{M_\delta}$ to denote its LFP. Furthermore, let us use $B_R(y)$ to denote its Jacobian. We note, firstly, that $B_R(r^*) = B_\delta(q_\delta^*)_{M_\delta}$. The way to see this is to note that $q_\delta^* = (r^*, \mathbf{1})$ and so the entries of both matrices are $\frac{\partial(P_\delta)_i}{\partial x_j}(q_\delta^*)$ for $x_i, x_j \in M_\delta$.

So, rephrased, we want to show $\rho(B_R(r^*)) < 1$, and we want to find a bound on $(I - B_R(r^*))^{-1}$. To do this, we need to follow the proof of Theorem 5 (i) in the case $y = r^*$. (That Theorem was proved in [6].)

We need to use Lemma 3, with $A = B_R(r^*)$ and $u = \mathbf{1} - r^*$. By Lemma 3.5 of [7], $B_R(r^*)(\mathbf{1} - r^*) \leq \mathbf{1} - r^*$. We want to find any β so that condition (I) of Lemma 3 applies to variables y_i such that either y_i has type \mathbb{Q} or else $R(1)_i < 1$. Namely for such variables y_i , it should be the case that $(B_R(r^*)(\mathbf{1} - r^*))_i \leq (1 - \beta)(\mathbf{1} - r^*)_i$.

Let us first note that, for any y_i , $r_i^* \leq 1 - 2^{-|P|}\delta$. We have shown that if a variable x_i depends on some variable x_j for which we have reduced a coefficient in $P_\delta(x)_j$, then $(q_\delta^*)_i \leq 1 - 2^{-|P|}\delta$. If $x_i \in M_\delta$ depends on no such variables, then $x_i \in M$. But then we have $q_i^* \leq 1 - 2^{-4|P|} \leq 1 - 2^{-|P|}\delta$ because we assumed that $\delta \leq 2^{-3|P|}$. So for any $x_i \in M_\delta$, $(q_\delta^*)_i \leq 1 - 2^{-|P|}\delta$.

In the case where $y_i = R(y)_i$ has form **Q**, for some y_j, y_k , $R(y)_i = y_j y_k$ and so

$$\begin{aligned}
B_r(r^*)(\mathbf{1} - r^*)_i &= r_j^*(1 - r_k^*) + r_k^*(1 - r_j^*) \\
&= r_j^* + r_k^* - 2r_j^* r_k^* \\
&= (1 - r_j^* r_k^*) - (1 + r_j^* r_k^* - r_j^* - r_k^*) \\
&= (1 - r_i^*) - (1 - r_k^*)(1 - r_j^*) \\
&= (1 - r_i^*) - \frac{1}{2}((1 - r_k^*)(1 - r_j^*) + (1 - r_j^*)(1 - r_k^*)) \\
&\leq (1 - r_i^*) - \frac{1}{2}2^{-|P|}\delta((1 - r_j^*) + (1 - r_k^*)) \\
&\leq (1 - r_i^*) - \frac{1}{2}2^{-|P|}\delta((1 - r_j^*) + (1 - r_k^*) - (1 - r_j^*)(1 - r_k^*)) \\
&= (1 - r_i^*) - \frac{1}{2}2^{-|P|}\delta(1 - r_i^*) \\
&= (1 - \frac{1}{2}2^{-|P|}\delta)(1 - r_i^*)
\end{aligned}$$

Some variables x_i with $P_\delta(\mathbf{1})_i < 1$ have $P(\mathbf{1})_i < 1$, in which case $P(\mathbf{1})_i \leq 1 - 2^{|P|}$. If a variable x_i has $P_\delta(\mathbf{1})_i < 1$ but $P(\mathbf{1})_i = 1$ then we have reduced some coefficient in $P_\delta(x)_i$ by multiplying it by $1 - \delta$ so we have $P_\delta(\mathbf{1})_i \leq 2^{-|P|}\delta$. So for any y_i with $R(\mathbf{1})_i < 1$, $R(\mathbf{1})_i \leq 2^{-|P|}\delta$. So if $R(\mathbf{1})_i < 1$,

$$\begin{aligned}
(B_R(r^*)(\mathbf{1} - r^*))_i &\leq (B_R(\frac{1}{2}(\mathbf{1} + r^*))(\mathbf{1} - r^*))_i \\
&\leq (R(\mathbf{1}))_i - (R(r^*))_i \\
&\leq (1 - 2^{-|P|}\delta) - (r^*)_i \\
&\leq (1 - 2^{-|P|}\delta)(1 - q_\delta^*)_i
\end{aligned}$$

So condition (I) of Lemma 3, with $\beta = 2^{-(|P|+1)}\delta$, applies to variables y_i which either have type **Q** or have $R_i(\mathbf{1}) < 1$.

It remains to find an α such that condition (II) of Lemma 3 that applies to y_i which either has type **L** and satisfies $R(\mathbf{1})_i = 1$. (Note that there aren't any variables of type **T** in M_δ , and thus none in y .) We need the following Lemma from [6]:

Lemma 12. (Lemma C.8 of [6]) For any PPS, $x=P(x)$, with LFP $0 < q^* < 1$, for any variable x_i either

- (I) the equation $x_i = P(x)_i$ is of type **Q**, or else $P(1)_i < 1$.
- (II) x_i depends on a variable x_j , such that $x_j = P(x)_i$ is of type **Q**, or else $P(1)_j < 1$.

So given y_i of type **L** and with $R_i(\mathbf{1}) = 1$, there is a sequence $y_{l_1}, y_{l_2}, \dots, y_{l_m}$ with $l_m = i$, with y_{l_1} of type **Q** or $R(\mathbf{1})_{l_m} < 1$ and for every $0 \leq k < m$, $R(y)_{l_{k+1}}$ contains a term with y_{l_k} . Without loss of generality, we consider the shortest such sequence. Then for $0 < k \leq m$, y_{l_k} does not have type **Q** so it must have

type L. Also $R(\mathbf{1})_{l_k} = 1$. So $R(y)_{l_k}$ contains a term $p_{l_k, l_{k-1}} y_{k-1}$. We have that, $B_R(r^*)_{l_k, l_{k-1}} = p_{l_k, l_{k-1}}$. Because $R(\mathbf{1})_{l_k} = 1$, this term has not been reduced in P_δ , so $p_{l_k, l_{k-1}}$ is a coefficient in $x = P(x)$. That this is the shortest sequence implies that each of these is a distinct coefficient in $x = P(x)$. So $\prod_{k=1}^{m-1} p_{l_{k+1}, l_k} \geq 2^{-|P|}$. Now $(B_R(r^*)^{m-1})_{i, l_m} \geq \prod_{k=1}^{m-1} B_R(r^*)_{l_{k+1}, l_k} = \prod_{k=1}^{m-1} p_{l_{k+1}, l_k} \geq 2^{-|P|}$.

So condition (II) of Lemma 3 applies to y_i of type L with $R_i(\mathbf{1}) = 1$ when $\alpha = 2^{-|P|}$.

We can now use Lemma 3 with $A = B_R(r^*)$, $u = \mathbf{1} - r^*$, $\alpha = 2^{-|P|}$ and $\beta = 2^{-|P|}\delta$, giving

$$\|(I - B_R(r^*))^{-1}\|_\infty \leq \frac{n}{(\mathbf{1} - r^*)_{\min}^2 2^{-|P|} 2^{-|P|} \delta}$$

We have argued that $(\mathbf{1} - r^*)_{\min} \geq 2^{-|P|}\delta$. Using $n \leq 2^{|P|}$ as a (very) conservative bound on n , we have:

$$\|(I - B_\delta(q_\delta^*)_{M_\delta})^{-1}\|_\infty \leq 2^{5|P|}\delta^{-3} \quad (5)$$

If A_δ is empty, then $B_\delta(q_\delta^*) = B_\delta(q_\delta^*)_{M_\delta}$ and so we are done.

Otherwise we appeal to Lemma 9 with the block decomposition $I - B_\delta(q_\delta^*) = \begin{pmatrix} I - B_\delta(q_\delta^*)_{M_\delta} & -B_\delta(q_\delta^*)_{M_\delta, A_\delta} \\ 0 & I - B_\delta(q_\delta^*)_{A_\delta} \end{pmatrix}$. Letting $\mathcal{Z} = (I - B_\delta(q_\delta^*)_{M_\delta})$, applying Lemma 9, we get:

$$\|(I - B_\delta(q_\delta^*))^{-1}\|_\infty \leq \max\{\|\mathcal{Z}^{-1}\|_\infty + \|\mathcal{Z}^{-1}\|_\infty \|B_\delta(q_\delta^*)_{M_\delta, A_\delta}\|_\infty \|(I - B_\delta(q_\delta^*)_{A_\delta})^{-1}\|_\infty, \|(I - B_\delta(q_\delta^*)_{A_\delta})^{-1}\|_\infty\}$$

and $\|(I - B_\delta(q_\delta^*)_{A_\delta})^{-1}\|_\infty \leq 2^{3|P|}$ and $\|B_\delta(q_\delta^*)_{M_\delta, A_\delta}\|_\infty \leq 2$. Combining with the bound above in (5), we get:

$$\|(I - B_\delta(q_\delta^*))^{-1}\|_\infty \leq \max\{2^{5|P|}\delta^{-3} + 2^{5|P|}\delta^{-3} 2^{3|P|} 2, 2^{3|P|}\}$$

Or, more simply, $\|(I - B_\delta(q_\delta^*))^{-1}\|_\infty \leq 2^{8|P|+2}\delta^{-3}$. \square

We are finally ready to prove Theorem 3, to which this entire section was dedicated.

Theorem 3. *For any $\epsilon > 0$, and for any SCFG, G , in SNF form, with $q^G > 0$, with critical depth $\mathfrak{c}(G)$, consider the new SCFG, G' , obtained from G by the following process: for each bottom-critical SCC, \mathcal{S} , of $x = P_G(x)$, find any rule $r = A \xrightarrow{p} B$ of G , such that A and B are both in \mathcal{S} (since G is in SNF, such a rule must exist in every critical SCC). Reduce the probability p , by setting it to $p' = p(1 - 2^{-(14|G|+3)2^{\mathfrak{c}(G)}} \epsilon^{2^{\mathfrak{c}(G)}})$. Do this for all bottom-critical SCCs. This defines G' , which is non-critical.*

Using G' instead of G , if we apply R-NM, with parameter $h+2$ to approximate the LFP solution $q^{G' \otimes D}$ of the MPS $y = P_{G' \otimes D}(y)$, then $\|q^{G \otimes D} - x^{[h+1]}\|_\infty \leq \epsilon$ where $h := \lceil \log d + (3 \cdot 2^{\mathfrak{c}(G)} + 1)(\log(1/\epsilon) + 14|G| + 3) \rceil$.

Thus we can compute the probability $q_A^{G,D} = \sum_{t \in F} q_{s_0 A t}^{G \otimes D}$ within additive error $\delta > 0$ in time polynomial in: $|G|$, $|D|$, $\log(1/\delta)$, and $2^{c(G)}$, in the standard Turing model of computation.

Proof (of Theorem 3).

Note that for an SCFG, G , and its corresponding PPS, $x = P_G(x)$, the bit encoding size of G is at least as big as that of the PPS. In other words, we have $|G| \geq |P_G|$. So, we can apply Theorem 9 to the PPS $x = P_G(x)$ with $\delta := 2^{-(14|G|+3)2^{c(G)}} \epsilon^{2^{c(G)}}$, yielding that $\|q^G - q^{G'}\|_\infty \leq \frac{\epsilon}{2}$ and $\|(I - B_{G'}(q^{G'}))^{-1}\|_\infty \leq 2^{8|G|+2+3(14|G|+3)2^{c(G)}} \epsilon^{-3 \cdot 2^{c(G)}}$. Now Lemma 1 and Lemma 2 (vi) allow us to convert this bound on $\|(I - B_{G'}(q^{G'}))^{-1}\|_\infty$ to a bound on $\|(I - B_{G' \otimes D}(q^{G' \otimes D}))^{-1}\|_\infty$. Namely:

$$\|(I - B_{G' \otimes D}(q^{G' \otimes D}))^{-1}\|_\infty \leq d 2^{8|G|+2+3(14|G|+3)2^{c(G)}} \epsilon^{-3 \cdot 2^{c(G)}}$$

Now Theorem 6 gives that $\|q_{G' \otimes D}^* - x^{[h+1]}\|_\infty \leq \frac{\epsilon}{2}$ since $h \geq \log \|(I - B_{G' \otimes D}(q_{G' \otimes D}^*))^{-1}\|_\infty + \log(1/\frac{\epsilon}{2})$. Thus

$$\begin{aligned} \|q^{G \otimes D} - x^{[h+1]}\|_\infty &\leq \|q^{G \otimes D} - q^{G' \otimes D}\|_\infty + \|q^{G' \otimes D} - x^{[h+1]}\|_\infty \\ &\leq \|q^G - q^{G'}\|_\infty + \|q^{G' \otimes D} - x^{[h+1]}\|_\infty \quad (\text{by Lemma 1 \& Lemma 2(vi)}) \\ &\leq \frac{\epsilon}{2} + \frac{\epsilon}{2} \\ &= \epsilon \end{aligned}$$

□

C Proof of Proposition 4

Recall that, for a string $\alpha \in (V \cup \Sigma)^*$, with $n = |V|$, $\kappa(\alpha)$ is the n -vector where, for $A \in V$, $\kappa_A(\alpha)$ is the number of times A appears in α . Recall that we define $C(r, \pi)$ to be the number of times the rule r is used in the derivation π , and we define $C(A, \pi) = \sum_{r \in R_A} C(r, \pi)$. For $A \in V$, define \mathbf{e}^A to be the unit n -vector with $(\mathbf{e}^A)_A = 1$ and $(\mathbf{e}^A)_B = 0$ for $B \neq A$. Define $K(\pi) = \sum_A C(A, \pi) \mathbf{e}^A$.

Recall that when doing parameter estimation (and EM) we use formula (2)

$$p(A \rightarrow \gamma) := \frac{\sum_\pi \mathcal{P}(\pi) C(A \rightarrow \gamma, \pi)}{\sum_\pi \mathcal{P}(\pi) C(A, \pi)}$$

to obtain (or update) the probabilities of rules in G .

Recall that $\mathcal{P}(\pi)$ is a probability distribution on the complete derivations of the grammar that start at a designated start nonterminal, S . Again, equation (2) only makes sense when the sums $\sum_\pi \mathcal{P}(\pi) C(A, \pi)$ are finite and nonzero, which we assume; we also assume every non-terminal and rule of \mathcal{H} appears in some complete derivation π with $\mathcal{P}(\pi) > 0$.

Proposition 4. *If we use parameter estimation to obtain SCFG G using equation (2), under the stated assumptions, then G is consistent, i.e. $q^G = \mathbf{1}$, and furthermore the PPS $x = P_G(x)$ is non-critical, i.e., $\rho(B_G(\mathbf{1})) < 1$.*

A first step toward establishing Proposition 4 is the following Lemma, from which we derive a (left) cone vector for $B_G(\mathbf{1})$, which ultimately allows us to show $\rho(B_G(\mathbf{1})) < 1$.

Lemma 13. *Let S denote the designated start nonterminal. Then*

$$\mathbf{e}^S = (I - B_G(\mathbf{1})^T) \left(\sum_{\pi} \mathcal{P}(\pi) K(\pi) \right)$$

Proof. Firstly, we need to relate $B_G(\mathbf{1})$ to the probabilities of the rules. Given a rule $A \rightarrow \gamma$ we define $B_{A \rightarrow \gamma}(x) := B_{G_{A \rightarrow \gamma}}(x)$ where $G_{A \rightarrow \gamma}$ is an SCFG with the same non-terminals and terminals as G but with only one rule, $A \xrightarrow{1} \gamma$, which has probability 1. So then $B_{A \rightarrow \gamma}(\mathbf{1})$ is zero outside the A row. We allow that G may or may not be in normal form. We can say that

$$P_G(x)_A = \sum_{r=(A \rightarrow \gamma) \in R_A} p(r) \prod_{B \in V} x_B^{\kappa_B(\gamma)}$$

In terms of the “partial” SCFGs, G_r , associated with each rule $r \in R$, this says $P_G(x)_A = \sum_{r \in R_A} p(r) P_{G_r}(x)_A$. The A row of $B_G(x)$ is then $\sum_{r \in R_A} p(r) B_r(x)_A$. Since $B_{A \rightarrow \gamma}(x_G)$ is zero outside of the A row, $B_G(x) = \sum_A \sum_{r \in R_A} p(r) B_r(x)$. That is:

$$B_G(x) = \sum_{r \in R} p(r) B_r(x) \quad (6)$$

So we can obtain $B_G(\mathbf{1})$ from each of the $B_r(\mathbf{1})$. $B_{A \rightarrow \gamma}(\mathbf{1})$ is zero except in the A row. For any non-terminal B ,

$B_{A \rightarrow \gamma}(x)_{A,B} = \frac{\partial}{\partial x_B} \prod_C x_C^{\kappa_C(\gamma)} = \kappa_B(\gamma) x_B^{\kappa_B(\gamma)-1} \prod_{C \neq B} x_C^{\kappa_C(\gamma)}$. Evaluated at $\mathbf{1}$, this yields:

$$(B_{A \rightarrow \gamma}(\mathbf{1}))_{A,B} = \kappa_B(\gamma) \quad (7)$$

Now we look at what happens to the count of non-terminals in the derivation π . We have $S \xRightarrow{\pi} w$ for some $w \in \Sigma^*$. That is, $\pi = r_1 r_2 \dots r_k \in R^*$, and $\alpha_0 \xRightarrow{r_1} \alpha_1 \xRightarrow{r_2} \alpha_2 \xRightarrow{r_3} \dots \xRightarrow{r_k} \alpha_m$, for $\alpha_0 = S$, $\alpha_m = w$ and some $\alpha_1, \alpha_2, \dots, \alpha_{m-1} \in (V \cup \Sigma)^*$.

Consider $\alpha_i \xRightarrow{r_i} \alpha_{i+1}$ for some $0 \leq i \leq m-1$. The rule r_i is $A_i \rightarrow \gamma_i$ for some non-terminal A_i and some string γ_i . Replacing A_i by γ_i affects the counts of the non-terminals by $\kappa(\alpha_{i+1}) - \kappa(\alpha_i) = \kappa(\gamma_i) - \mathbf{e}^{A_i}$. Note that for any nonterminal A , and rule $A \rightarrow \gamma$, we have $B_{A \rightarrow \gamma}(\mathbf{1})^T \mathbf{e}^A = \kappa(\gamma)$, by equation (7), so

$$(I - B_{A \rightarrow \gamma}(\mathbf{1})^T) \mathbf{e}^A = \mathbf{e}^A - \kappa(\gamma) \quad (8)$$

Since for any string $w \in \Sigma^*$, we have $\kappa(w) = \mathbf{0}$, we get:

$$\begin{aligned}
\mathbf{e}^S &= \mathbf{e}^S - \kappa(w) \\
&= \sum_{i=0}^{m-1} \kappa(\alpha_i) - \kappa(\alpha_{i+1}) \\
&= \sum_A \sum_{(A \rightarrow \gamma) \in R_A} (C(A \rightarrow \gamma, \pi))(\mathbf{e}^A - \kappa(\gamma)) \\
&= \sum_A \sum_{(A \rightarrow \gamma) \in R_A} (C(A \rightarrow \gamma, \pi))(I - B_{A \rightarrow \gamma}(\mathbf{1})^T) \mathbf{e}^A \quad (\text{by (8)})
\end{aligned}$$

This is true for any complete derivation π , so we can use the probability distribution $\mathcal{P}(\pi)$, which has $\sum_{\pi} \mathcal{P}(\pi) = 1$ to obtain:

$$\begin{aligned}
\mathbf{e}^S &= \sum_{\pi} \mathcal{P}(\pi) \sum_A \sum_{(A \rightarrow \gamma) \in R_A} (C(A \rightarrow \gamma, \pi))(I - B_{A \rightarrow \gamma}(\mathbf{1})^T) \mathbf{e}^A \\
&= \sum_{A \in V} \left(\sum_{(A \rightarrow \gamma) \in R_A} \sum_{\pi} \mathcal{P}(\pi) (C(A \rightarrow \gamma, \pi))(I - B_{A \rightarrow \gamma}(\mathbf{1})^T) \right) \mathbf{e}^A \\
&= \sum_A (I - B_G(\mathbf{1})^T) \left(\sum_{\pi} \mathcal{P}(\pi) C(A, \pi) \right) \mathbf{e}^A \\
&= (I - B_G(\mathbf{1})^T) \left(\sum_{\pi} \mathcal{P}(\pi) K(\pi) \right)
\end{aligned}$$

□

Proof (Proof of Theorem 4). Define $v = (\sum_{\pi} \mathcal{P}(\pi) K(\pi))$. Then we have that $v = B_G(\mathbf{1})^T v + \mathbf{e}^S$. We want to use Lemma 3 to show that $\rho(B_G(\mathbf{1})^T) < 1$. We can do this by applying it to the vector $u = \frac{1}{\|v\|_{\infty}} v$. We do not need explicit bounds on α , β and u_{\min} , but we need to show that the conditions hold for some positive α , β and u_{\min} . Firstly, we note that $v > 0$, since every non-terminal in G appears in some derivation π with $\mathcal{P}(\pi) > 0$. So $u > 0$. Since $u = \frac{1}{\|v\|_{\infty}} v$, $\|u\|_{\infty} = 1$. Note that $u = \frac{1}{\|v\|_{\infty}} (B_G(\mathbf{1})^T v + \mathbf{e}^S) = B_G(\mathbf{1})^T u + \frac{1}{\|v\|_{\infty}} \mathbf{e}^S$. Thus $B_G(\mathbf{1})^T u = u - \frac{1}{\|v\|_{\infty}} \mathbf{e}^S \leq u$. In the S coordinate (and only in the S coordinate), we have that $(B_G(\mathbf{1})^T u)_S = u_S - \frac{1}{\|v\|_{\infty}} < u_S$, so there is some $\beta > 0$ for which $(B_G(\mathbf{1})^T u)_S \leq (1 - \beta)u_S$. For this β , u_S satisfies condition (I) of Lemma 3. We need to find an α for which all non-terminals other than S satisfy condition (II) of Lemma 3.

Consider a non-terminal $A \neq S$. A appears in some complete derivation π with $\mathcal{P}(\pi) > 0$. There is some sequence of (not necessarily consecutive) rules $r_i : D_i \rightarrow \gamma_i$, $i = 1, \dots, k$, appearing in that order in π , such that $D_1 = S$, $D_i \in \gamma_{i-1}$ for all $2 \leq i \leq k$, and $A \in \gamma_k$. Without loss of generality $k \leq n$, since otherwise there must be i, j with $2 \leq i < j \leq k$ such that $D_i = D_j$ and so the shorter sequence $r_1, \dots, r_{i-1}, r_j, \dots, r_k$ would have satisfied the above conditions. For any $1 \leq i \leq k-1$, $(B_{r_i}(\mathbf{1}))_{D_i, D_{i+1}} = \kappa(\gamma_i)_{D_{i+1}} \geq 1$, and similarly $(B_{r_k}(\mathbf{1}))_{D_k, A} \geq 1$. Now any r_j , with $1 \leq j \leq k$, appears in π which has $\mathcal{P}(\pi) > 0$. So $p(r_j) > 0$.

But $B_G(\mathbf{1}) \geq p(r_j)B_{r_j}(\mathbf{1})$. So for any $1 \leq i \leq k-1$, $(B_G(\mathbf{1}))_{D_i, D_{i+1}} \geq p(r_i) > 0$ and similarly $B_G(\mathbf{1})_{D_k, A} > 0$. So $(B_G(\mathbf{1})^k)_{S, A} > 0$. Then $((B_G(\mathbf{1})^T)^k)_{A, S} = ((B_G(\mathbf{1})^k)^T)_{A, S} = (B_G(\mathbf{1})^k)_{S, A} > 0$. We then define $\alpha_A = ((B_G(\mathbf{1})^T)^k)_{A, S}$. If we take $\alpha = \min_{\{A \in V \mid A \neq S\}} \alpha_A$, then $\alpha > 0$ and all non-terminals $A \neq S$ satisfy condition (II) of Lemma 3: i.e., for each $A \neq S$, there is a k with $((B_G(\mathbf{1})^T)^k)_{A, S} \geq \alpha$. We can now apply Lemma 3 which yields that $\rho(B_G(\mathbf{1})^T) < 1$. So $\rho(B_G(\mathbf{1})) = \rho(B_G(\mathbf{1})^T) < 1$. So, G is not critical. Consistency of G , i.e., the fact that $q^G = \mathbf{1}$, also follows. This holds because, firstly, we can easily see that G is a *proper* SCFG. In other words, for any nonterminal A , the sum of the rule probabilities is 1, because $\sum_{r \in R_A} p(r) = \sum_{r \in R_A} \frac{\sum_{\pi} \mathcal{P}(\pi) C(r, \pi)}{\sum_{\pi} \mathcal{P}(\pi) C(A, \pi)} = 1$.

Thus, G has a PPS, $x = P_G(x)$, such that $P_G(\mathbf{1}) = \mathbf{1}$, and $\rho(B_G(\mathbf{1})) < 1$. Lemma 6.3 of [8] tells us that for any vectors $0 \leq x \leq y$, $B_G(y)(y - x) \geq P_G(y) - P_G(x)$. Let $y = \mathbf{1}$, and let $x = q^G$. Then we have $B_G(\mathbf{1})(\mathbf{1} - q^G) \geq P_G(\mathbf{1}) - P_G(q^G) = \mathbf{1} - q^G$, since we have argued both $\mathbf{1}$ and q^G are fixed points of P_G . But $B_G(\mathbf{1})$ is a non-negative square matrix, and $(\mathbf{1} - q^G) \geq 0$. Theorem 8.3.2 of [10] tells us that for a square matrix $M \geq 0$, and vector $v \geq 0$, if $v \neq 0$ and $Mv \geq v$, then $\rho(M) \geq 1$. We know that $B_G(\mathbf{1})(\mathbf{1} - q^G) \geq \mathbf{1} - q^G$, but we have already established that $\rho(B_G(\mathbf{1})) < 1$. Thus it must be the case that $(\mathbf{1} - q^G) = 0$. In other words, G is consistent. \square

D A bad example for infix probabilities

We now present a family of SCFGs, G_n , of size $O(n)$, and with critical-depth n , and we give a fixed 3-state DFA, D . We use these to indicate why it is likely to be difficult to overcome the exponential dependence on critical-depth of the given SCFG, G , in order to obtain a P-time algorithms for computing the probability (within desired precision) that an arbitrary G generates a string in $L(D)$.

The DFA D , is depicted in Figure 1. It has only 3 states and the property it checks is whether aa is an “infix” of the string. In other words, $L(D) = \{waaw' \mid w \in \Sigma^* \text{ and } w' \in \Sigma^*\}$. The family of SCFGs G_n is defined by the following rules:

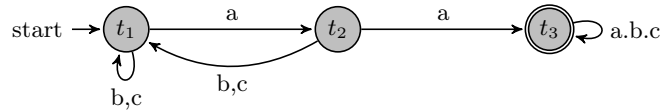


Fig. 1. Automaton for the infix aa

$$\begin{aligned}
 A_0 &\xrightarrow{0.5} A_0 A_0 \\
 A_0 &\xrightarrow{0.5} A_1 \\
 A_1 &\xrightarrow{0.5} A_1 A_1 \\
 A_1 &\xrightarrow{0.5} A_2
 \end{aligned}$$

$$\dots$$

$$A_n \xrightarrow{1} caB_nac$$

$$B_n \xrightarrow{1} B_{n-1}B_{n-1}$$

$$B_{n-1} \xrightarrow{1} B_{n-2}B_{n-2}$$

$$\dots$$

$$B_0 \xrightarrow{0.5} \epsilon$$

$$B_0 \xrightarrow{0.5} b$$

Proposition 7. $q^{G_n} = 1$. In other words, the probability of termination (generating a finite string) starting at any nonterminal in G_n is 1. Furthermore, $q_{(t_1 A_0 t_3)}^{G_n \otimes D} = \frac{1}{2}$ is the probability that this SCFG G_n , starting at A_0 , generates a string which has infix aa . On the other hand, $q_{(t_1 A_i t_3)}^{G_n \otimes D} = 2^{-2^i}$ is the same probability, starting at A_i .

The proof of this proposition is not at all difficult (using simple induction, and the formula for solving quadratic equations).

Let us argue why this causes severe difficulties for the approximate computation of $q^{G \otimes D}$. Note that $q_{(t_1 A_0 t_3)}^{G_n \otimes D} = \frac{1}{2}$ and $q_{(t_1 A_n t_3)}^{G_n \otimes D} = 2^{-2^n}$. However, in the product MPS $y = P_{G \otimes D}(y)$ the variable $y_{(t_1 A_0 t_3)}$ depends on the variable $y_{(t_1 A_n t_3)}$, and furthermore, if we, for example, “under-approximate” $q_{(t_1 A_n t_3)}^{G_n \otimes D} = 2^{-2^n}$, and instead set $y_{(t_1 A_n t_3)} := 0$, or, what effectively achieves the same result, if we change the product MPS by setting $P_{G \otimes D}(y)_{t_1 A_n t_3} \equiv 0$, then in the resulting modified MPS, with new LFP $\tilde{q}^{G_n \otimes D}$, we would get $\tilde{q}_{(t_1 A_0 t_3)}^{G_n \otimes D} = 0$.

Likewise, one can show that if we “over-approximate” $q_{(t_1 A_n t_3)}^{G_n \otimes D}$, even very slightly, setting $P_{G \otimes D}(y)_{t_1 A_n t_3} \equiv \frac{1}{2^{\text{poly}}}$ in a consistent way, then we will end up with a new LFP $\tilde{q}^{G_n \otimes D}$, such that $\tilde{q}_{(t_1 A_0 t_3)}^{G_n \otimes D} \approx 1$ (in other words, very close to 1).

In both cases, the resulting approximate solution $\tilde{q}_{(t_1 A_0 t_3)}^{G_n \otimes D}$ is terribly far from the actual solution $\frac{1}{2}$. (Note that this is irrespective of the algorithm that is used to compute the other probabilities.)

Furthermore, we can not in any way use the fact that we can detect in P-time and remove variables x_A from the PPS $x = P_{G_n}(x)$ for which $q_A^{G_n} = 1$, because indeed $q^G = 1$, and yet in the product $q^{G \otimes D}$ there are coordinates with wildly different probabilities that we wish to compute.