Analysis of Two-variable Recurrence Relations with Application to Parameterized Approximations

Ariel Kulik[*] Hadas Shachnai[†]

Abstract

In this paper we introduce randomized branching as a tool for parameterized approximation and develop the mathematical machinery for its analysis. Our algorithms improve the best known running times of parameterized approximation algorithms for Vertex Cover and 3-Hitting Set for a wide range of approximation ratios. One notable example is a simple parameterized random 1.5-approximation algorithm for Vertex Cover, whose running time of $O^*(1.01657^k)$ substantially improves the best known runnning time of $O^*(1.0883^k)$ [Brankovic and Fernau, 2013]. For 3-Hitting Set we present a parameterized random 2-approximation algorithm with running time of $O^*(1.0659^k)$, improving the best known $O^*(1.29^k)$ algorithm of [Brankovic and Fernau, 2012].

The running times of our algorithms are derived from an asymptotic analysis of a wide class of two-variable recurrence relations of the form:

$$p(b, k) = \min_{1 \leq j \leq N} \sum_{i=1}^{r_j} \bar{\gamma}_i^j \cdot p(b - \bar{b}_i^j, k - \bar{k}_i^j),$$

where $\bar{b}^j$ and $\bar{k}^j$ are vectors of natural numbers, and $\bar{\gamma}^j$ is a probability distribution over r_j elements, for $1 \leq j \leq N$. Our main theorem asserts that for any $\alpha > 0$,

$$\lim_{k \to \infty} \frac{1}{k} \log p(\alpha k, k) = - \max_{1 \leq j \leq N} M_j,$$

where M_j depends only on α, $\bar{\gamma}^j$, $\bar{b}^j$ and $\bar{k}^j$, and can be efficiently calculated by solving a simple numerical optimization problem. To this end, we show an equivalence between the recurrence and a stochastic process. We analyze this process using the *method of types*, by introducing an adaptation of Sanov's theorem to our setting. We believe our novel analysis of recurrence relations which is of independent interest is a main contribution of this paper.

[*]Computer Science Department, Technion, Haifa 3200003, Israel. E-mail: `kulik@cs.technion.ac.il`
[†]Computer Science Department, Technion, Haifa 3200003, Israel. E-mail: `hadas@cs.technion.ac.il`

1 Introduction

In search of tools for deriving efficient parameterized approximations, we explore the power of randomization in branching algorithms. Recall that a cover of an undirected graph $G = (V, E)$ is a subset $S \subseteq V$ such that for any $(u, v) \in E$ it holds that $S \cap \{u, v\} \neq \emptyset$. The *Vertex Cover* problem is to find a cover of minimum cardinality for G. In Vertex Cover parameterized by the solution size, k, we are given an integer parameter $k \geq 1$, and we wish to determine if G has a vertex cover of size k in time $O^*(f(k))$, for some computable function f.[1]

Consider the following simple algorithm for the problem. Recursively pick a vertex v of degree at least 3, and branch over the following two options: v is in the cover, or three of v's neighbors are in the cover. If the maximal degree is 2 or less then find a minimal vertex cover in polynomial time. The algorithm has a running time $O^*(1.4656^k)$ (see Chapter 3 in [15] for more details).

The randomized branching version of this algorithm replaces branching by a random selection with some probability $\gamma \in (0, 1)$. In each recursive call the algorithm selects either v or three of its neighbors into the solution, with probabilities γ and $1 - \gamma$, respectively (see Algorithm 1 for a formal description). If v is in a minimal cover then the algorithm has probability γ to decrease the minimal cover size by one, and probability $1 - \gamma$ to select three vertices into the solution, possibly with no decrease in the minimal cover size. A similar argument holds in case v is not in a minimal cover. This suggests that the function $p(b, k)$ defined in equation (1) lower bounds the probability the above algorithm returns a cover of size b, given a graph which has a cover of size k.

$$
\begin{aligned}
p(b, k) = \quad & \min \begin{cases} \gamma \cdot p(b-1, k-1) + & (1-\gamma) \cdot p(b-3, k) \\ \gamma \cdot p(b-1, k) \quad + & (1-\gamma) \cdot p(b-3, k-3) \qquad k \geq 3 \end{cases} \\
& p(b, k) = 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall b < 0, k \in \mathbb{Z} \\
& p(b, k) = 1 \qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall b \geq 0, k \leq 0
\end{aligned} \tag{1}
$$

Thus, for any $\alpha > 1$, we can obtain an α-approximation with constant probability by repeating the randomized branching process $\frac{1}{p(\alpha k, k)}$ times. While $p(b, k)$ can be evaluated using dynamic programming, for any $b, k \geq 0$, finding the asymptotic behavior of $\frac{1}{p(\alpha k, k)}$ as $k \to \infty$, which dominates the running time of our algorithm, is less trivial.

1.1 Our Results

In this paper we show that randomized branching is a highly efficient tool in the development of parameterized approximation algorithms for Vertex Cover and 3-Hitting Set, leading to significant improvements in running times over algorithms developed by using existing tools.[2] One notable example is a simple parameterized random 1.5-approximation algorithm for Vertex Cover, whose running time of $O^*(1.01657^k)$ substantially improves the currently best known $O^*(1.0883^k)$ algorithm for the problem [10].

To evaluate the running times of our algorithms, we develop mathematical tools for analyzing the asymptotic behavior of a wide class of two-variable recurrence relations generalizing the relation in (1). To this end, we introduce an adaptation of Sanov's theorem [31] (see also [13]) to our setting, which facilitates the use of *method of types* and *information theory* for the first time in the analysis of *branching* algorithms. We believe our novel analysis of recurrence relations which is of independent interest is a main contribution of this paper.

(a) Vertex Cover   (b) 3-Hitting Set

Figure 1: Results for Vertex Cover and 3-Hitting Set. A dot at (α, c) means that the respective algorithm outputs α-approximation in time $O^*(c^k)$ or $O^*\big((c+\varepsilon)^k\big)$ for any $\varepsilon > 0$.

1.1.1 Vertex Cover and 3-Hitting Set

We say that an algorithm $\mathcal{A}$ is a *parameterized random α-approximation for Vertex Cover* if, given a graph G and a parameter k, such that G has a vertex cover of size k, $\mathcal{A}$ returns a vertex cover S of G satisfying $|S| \le \alpha k$ with constant probability $\lambda > 0$, and has running time $O^*(f(k))$. We refer the reader to [19, 9, 26] for similar and more general definitions.

Vertex Cover: Our results for Vertex Cover include two parameterized random α-approximation algorithms, ENHANCEDVC3* and BETTERVC (presented in Sections 2 and 4, respectively). Algorithm ENHANCEDVC3* uses a single branching rule (either v or $N(v)$ are in a minimal cover) and has the best running times for approximation ratios greater than 1.4. We note that this simple algorithm outputs a 1.5-approximation in time $O^*(1.01657^k)$.

Algorithm BETTERVC is more complex. It is based on a parameterized $O^*(1.33^k)$ algorithm for Vertex Cover presented in [29]. BETTERVC achieves the best running times for approximation ratios smaller than 1.4. This algorithm shows that applying randomization in a sophisticated branching algorithm can result in an excellent tradeoff between approximation and time complexity for approximation ratios approaching 1.

The table below compares the running time of the best algorithm presented in this paper for a given approximation ratio to the previous best results due to Brankovic and Fernau [10]. A value of c for ratio α means that the respective algorithm yields an α-approximation with running time $O^*(c^k)$. The set of values selected for α matches the set of approximation ratios listed in [10].

ratio	1.1	1.2	1.3	1.4	1.5	1.666	1.75	1.8	1.9
BF [10]	1.235	1.197	1.160	1.1232	1.0883	1.0396	1.0243	1.0166	1.0051
This paper	1.160	1.096	1.058	1.0331	1.0166	1.0043	1.0016	1.00073	1.000083

Figure 1a shows a graphical comparison between our results and the previous best known results [10, 19].

---

[1] The notation O^* hides factors polynomial in the input size.

[2] See Section 1.1.1 for a formal definition of 3-Hitting Set.

3-Hitting Set: The input for 3-Hitting Set is a hypergraph $G = (V, E)$, where each hyperedge e contains at most 3 vertices, i.e., $|e| \leq 3$. We refer to such hypergraph as 3-*hypergraph*. We say that a subset $S \subseteq V$ is a *hitting set* if, for every $e \in E$, $e \cap S \neq \emptyset$. The objective is to find a hitting set of minimum cardinality. In the parameterized version, the goal is to determine if the input graph has a hitting set of at most k vertices, where $k \geq 1$ is the parameter.

We say that an algorithm $\mathcal{A}$ is a *parameterized random α-approximation for 3-Hitting Set* if, given a 3-hypergraph G and a parameter k, such that G has a hitting set of size k, $\mathcal{A}$ returns a hitting set S of G satisfying $|S| \leq \alpha k$ with constant probability $\lambda > 0$, and has running time $O^*(f(k))$.

In Section 3 we present a parameterized random α-approximation algorithm for 3-Hitting Set for any $1 < \alpha < 3$. The algorithm, 3HS (Algorithm 5) can be viewed as an adaptation of ENHANCEDVC3* to hypergraphs, using the following observation. For any $v \in V$ we define the *neighbors graph* of v as the hypergraph in which $\{u, w\}$ (or $\{u\}$) is an edge if $\{u, v, w\}$ ($\{u, v\}$) is an edge in the original hypergraph. It holds that for any hitting set S either $v \in S$ or S contains a hitting set of the neighbors graph of v. The actual branching rules of 3HS were determined via computer-aided search tree generation, using the above observation.

While 3HS may not be the best for approximation ratios close to 1, it yields a significant improvement over previous results for higher approximation ratios. For $\alpha = 2$ the running time is $O^*(1.0659^k)$, substantially improving the best known result of $O^*(1.29^k)$ due to [9]. Figure 1b gives a graphical comparison between the running times achieved in this paper and the results of [9] and [19].

We note that while our algorithms yield significant improvements in running times for both Vertex Cover and 3-Hitting Set over the algorithms of [9, 10] and [19], the previous algorithms are deterministic; our algorithms use randomization as a key tool.

1.1.2 Recurrence Relations

The objective of our algorithms is to find a cover of a graph under the restriction that this cover must not exceed a given budget. The algorithms consist of a recursive application of a random branching step. Each time the step is executed it adds vertices to the solution, thereby decreasing the available budget, and possibly reducing the number of vertices required to complete the solution. To analyze the running times of our algorithms, we need to evaluate the probability of obtaining a cover satisfying the budget constraint.

Similar to branching algorithms, this property can be formulated using a recurrence relation. In our case, the recurrence relation defines a function $p : \mathbb{Z} \times \mathbb{N} \to [0, 1]$ satisfying the following equations.[3]

$$
\begin{aligned}
p(b, k) &= \min_{\{1 \leq j \leq N \mid \; \bar{k}^j \leq k\}} \sum_{i=1}^{r_j} \bar{\gamma}_i^j \cdot p(b - \bar{b}_i^j, k - \bar{k}_i^j) \\
p(b, k) &= 0 \qquad\qquad\qquad\qquad\qquad\qquad \forall b < 0, k \in \mathbb{N} \\
p(b, 0) &= 1 \qquad\qquad\qquad\qquad\qquad\qquad \forall b \geq 0,
\end{aligned}
\tag{2}
$$

where $N \in \mathbb{N}$, and for any $1 \leq j \leq N$ the following hold: $\bar{b}^j \in \mathbb{N}_+^{r_j}$, $\bar{k}^j \in \mathbb{N}^{r_j}$ and $\bar{\gamma}^j \in \mathbb{R}_+^{r_j}$ with $\sum_{i=1}^{r_j} \bar{\gamma}_i^j = 1$. We say that $\bar{k}^j \leq k$ if $\bar{k}_i^j \leq k \; \forall 1 \leq i \leq r_j$. We refer to the recurrence relation in (2) as the **composite recurrence** of $\{(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j) \mid 1 \leq j \leq N\}$. Note that for the recurrence to be properly defined, there must be $1 \leq j \leq N$ such that $\bar{k}^j \leq 1$ (otherwise the min operation in (2) may be taken over an empty set). Throughout this section we use the word *term* when referring to triplets such as $(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)$.

In the context of our randomized branching algorithms, the number of terms, N, corresponds to the number of possible branching *states* (note, this is different than the number of *branching*

---

[3] Throughout the paper we use $\mathbb{N}$ (resp. $\mathbb{N}_+$) to denote the non-negative (resp. positive) integers ($\mathbb{N} = \mathbb{N}_+ \cup \{0\}$).

rules). For example, in Algorithm 1 (See Section 2 and an informal outline at the beginning of Section 1) there are two possible states: either v is in an optimal cover, or its neighbors are. Indeed, the analysis of the algorithm utilizes a composite relation with $N = 2$ as appears in (1).

To evaluate the running times of our algorithms we need to analyze the asymptotic behavior of $p(\alpha k, k)$ for a fixed α as k grows to infinity. With some surprise, we did not find an existing analysis of this behavior, even for $N = 1$. The main technical contribution of this paper is Theorem 2 that gives such analysis for any $N \geq 1$. We emphasize that while the recurrence relations we want to solve are derived from coverage problems, our solution is generic and can be used for any composite recurrence.

We say that a vector $\bar{q} \in \mathbb{R}^r_{\geq 0}$ is a *distribution* if $\sum_{i=1}^r \bar{q}_i = 1$ and use $D\left(\cdot\|\cdot\right)$ to denote **Kullback-Leibler divergence** [13].[4] To state our main result we need the next definition. For short, associate the term $(\bar{b}, \bar{k}, \bar{\gamma})$ with the expression $\sum_{i=1}^r \bar{\gamma}_i \cdot p(b - \bar{b}_i, k - \bar{k}_i)$.

Definition 1. *Let $\bar{b} \in \mathbb{N}^r_+$, $\bar{k} \in \mathbb{N}^r$ and $\bar{\gamma} \in \mathbb{R}^r_{\geq 0}$ with $\sum_{i=1}^r \bar{\gamma}_i = 1$. Then for $\alpha > 0$, the α-branching number of the term $(\bar{b}, \bar{k}, \bar{\gamma})$ is the optimal value M^* of the following minimization problem over $\bar{q} \in \mathbb{R}^r_{\geq 0}$:*

$$M^* = \min\left\{ \frac{1}{\sum_{i=1}^r \bar{q}_i \cdot \bar{k}_i} D\left(\bar{q}\|\bar{\gamma}\right) \,\middle|\, \sum_{i=1}^r \bar{q}_i \cdot \bar{b}_i \leq \alpha \sum_{i=1}^r \bar{q}_i \cdot \bar{k}_i, \ \bar{q} \text{ is a distribution} \right\} \tag{3}$$

If the optimization above does not have a feasible solution then $M^* = \infty$.

Our main result is the following.

Theorem 2. *Let p be the composite recurrence of $\{(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)|\ 1 \leq j \leq N\}$, and $\alpha > 0$. Denote by M_j the α-branching number of $(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)$, and let $M = \max\{M_j | 1 \leq j \leq N\}$. If $M < \infty$ then[5]*

$$\lim_{k \to \infty} \frac{\log p\left(\lfloor \alpha k \rfloor, k\right)}{k} = -M.$$

Intuitively, Theorem 2 asserts that $p(\alpha k, k) \approx \exp(-M)^k$. Furthermore, it shows that the asymptotics of $p(\alpha k, k) \approx \exp(-M)^k$ is dominated by the "worst" term in p. We note that the optimization problem (3) is *quasiconvex*. Furthermore, all of the numerical problems in this paper arising as consequences of (3) and Theorem 2 are *quasiconvex*, and as such can be solved efficiently using standard tools (these problems involve the optimization of $\bar{\gamma}^j$ as well). Moreover, most of these problems have a nearly closed form solution.

It is easy to show that for p as defined in (2) and every $b, k, n \in \mathbb{N}_+$ it holds that $p(nb, nk) \geq (p(b, k))^n$. This suggests that p can be lower bounded empirically by $p(\alpha k, k) = \Omega(c^k)$ where $c = (p(\alpha k_0, k_0))^{\frac{1}{k_0}}$ for any fixed k_0. Indeed, this simple approach can be used in practice to derive a fairly good lower bound for p in simple cases such as (1). However, it lacks both the scale and insight required to derive the algorithmic results presented in this paper. Furthermore, Theorem 2 readily gives the desired solution, thus eliminating the need for an empirical approach as described above.

The observation that the asymptotic behavior of $p(b, k)$ is dominated by the highest α-branching number of the terms in p served as a main guiding principle for designing the algorithms in this paper. Most notably, the 1.5-approximation for Vertex Cover was explicitly derived by this insight (see Section 2.2). In addition, Theorem 2 reduces the problem of optimizing the values of $\bar{\gamma}^j$ of the terms of p (e.g., the selection of γ in (1)) to multiple simple continuous quasiconvex optimization problems. In contrast, the empirical approach provides no tools for optimizing the distributions $\bar{\gamma}^j$. This was crucial for deriving all of our algorithmic results, in particular the results for 3-Hitting Set (see Section 3) which involve multiple (computer generated) branching rules.

---

[4]Formally, for $\bar{c}, \bar{d} \in \mathbb{R}^k$ define $D\left(\bar{c}\|\bar{d}\right) = \sum_{i=1}^k \bar{c}_i \log \frac{\bar{c}_i}{\bar{d}_i}$.

[5]Throughout the paper we refer by log to the natural logarithm.

The proof of Theorem 2 is given in Section 6, which is written as a stand-alone part in this paper (with the exception of Section 6.1, which gives some intuition to our proof technique).

We use in the proof of Theorem 2 several self-contained weaker results. The first of these results, Lemma 12, states that $\liminf_{k\to\infty} \frac{1}{k} \log p(\lfloor \alpha(1+\varepsilon)k \rfloor, k) \geq -M(1+\varepsilon)$ for any $\varepsilon > 0$. While the proof of this lemma is relatively short, it suffices for deriving all the algorithmic results presented in this paper.

We note that the requirement that the minimum operation in (2) is only taken over values of j such that $\bar{k}^j \leq k$ is indeed important: the statement of Theorem 2 may not hold if the requirement is removed due to several corner cases.[6] Algorithmically, the condition $\bar{k}^j \leq k$ represents the requirement that a branching step cannot lead to a negative size minimum cover for the remaining graph. Therefore, the condition always holds.

1.2 Tools and Techniques for the Analysis of Recurrence Relations

In the following we give an informal introduction to the tools and ideas used in the proof of Theorem 2 by studying "simpler" recurrence relations. In particular, we motivate the formula in (3). Given $(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)$ for $1 \leq j \leq N$ as in Theorem 2, define N "simpler" functions $p_j : \mathbb{Z} \times \mathbb{Z} \to [0, 1]$ for $1 \leq j \leq N$, satisfying

$$p_j(b, k) = \sum_{i=1}^{r_j} \bar{\gamma}_i^j \cdot p_j(b - \bar{b}_i^j, k - \bar{k}_i^j), \tag{4}$$

along with $p_j(b, k) = 1$ for any $b \geq 0$ and $k \leq 0$, and $p_j(b, k) = 0$ for $b < 0$. One can easily give a probabilistic interpretation for p_j. For $1 \leq j \leq N$, let $(Y_n^j)_{n=1}^{\infty}$ be a series of $i.i.d.$ random variables such that $\Pr(Y_n^j = i) = \bar{\gamma}_i^j$ for any $1 \leq i \leq r_j$. It can be easily shown that

$$p_j(b, k) = \Pr\left(\exists n : \sum_{\ell=1}^{n} \bar{b}_{Y_\ell^j}^j \leq b \text{ and } \sum_{\ell=1}^{n} \bar{k}_{Y_\ell^j}^j \geq k\right). \tag{5}$$

In the context of Vertex Cover, we can interpret $Y_n^j = i$ as the event: "In the nth step of the algorithm, $\bar{b}_i^j$ vertices were added to the cover and reduced the minimal vertex cover by $\bar{k}_i^j$". Within this interpretation, $p_j(b, k)$ is the probability that at some point in the algorithm execution the size of the minimal vertex cover has decreased by k while at most b vertices were added to the cover.

We utilize the *method of types* for analyzing the asymptotic behavior of p_j. The *type* $\mathcal{T}(a_1, \ldots, a_n)$ of $(a_1, \ldots, a_n) \in \{1, \ldots, r_j\}^n$ is the relative frequency of each $i \in \{1, \ldots, r_j\}$ in $(a_1, \ldots, a_n)$; that is, $\mathcal{T}(a_1, \ldots, a_n) = T \in \mathbb{R}_{\geq 0}^{r_j}$, where $T_i = \frac{|\{\ell| a_\ell = i\}|}{n}$. It follows that the type of a sequence is a distribution.

One of the important results attributed to the method of types is Sanov's theorem [31]. Given a set Q of distributions, i.e., $Q \subseteq \{\bar{q} \in \mathbb{R}_{\geq 0}^{r_j} | \bar{q} \text{ is a distribution}\}$, the theorem states (under a few technical conditions omitted here) that

$$\lim_{n\to\infty} \frac{1}{n} \log \Pr\left(\mathcal{T}(Y_1^j, \ldots, Y_n^j) \in Q\right) = -\min_{\bar{q} \in Q} D\left(\bar{q} \| \bar{\gamma}^j\right). \tag{6}$$

Informally, equation (6) implies that the probability the type of a random sequence is in Q is dominated by the distribution in Q closest to the distribution $\bar{\gamma}^j$ from which the sequence is drawn, with the distance measured by $D(\cdot \| \cdot)$, the Kullback-Leibler divergence.

---

[6] Consider, for example, the redundant recurrence $p(b, k) = p(b - 6, k - 4)$ with $p(b, k) = 0$ for $b < 0$ and $p(b, k) = 1$ for $k \leq 0 \leq b$. In this case, $\lim_{k\to\infty} \frac{1}{k} \log p(\lfloor 1.5k \rfloor, k)$ does not exist.

For $\beta > 0$ let Q_β^j be the set of all distributions $\bar{q}$ such that, if a random variable X is sampled according to $\bar{q}$ (i.e., $\Pr(X = i) = \bar{q}_i$), then it holds that $E[\bar{k}_X^j] \geq \beta$ and $E[\bar{b}_X^j] \leq \alpha\beta$. Formally,

$$Q_\beta^j = \left\{ \bar{q} \;\middle|\; \sum_{i=1}^{r_j} \bar{q}_i \bar{b}_i^j \leq \alpha\beta, \sum_{i=1}^{r_j} \bar{q}_i \bar{k}_i^j \geq \beta, \; \bar{q} \text{ is a distribution} \right\}.$$

Therefore, $\mathcal{T}(a_1, \ldots, a_n) \in Q_{\frac{k}{n}}^j$ iff $\sum_{\ell=1}^n \bar{k}_{a_\ell}^j \geq \frac{k}{n} \cdot n = k$ and $\sum_{\ell=1}^n \bar{b}_{a_\ell}^j \leq \alpha \frac{k}{n} \cdot n = \alpha k$. The sets Q_β^j can be used to write the event in (5) in terms of types.

$$p_j(\alpha k, k) = \Pr\left(\exists n : \sum_{\ell=1}^n \bar{b}_{Y_\ell^j}^j \leq b, \sum_{\ell=1}^n \bar{k}_{Y_\ell^j}^j \geq k \right) = \Pr\left(\exists n : \mathcal{T}(Y_1^j, \ldots, Y_n^j) \in Q_{\frac{k}{n}}^j \right)$$

It follows from Sanov's theorem that

$$\Pr\left(\mathcal{T}(Y_1^j, \ldots, Y_{\frac{k}{\beta}}^j) \in Q_\beta^j \right) \approx \exp\left(-\frac{k}{\beta} \cdot \min_{\bar{q} \in Q_\beta^j} D\left(\bar{q} \| \bar{\gamma}^j \right) \right).$$

Let $\beta^{j,*}, \bar{q}^{j,*} = \arg\min_{\beta, \bar{q} \in Q_\beta^j} \frac{1}{\beta} D\left(\bar{q} \| \bar{\gamma}^j \right)$. We can then lower bound $p_j(\alpha k, k)$ by focusing on sequences of length $n = \frac{k}{\beta^{j,*}}$ (we ignore integrality issues in this informal overview).

$$p_j(\alpha k, k) = \Pr\left(\exists n : \mathcal{T}(Y_1^j, \ldots, Y_n^j) \in Q_{\frac{k}{n}}^j \right)$$
$$\geq \Pr\left(\mathcal{T}(Y_1^j, \ldots, Y_{\frac{k}{\beta^{j,*}}}^j) \in Q_{\beta^{j,*}}^j \right) \approx \exp\left(-k \frac{1}{\beta^{j,*}} D\left(\bar{q}^{j,*} \| \bar{\gamma}^j \right) \right).$$

It is easy to show that we can limit in (5) the values of n such that $c^j \cdot b \leq n \leq b$, where c^j is a constant (in fact, $c^j = \left(\max_{1 \leq i \leq r_j} \bar{b}_i^j \right)^{-1}$). This can be used to establish a matching upper bound.

$$p_j(\alpha k, k) = \Pr\left(\exists n : \mathcal{T}(Y_1^j, \ldots, Y_n^j) \in Q_{\frac{k}{n}}^j \right) = \Pr\left(\exists c^j \alpha k \leq n \leq \alpha k : \mathcal{T}(Y_1^j, \ldots, Y_n^j) \in Q_{\frac{k}{n}}^j \right)$$
$$\leq \sum_{n = c^j \alpha k}^{\alpha k} \Pr\left(\mathcal{T}(Y_1^j, \ldots, Y_n^j) \in Q_{\frac{k}{n}}^j \right) \approx \sum_{n = c^j \alpha k}^{\alpha k} \exp\left(-k \cdot \frac{1}{\left(\frac{k}{n} \right)} \cdot \min_{\bar{q} \in Q_{\frac{k}{n}}^j} D\left(\bar{q} \| \bar{\gamma}^j \right) \right)$$
$$\leq \sum_{n = c^j \alpha k}^{\alpha k} \exp\left(-k \frac{1}{\beta^{j,*}} D\left(\bar{q}^{j,*} \| \bar{\gamma}^j \right) \right) = k \cdot \alpha(1 - c^j) \exp\left(-k \frac{1}{\beta^{j,*}} D\left(\bar{q}^{j,*} \| \bar{\gamma}^j \right) \right).$$

Hence, we can expect that $\lim_{k \to \infty} \frac{1}{k} \log p_j(\alpha k, k) = \frac{1}{\beta^{j,*}} D\left(\bar{q}^{j,*} \| \bar{\gamma}^j \right)$. It can be easily shown that $\beta^{j,*} = \sum_{i=1}^{r_j} \bar{q}_i^{j,*} \bar{k}_i^j$ (otherwise the values are not optimal, contradicting their definition). Thus, we conclude that $\frac{1}{\beta^{j,*}} D\left(\bar{q}^{j,*} \| \bar{\gamma}^j \right) = M_j$, where M_j is the α-branching number of $(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)$, as given in Definition 1.

Using the ideas in the above discussion, it can be shown that for any $\varepsilon > 0$,

$$\liminf_{k \to \infty} \frac{1}{k} \log p_j((\alpha + \varepsilon)k, k) \geq -M_j \quad \text{and} \quad \limsup_{k \to \infty} \frac{1}{k} \log p_j(\alpha k, k) \leq -M_j. \tag{7}$$

While (7) provides a clear indication for the asymptotic behavior of $p_j(\alpha k, k)$, it only makes a little progress in understanding the asymptotics of $p(\alpha k, k)$. By definition, it is expected that $p \leq p_j$ for $1 \leq j \leq N$. Thus, following (7) we expect that $\limsup_{k \to \infty} \frac{1}{k} \log p(\alpha k, k) \leq -M$,

6

where $M = \max_{1 \leq j \leq N} M_j$. Our main result, stated in Theorem 2, asserts that the asymptotic behavior of $\frac{1}{k} \log p(\alpha k, k)$ can also be *lower bounded* by $-M$.

Similar to the above sketch of analysis for p_j, the proof of Theorem 2 uses a probabilistic analysis based on the method of types. The stochastic process $(X_n)_{n=1}^{\infty}$ used in the proof is one in which X_n is drawn according to one of the distributions $\bar{\gamma}^j$, where j is selected by an (arbitrary) function of $X_1, \ldots, X_{n-1}$. Though the method of types is mostly used in the context of *i.i.d.* random variables, we show that some of the basic properties of types (as defined above) can be adapted to the process $(X_n)_{n=1}^{\infty}$. These properties are used for proving Theorem 2.

We note that the proof of Theorem 2 is inspired by the proof of Sanov's theorem. However, there is a significant difference between the proofs. In Sanov's theorem, properties of the set Q are used to derive a series of types $(T_n^j)_{n=1}^{\infty}$ such that $T_n^j \in Q$, T_j^n is a type of a length n sequence, and $T_n^j \to \bar{q}^{j,*}$. Then, the probability the type of a length n sequence is in Q is lower bounded by its probability to be T_n^j. The above steps cannot be applied to the new process $(X_n)_{n=1}^{\infty}$, due to the arbitrary distribution by which X_n is drawn. Instead, we use a generative approach which bears some similarity to the *probabilistic method* [2] (see Lemma 10).

1.3 Related Work

Vertex Cover is one of the fundamental problems in computer science, and a testbed for new techniques in parameterized complexity. The problem admits a polynomial time 2-approximation, which cannot be improved under the *Unique Games Conjecture (UGC)* [25]. Vertex Cover has been widely studied from the viewpoint of parameterized complexity. We say that a problem (with a particular parameter k) is *fixed-parameter tractable (FPT)* if it can be solved in time $f(k) \cdot \text{poly}(n)$, where f is some computable function depending only on k. Vertex Cover parameterized by the solution size is well known to be FPT (see, e.g., [29]). The fastest running time of an FPT algorithm for the problem is $O^*(1.273^k)$ due to Chen et at. [12]. It is also known that there is no $2^{o(k)} \cdot \text{poly}(n)$ algorithm for the problem, under the *exponential time hypothesis (ETH)*.

In [7] it was shown that there is no $(7/6 - \varepsilon)$ approximation for Vertex Cover with running time $O(2^{n^{1-\delta}})$ for any $\delta > 0$ under ETH. In [28] Manurangsi and Trevisan showed a $(2 - 1/O(r))$-approximation for the problem with running time $O^*(\exp(n \cdot 2^{-r^2}))$, improving upon earlier results of [4]. To the best of our knowledge, the existence of a $(2 - \varepsilon)$-approximation for Vertex Cover with running time $2^{o(n)}$ is still open.

The above results suggest that for $\alpha < 7/6$ subexponential α-approximation algorithms are unlikely to exist, and even as the approximation ratio approaches 2 the barrier of exponential running time remains unbreached. This motivates our study of parameterized α-approximation algorithms for Vertex Cover, for $1 < \alpha < 2$, whose running times are exponential in the solution size, k.

Brankovic and Fernau presented in [10] a branching algorithm that yields a parameterized 1.5-approximation for Vertex Cover with running time $O^*(1.0883^k)$. In [19] Fellows et at. presented an α-approximation algorithm whose running time is $O^*(1.273^{(2-\alpha)k})$, for any $1 \leq \alpha \leq 2$. A similar result was obtained in [8] using a different technique.

Similar to Vertex Cover, 3-Hitting Set cannot be approximated within a constant factor better than 3 under UGC [25], and there is no subexponential algorithm for the problem under ETH. The best known parameterized algorithm for the problem has running time of $O^*(2.076^k)$ [32]. Previous works on parameterized approximation for 3-Hitting Set resulted in an α-approximation in time $O^*(2.076^{k(3-\alpha)/2})$ due to [19], for any $1 \leq \alpha \leq 3$, and a 2-approximation in time $O^*(1.29^k)$ using a branching algorithm by Brankovic and Fernau [9].

Randomized branching is a well known approach for algorithm design (see, e.g, [5, 6, 27]). Often, the analysis of such algorithms is narrowed to evaluating the probability that in *every* branching step the algorithm makes a correct branching choice (in contrast, in our analysis the

aim is to bound the number of incorrect steps). This leads to a one-variable recurrence which can be simply solved. Randomized branching has been used for approximation in [4], along with a tailored analysis for the approximation ratio.

The idea of sampling leaves from a branching tree was studied in the past from a different perspective. Specifically, it was used in [16] to justify one-sided probabilistic polynomial algorithms as a computational model for branching algorithms. Within this model, the authors derived lower bounds for branching algorithms.

Previous works on parameterized approximations for both Vertex Cover and 3-Hitting Set either considered approximative preprocessing [19] or used approximative (worsening) steps within branching algorithms [9, 10]. While these techniques use the approximative step explicitly at given stages of the algorithm execution, in randomized branching the approximative step takes the form of an incorrect branching decision, which may add unnecessary vertices to the solution. As incorrect branching is not restricted to a specific stage, a degree of freedom is added to the number of *good paths* within the branching tree. This degree of freedom in turn increases the probability of finding an approximate solution. This gives some intuition to the improved performance of our algorithms.

1.3.1 Recurrence Relations and the Method of Types

The analysis of single variable recurrence relations (e.g., $f(n) = \sum_{i=1}^{N} f(n-a_i)$) is a cornerstone in the analysis of parameterized branching algorithms that is often included in introductory textbooks on parameterized algorithms (see, e.g., [29, 15]).

In [18] Eppstein introduced a technique for computing the asymptotic behavior of multivariate recurrences of the form $f(x) = \max_i \sum_j f(x - \delta_{i,j})$, where $f : \mathbb{Z}^d \to \mathbb{Z}$ and $\delta_{i,j} \in \mathbb{N}^d$. For any $t \in \mathbb{N}^d$, the technique shows how to compute a constant c such that $f(nt) \approx c^n$ up to a polynomial factor. The technique is based on a tight reduction of the multivariate recurrence to a solvable single variable recurrence. A matching lower bound to the result of the quasiconvex program is derived using a random walk, which bears some similarity to the reduction used in this paper from a recurrence to a stochastic process. Nevertheless, the analysis in this paper is significantly different.

The result in [18] is commonly used in the analysis parameterized algorithms, and specifically within the context of Measure and Conquer [21]. To the best of our knowledge, these works commonly utilize solutions for recurrences, in a wise and sophisticated manner, to derive running times for algorithms, but do not deal with solving the recurrence relations themselves.

We emphasize that the recurrences considered in [18] are different from the recurrences studied is this paper. The difference seems to be more than merely technical. The recurrences in [18] commonly measure the size of a branching tree, while our recurrence relations are aimed at bounding the number of leaves adhering to certain property within the tree. In fact, the size of the branching trees considered in this paper can be easily evaluated using standard single variable recurrence relations. We are not aware of other works relating to the analysis of similar multivariate recurrences.

The *method of types* is a powerful technique developed mostly within the context of information theory in a line of works, starting from the early works of Sanov [31] and Hoeffding [23]. The current form of the method is attributed to the works of Csiszar et al. [14]. Along with Sanov's theorem, the prominent results attained using the method of types are universal block coding and hypothesis testing (we refer the reader to the survey in [14] and to Chapter 11 in [13]). Though the method of types is considered a basic tool in information theory, it seems much less known in theoretical computer science.

1.4 Organization

Section 2 includes a technical introduction to randomized branching using several algorithms for Vertex Cover which gradually reveal the main algorithmic ideas presented in this paper. The algorithmic results for 3-Hitting Set and a more sophisticated algorithm for Vertex Cover are given in Sections 3 and 4. An overview of the numerical tools used to calculate the running times of our algorithms, based on Theorem 2, is given in Section 5. Section 6 gives the proof of Theorem 2. Finally, in Section 7 we discuss open problems and some directions for future work.

2 Our Technique: Warm-up

We start by completing the analysis of the algorithm presented in Section 1. A formal description of the algorithm, $VC3_\gamma$, is given in Algorithm 1. While the performance of Algorithm 1 can be significantly improved, as we show below, it demonstrates the main tools and concepts developed in this paper, and its analysis involves only few technical details. Interestingly, already this simple algorithm improves the previous state of the art results for a wide range of approximation ratios. Sections 2.1 and 2.2 present variants of Algorithm 1, which perform even better. Each section introduces some new ideas. The results of the algorithms presented in this section are depicted in Figure 2.

Clearly, Algorithm 1 has a polynomial running time. Also, it always returns a vertex cover of the input graph G. Let $\mathcal{G}_k$ be the set of graphs with a vertex cover of size k or less. Also, let $P_\gamma(b, k)$ be the minimal probability that Algorithm 1 returns a solution of size at most b, given a graph $G \in \mathcal{G}_k$. That is, $P_\gamma(b, k) = \min_{G \in \mathcal{G}_k} \Pr[\ |VC3_\gamma(G)| \le b\ ]$. By the arguments given in Section 1, it is easy to prove that $P(b, k) \ge p_\gamma(b, k)$, where $p_\gamma(b, k)$ is defined by the following recurrence relation.

$$
\begin{aligned}
p_\gamma(b, k) = \quad & \min \begin{cases} \gamma p_\gamma(b - 1, k - 1) + (1 - \gamma)p_\gamma(b - 3, k) & k \ge 1 \\ \gamma p_\gamma(b - 1, k) + (1 - \gamma)p_\gamma(b - 3, k - 3) & k \ge 3 \end{cases} \\
p_\gamma(b, k) = 0 & \hspace{5cm} \forall b < 0 \\
p_\gamma(b, k) = 1 & \hspace{5cm} \forall b \ge 0, k = 0
\end{aligned}
\tag{8}
$$

That is, p_γ is the composite recurrence of $\left\{ (\bar{b}^j, \bar{k}^j, \gamma^j) |\ j = 1, 2 \right\}$ with $\bar{b}^1 = \bar{b}^2 = (1, 3)$, $\bar{\gamma}^1 = \bar{\gamma}^2 = (\gamma, 1 - \gamma)$, $\bar{k}^1 = (1, 0)$ and $\bar{k}^2 = (0, 3)$. Note that in this case $N = 2$ and $r_1 = r_2 = 2$ (recall that a composite recurrence is defined in Section 1.1.2).

Hence, by repeating the execution of Algorithm 1 $p_\gamma(b, k)^{-1}$ times, we have a constant probability to find a cover of size b or less, for any $G \in \mathcal{G}_k$. This is achieved by using Algorithm 2, taking Algorithm 1 as $\mathcal{A}$ and $p = p_\gamma$. We call the resulting algorithm α-VC3.

Algorithm 1 $VC3_\gamma$

    Input: An undirected graph G

1: **if** G has a vertex v with degree 3 or more **then**
2:     Let u_1, u_2, u_3 be 3 of v's neighbors.
3:     With probability γ set $S = \{v\}$ and $S = \{u_1, u_2, u_3\}$ with probability $1 - \gamma$.
4:     Use a recursive call to evaluate $R = VC3_\gamma(G \setminus S)$, and return $R \cup S$.
5: **else** the maximal degree in G is not greater than 2
6:     Find an optimal cover S of G in polynomial time and return it.

We note that if $G \in \mathcal{G}_k$ then α-VC3 returns a cover of size at most αk with constant probability. Clearly, the running time of the algorithm is $O^*((p_\gamma(\alpha k, k))^{-1})$. We resort to Theorem 2 to obtain a better understanding of the running time.

Algorithm 2 α-APPROX

Input: An undirected graph G, a parameter k, an algorithm $\mathcal{A}$ and a recurrence relation p.

1: Evaluate $r = p(\alpha k, k)$ using dynamic programming.
2: Execute $\mathcal{A}(G)$ for r^{-1} times. Return the minimal cover found.

For any $\alpha > 1$ and $\gamma \in (0,1)$, we can calculate the α-branching numbers $M_1^{\alpha,\gamma}, M_2^{\alpha,\gamma}$ of $(\bar{b}^1, \bar{k}^1, \bar{\gamma}^1), (\bar{b}^2, \bar{k}^2, \bar{\gamma}^2)$, respectively, by numerically solving the optimization problem (3). Let $M^{\alpha,\gamma} = \max\{M_1^{\alpha,\gamma}, M_2^{\alpha,\gamma}\}$. Therefore, by Theorem 2 we have $\lim_{k\to\infty} \frac{\log p_\gamma(\alpha k, k)}{k} = -M^{\alpha,\gamma}$. Thus, for any $\varepsilon > 0$ and large enough k, it holds that $\frac{\log p_\gamma(\alpha k, k)}{k} > -M^{\alpha,\gamma} - \varepsilon$, and equivalently $(p_\gamma(\alpha k, k))^{-1} < \exp(M^{\alpha,\gamma} + \varepsilon)^k$. We conclude that the running time of α-VC3 is $O^*((p_\gamma(\alpha k, k))^{-1}) = O^*(\exp(M^{\alpha,\gamma} + \varepsilon)^k)$ for any $\varepsilon > 0$.

For any $\alpha > 1$, we can numerically find the value of γ for which $M^{\alpha,\gamma}$ is minimal. Let γ_α be this value. Then, for any $\alpha > 1$ algorithm α-VC3 is a parameterized random α-approximation for Vertex Cover with running time $O^*(\exp(M^{\alpha,\gamma_\alpha} + \varepsilon)^k)$ (for any $\varepsilon > 0$). For example, for $\alpha = 1.5$ we get that α-VC3 has a running time of $O^*(1.043642^k)$. In Figure 2 the value of $\exp(M^{\alpha,\gamma_\alpha})$ is presented as a function of α. An overview of the methods used for the numerical optimizations is given in Section 5.

2.1 A Refined Analysis of Incorrect Branching

Standard branching algorithms derive several simpler sub-instances from a given instance with a guarantee that an optimal solution to one (specific yet unknown) of the sub-instances leads to an optimal solution. Therefore, the analysis is focused on this specific sub-instance and ignores the effect of other sub-instances on the optimum. This is not the case when using randomized branching for *approximation*, where the reduction in the minimal cover size by an incorrect branching can lead to an improved running time, as we demonstrate below.

Consider the following observation. If v is a vertex of degree exactly 3 and the algorithm (e.g., Algorithm 1) selects its three neighbors $\{u_1, u_2, u_3\}$ to the cover, then even if none of $\{u_1, u_2, u_3\}$ belongs to an optimal cover, the size of the optimal cover decreases by one (as v is a part of an optimal cover, but is no more required). This observation can be extended to any fixed degree of v.

Algorithm 3 takes advantage of this property by using a different probability for selecting v or its neighbors depending on its degree, as well as selecting all the neighbors of v in case the degree of v is smaller than Δ, for some fixed $\Delta \in \mathbb{N}$.

Algorithm 3 VC3*$_{\gamma_3,\gamma_4,\dots,\gamma_\Delta}$

    Input: An undirected graph G

1: **if** G has a vertex v with degree 3 or more **then**
2:     If $d < \Delta$ let $U = N(v)$, otherwise let U be a subset of $N(v)$ of size exactly Δ.
3:     With probability γ_d set $S = \{v\}$ and $S = U$ with probability $1 - \gamma_d$.
4:     Use a recursive call to evaluate $R = \text{VC3*}_{\gamma_3,\gamma_4,\dots,\gamma_\Delta}(G \setminus S)$, and return $R \cup S$.
5: **else** the maximal degree in G is 2
6:     Find an optimal cover S of G in polynomial time and return S.

Clearly, Algorithm 3 is polynomial and always returns a cover of G. Similar to Algorithm 1, it can be shown that the probability Algorithm 3 returns a solution of size b, given a graph

Figure 2: Results of Section 2. A dot at (α, c) means that the respective algorithm provides α-approximation for Vertex Cover with running time $O^*(c^k)$ or $O^*\big((c+\varepsilon)^k\big)$ for every $\varepsilon > 0$.

$G \in \mathcal{G}_k$, is at least $p_{\gamma_3, \gamma_4, \dots, \gamma_\Delta}(b, k)$, where $p_{\gamma_3, \gamma_4, \dots, \gamma_\Delta}$ is given by

$$p_{\gamma_3, \gamma_4, \dots, \gamma_\Delta}(b, k) = \min \begin{cases} \gamma_d \cdot p(b-1, k-1) \; + \; (1-\gamma_d) \cdot p(b-d, k-1) & 3 \le d < \Delta \\ \gamma_d \cdot p(b-1, k) \; + \; (1-\gamma_d) \cdot p(b-d, k-d) & 3 \le d < \Delta, k \ge d \\ \gamma_\Delta \cdot p(b-1, k-1) \; + \; (1-\gamma_\Delta) \cdot p(b-\Delta, k) \\ \gamma_\Delta \cdot p(b-1, k) \; + \; (1-\gamma_\Delta) \cdot p(b-\Delta, k-\Delta) & k \ge \Delta \end{cases}$$

$$(9)$$

with $p_{\gamma_3, \gamma_4, \dots, \gamma_\Delta}(b, k) = 0$ for $b < 0$ and $p_{\gamma_3, \gamma_4, \dots, \gamma_\Delta}(b, k) = 1$ for $b \ge 0$ and $k = 0$. Clearly, $p_{\gamma_3, \gamma_4, \dots, \gamma_\Delta}$ is a composite recurrence relation of the $N = 2(\Delta - 2)$ triplets

$$\{ \; ((1,d), (1,1), (\gamma_d, 1-\gamma_d)) \mid 3 \le d < \Delta \; \} \; \cup$$
$$\{ \; ((1,d), (0,d), (\gamma_d, 1-\gamma_d)) \mid 3 \le d < \Delta \; \} \; \cup$$
$$\{ \; ((1,\Delta), (1,0), (\gamma_\Delta, 1-\gamma_\Delta)), \; ((1,\Delta), (0,\Delta), (\gamma_\Delta, 1-\gamma_\Delta)) \; \}.$$

And as before, we can derive an approximation algorithm by using Algorithm 2 with Algorithm 3 as $\mathcal{A}$ and $p = p_{\gamma_3, \gamma_4, \dots, \gamma_\Delta}$. Let α-VC3* be this algorithm. Clearly, α-VC3* is a random parameterized α-approximations algorithm for Vertex Cover.

Arbitrarily, we select $\Delta = 100$. As before, for every $1 < \alpha < 2$ and $1 \le d < \Delta$ we can find the value $\gamma_{\alpha, d}$ such that the maximal α-branching number of $((1,d), (1,1), (\gamma_{\alpha, d}, 1-\gamma_{\alpha, d}))$ and $((1,d), (0,d), (\gamma_{\alpha, d}, 1-\gamma_{\alpha, d}))$ is minimal. Let $M_{\alpha, d}$ be this value. Also, we can find the value $\gamma_{\alpha, \Delta}$ such that the maximal α-branching number of $((1,\Delta), (1,0), (\gamma_{\alpha, \Delta}, 1-\gamma_{\alpha, \Delta}))$ and $((1,\Delta), (0,\Delta), (\gamma_{\alpha, \Delta}, 1-\gamma_{\alpha, \Delta}))$ is minimal and let $M_{\alpha, \Delta}$ be this value. Let M_α be the maximal branching number of these triplets for a given value of α ($M_\alpha = \max_{1 \le d \le \Delta} M_{\alpha, d}$). Then by Theorem 2, for any $\varepsilon > 0$ and large enough k, it holds that $p(\alpha k, k) \ge \exp(-M_\alpha - \varepsilon)$, and therefore the running time of α-VC3* is $O^*\big(\exp(M_\alpha + \varepsilon)^k\big)$. For $\alpha = 1.5$ the running time is $O^*(1.0172^k)$. Figure 2 shows $\exp(M_\alpha)$ as a function of α.

2.2 Further Insights from using α-Branching Numbers

In the context of classic branching algorithms, the running time of an algorithm is dominated by the highest branching number of the branching rules used by the algorithm (see, e.g., [29, 15]). This observation is commonly used in the design of (exact) branching algorithms. Theorem

11

2 asserts that essentially the same holds for parameterized approximation using randomized branching. In the following we show how to use it to improve the running time of VC*. With some surprise, we were unable to find a similar result referring to the recurrence relations in [18].

Consider algorithm α-VC3* of Section 2.1, whose time complexity is the inverse of the function in (9). As an example, for $\alpha = 1.5$ we can sort the values $M_{\alpha,d}$ to understand which value of d dominates the running time. We show the nine highest values in the table below.

d	5	6	4	7	8	9	10	11	3
$\exp(M_{1.5,d})$	1.0171	1.0165	1.0164	1.0156	1.0146	1.0136	1.0128	1.0120	1.0118

This suggests that avoiding branching over degree 5 vertices leads to an $O^*(1.0165^k)$ algorithm. In fact, tools to do so have already been used in previous works, such as [30]. The basic idea is that as long as there is a vertex v in the graph of degree different than 5 the algorithm branches on it. If all vertices in the graph are of degree 5 the algorithm has to perform a branching on a degree 5 vertex; however, such event cannot happen more than once along a branching path. Therefore, the algorithm can use non-randomized branching in this case while maintaining a polynomial running time.

Algorithm 4 ENHANCEDVC3*

Input: An undirected graph $G = (V, E)$
Configuration Parameters: The algorithm depends on several parameters that should be configured. These include $\Delta \in \mathbb{N}$, $\delta \in \mathbb{N}$, $2 \leq \delta < \Delta$, and $\gamma_2, \ldots, \gamma_{\delta-1}, \gamma_{\delta+1}, \ldots, \gamma_\Delta \in (0, 1)$.

1: If the empty set is a cover of G return $\emptyset$.
2: **if** G is not connected **then**
3:     Let C be a component of G. Return ENHANCEDVC3*$(C) \cup$ ENHANCEDVC*$(G - C)$.
4: If G has a vertex v of degree 1, let u be its neighbor. Return ENHANCEDVC3*$(G \backslash \{u\}) \cup \{u\}$.
5: **if** G has a vertex v of degree $d \neq \delta$ **then**
6:     Let $U = N(v)$ if $d < \Delta$ and $U \subseteq N(v)$ with $|U| = \Delta$ otherwise.
7:     Let $S = \{v\}$ with probability γ_d and $S = U$ otherwise.
8:     Return ENHANCEDVC3*$(G \setminus S) \cup S$
9: If G is a regular graph (of degree δ), select an arbitrary edge $(v_1, v_2) \in E$. Evaluate $S_1 = $ ENHANCEDVC3*$(G \setminus v_1) \cup v_1$ and $S_2 = $ ENHANCEDVC3*$(G \setminus v_2) \cup v_2$. Return the smaller set between S_1 and S_2.

Consider Algorithm 4. It can be shown that its running time is polynomial (similar to the proof of Lemma 4 in Section 4), and the probability that the algorithm returns a solution of size b, given $G \in \mathcal{G}_k$, is at least

$$p(b, k) = \min \begin{cases} \gamma_d \cdot p(b-1, k-1) + (1 - \gamma_d) \cdot p(b-d, k-1) & 3 \leq d < \Delta, d \neq \delta \\ \gamma_d \cdot p(b-1, k) + (1 - \gamma_d) \cdot p(b-d, k-d) & 3 \leq d < \Delta, d \neq \delta, k \geq d \\ \gamma_\Delta \cdot p(b-1, k-1) + (1 - \gamma_\Delta) \cdot p(b-\Delta, k) & \\ \gamma_\Delta \cdot p(b-1, k) + (1 - \gamma_\Delta) \cdot p(b-\Delta, k-\Delta) & k \geq d \end{cases} \quad (10)$$

As before, we use the lower bound derived from the recurrence relation to obtain a random parameterized α-approximation algorithm as with running time $O^*\left(\frac{1}{p(\alpha k, k)}\right)$ by using Algorithm 2 with Algorithm 4 as $\mathcal{A}$ and the recurrence relation p as given in (10). Let α-ENHANCEDVC3* be this algorithm.

For any $1 < \alpha < 2$ and $2 \leq d \leq \Delta$ we can find the value $M_{\alpha,d}$ as in Section 2.1. If $\delta' = \arg\max_{2 \leq d \leq N} M_{\alpha,d} \neq \Delta$ we can set $\delta = \delta'$; therefore, the run time of α-ENHANCEDVC3* is $O^*(\exp(M_\alpha + \varepsilon)^k)$ when M is the *second* largest number of $M_{\alpha,2}, \ldots, M_{\alpha,\Delta-1}$ (or $M_{\alpha,\Delta}$ if

Figure 3: An example of a neighbors graph. A hypergraph H is illustrated in 3a. The neighbors graph of v_1, $\mathsf{NG}(v_1)$, is given in 3b.

$\delta' = \Delta$). The value of $\exp(M_\alpha)$ as a function of α is shown in Figure 2. For $\alpha = 1.5$ the run time of the algorithm is $O^*(1.01657^k)$. This is the best running time for the given approximation ratio presented in this paper. The following table compares the running times of α-ENHANVEDVC3* and α-VC3* for several values of α.

α	1.2	1.3	1.4	1.5	1.6	1.7
α-VC3*	1.12548^k	1.06804^k	1.03501^k	1.01713^k	1.00754^k	1.00280^k
α-ENHANCEDVC3*	1.12386^k	1.06420^k	1.03320^k	1.01657^k	1.00751^k	1.00277^k

3   Application for 3-Hitting Set

In this section we present a parameterized approximation algorithm for 3-Hitting Set. The algorithm draws some ideas from VC3* (see Section 2.1), which relies on two basic observations. The first is that for any vertex v of a graph G and a vertex cover S, either $v \in S$ or $N(v) \subseteq S$. The second observation is that, even if v is in a minimum vertex cover, removing $N(v)$ from the graph decreases the size of a minimum cover at least by one.

Consider the following analog of the above statement for 3-Hitting Set. Given a 3-hypergraph $H = (V, E)$, for any $v \in V$ define the *neighbors graph* of v as the hypergraph $\mathsf{NG}(v) = (V_v, E_v)$ with $V_v = \{u | \exists e \in E : u, v \in e\}$ and $E_v = \{e \setminus \{v\} | e \in E, v \in e\}$ (see an example in Figure 3). Clearly, for every $e \in \mathsf{NG}(v)$ it holds that $|e| \leq 2$ (the neighbors graph is essentially a standard undirected graph with the addition of single node edges). Similar to the case of Vertex Cover, for any $v \in V$ and a hitting set S of H, either $v \in S$ or there is a minimal hitting set T of $\mathsf{NG}(v)$ such that $T \subseteq S$.[7] Also, if v belongs to a minimum hitting set of H then removing a minimal hitting set of $\mathsf{NG}(v)$ from H decreases the minimum hitting set size at least by 1.

Let $v \in V$ such that $\{v\} \notin E$, then the neighbors graph of v admits a specific structure. It has up to $2 \cdot \deg(v)$ vertices, exactly $\deg(v)$ edges (there may be edges with a single vertex) and no isolated vertices. Therefore, the number of possible graphs $\mathsf{NG}(v)$ for vertices of bounded degree is finite up to isomorphism.

For some fixed $\Delta \in \mathbb{N}$, we construct a set $\mathcal{G}_\Delta$ of hypergraphs, such that $\mathsf{NG}(v)$ is isomorphic to a hypergraph in $\mathcal{G}_\Delta$ for any v with $\deg(v) \leq \Delta$. Let $\mathcal{G}'_\Delta$ be the set of hypergraphs (V, E) with no isolated vertices, such that $V \subseteq \{1, 2, \ldots, 2\Delta\}$, $|E| \leq \Delta$, and $|e| \leq 2 \; \forall e \in E$. Let $\mathcal{G}_\Delta \subseteq \mathcal{G}'_\Delta$ be a minimal set of hypergraphs such that for any $G' \in \mathcal{G}'_\Delta$ there is $G \in \mathcal{G}_\Delta$ that is isomorphic to G'. Thus, $\mathcal{G}_\Delta$ can be derived from $\mathcal{G}'_\Delta$ by removing isomorphic hypergraphs. It is easy to see that the set $\mathcal{G}_\Delta$ is finite.   Also, for every $G \in \mathcal{G}_\Delta$ let $C_1^G, \ldots, C_{m^G}^G$ be all the minimal hitting sets of G. Clearly, the set $\{C_i^G | G \in \mathcal{G}_\Delta, 1 \leq i \leq m^G\}$ has a finite cardinality.

---

[7]A set T is a minimal hitting set of a hypergraph H if T is a hitting set and no strict subset $T' \subsetneq T$ is also a hitting set of H.

13

We need one more technical definition before introducing our algorithm. Given a 3-hypergraph $H = (V, E)$, a vertex $v \in V$ and $F \subseteq E$ such that $v \in e$ for any $e \in F$, define the *induced graph* of v and F as the hypergraph $\mathsf{Ind}(v, F) = (V_{v,F}, E_{v,F})$ with $V_{v,F} = \{u \mid \exists e \in F : u \in e \setminus \{v\}\}$ and $E_{v,F} = \{e \setminus \{v\} \mid e \in F\}$. By definition, it also holds that the cardinality of edges in $\mathsf{Ind}(v, F)$ is at most 2 and $\mathsf{Ind}(v, F)$ has no isolated vertices. It follows that $\mathsf{NG}(v) = \mathsf{Ind}(v, \{e \in E \mid v \in e\})$. Our algorithm uses induced graphs to handle vertices of degree larger than Δ. Similar to the neighbors graph, the induced graph $\mathsf{Ind}(v, F)$ satisfies the following. Let S be a hiting set of the hypergraph H, then either $v \in S$ or there is a hitting set T of $\mathsf{Ind}(v, F)$ such that $T \subseteq S$.

Algorithm 5 3HS

Input: A 3-hypergraph $H = (V, E)$

Configuration Parameters: $\bar{\gamma}^G \in \mathbb{R}^{m^G+1}_{\geq 0}$ with $\sum_{i=1}^{m^G+1} \bar{\gamma}_i^G = 1$ for any $G \in \mathcal{G}_\Delta$.

Notation: Define $H \setminus U = (V', E')$ with $V' = V \setminus U$ and $E' = \{e \in E \mid e \cap U = \emptyset\}$

1: If the empty set is a hitting set of H return $\emptyset$.
2: If there is $\{v\} \in E$ then return $\mathrm{3HS}(H \setminus \{v\}) \cup \{v\}$.
3: Pick an arbitrary vertex v. If $\deg(v) \leq \Delta$ set $N = \mathsf{NG}(v)$. Otherwise, set $N = \mathsf{Ind}(v, F)$ with an arbitrary set $F \subseteq E$ of Δ edges such that $\forall e \in F : v \in e$.
4: Find a hypergraph $G \in \mathcal{G}_\Delta$ such that N and G are isomorphic. Let φ be the vertex isomorphism function from G to N.
5: Select $S = \{v\}$ with probability $\bar{\gamma}_{m^G+1}^G$ and $S = \varphi(C_i^G)$ with probability $\bar{\gamma}_i^G$ for $1 \leq i \leq m^G$. Return $\mathrm{3HS}(H \setminus S) \cup S$.

The above observations are used to derive Algorithm 5. It is easy to see that the algorithm always returns a hitting set of the input hypergraph H. Also, the size of H strictly decreases between recursive calls, and the processing time of each recursive call is polynomial. Therefore, the algorithm has polynomial running time (note that since Δ is a fixed constant, finding a graph G isomorphic to N takes constant time). It is also easy to verify the algorithm indeed always finds an hypergraph $G \in \mathcal{G}_\Delta$ isomorphic to N in Line 4.

Consider the following recurrence relation:

$$p(b, k) =$$
$$\min \begin{cases} \bar{\gamma}_{m^G+1}^G \cdot p(b-1, k) + \\ \quad + \sum_{i=1}^{m^G} \bar{\gamma}_i^G \cdot p\left(b - |C_i^G|, k - |C_i^G \cap C_j^G|\right) & \forall G \in \mathcal{G}_\Delta, 1 \leq j \leq m^G : |C_j^G| \leq k \\ \bar{\gamma}_{m^G+1}^G \cdot p(b-1, k-1) + \\ \quad + \sum_{i=1}^{m^G} \bar{\gamma}_i^G \cdot p\left(b - |C_i^G|, k - \mathbb{1}_{\|G\| < \Delta}\right) & \forall G \in \mathcal{G}_\Delta, 1 \leq j \leq m^G \\ p(b-1, k-1) \end{cases} \tag{11}$$

Also, $p(b, k) = 0$ for $b < 0$, and $p(b, 0) = 1$ for $b \geq 0$. Let $\|G\|$ be the number of edges in G. We set $\mathbb{1}_{\|G\| < \Delta} = 1$ if $\|G\| < \Delta$ and $\mathbb{1}_{\|G\| < \Delta} = 0$ otherwise. Let $P(b, H)$ be the probability that Algorithm 5 returns a hitting set of size b or less, given the 3-hypergraph H. With a slight abuse of notation, let $P(b, k)$ the minimal (infimum) value of $P(b, H)$ for a 3-hypergraph H which has a hitting set of size k or less. The next lemma follows easily from the above discussion. We give a formal proof for completeness.

Lemma 3. *For every $b \in \mathbb{Z}$ and $k \in \mathbb{N}$, $P(b, k) \geq p(b, k)$.*

Proof. We prove the claim by induction on b. For $b < 0$ we have $P(b, k) = 0 = p(b, k)$, therefore the claim holds. For $b \in \mathbb{N}$, assume the claim holds for any smaller value of b. Let $k \in \mathbb{N}$, and H a 3-hypergraph with a hitting set T, $|T| \leq k$. If the algorithm returns $\emptyset$ (Line 2 of the algorithm) then $P(b, H) = 1 \geq p(b, k)$. Also, if there is an edge $\{v\} \in E$ then $v \in T$ (otherwise it is not an hitting set), and therefore $T \setminus \{v\}$ is a hitting set of $H \setminus \{v\}$. Thus,

$$P(b, H) \geq P(b-1, H \setminus \{v\}) \geq P(b-1, k-1) \geq p(b-1, k-1) \geq p(b, k).$$

14

Otherwise, let v be the vertex selected in Line 3 of the algorithm, let N be the selected hypergraph, $G \in \mathcal{G}_\Delta$ the hypergraph isomorphic to N, φ the vertex isomorphism from G to N, and S the randomly selected set in Line 5.

If $v \in T$, note that the set $T \setminus \{v\}$ is a hitting set of $H \setminus \{v\}$; thus, $H \setminus \{v\}$ has a hitting set of size $k - 1$ (or less). Also, if it further holds that $\|G\| < \Delta$ then $N = \mathsf{NG}(v)$. In this case, we have that $T \setminus \{v\}$ is a hitting set of $H \setminus \varphi(C_i^G)$ for all $1 \leq i \leq m^G$. Let e be an edge in $H \setminus \varphi(C_i^G)$. If $v \in e$ then $e \setminus \{v\}$ is an edge in N. As $\varphi(C_i^G)$ is a hitting set of N we have $e \cap \varphi(C_i^G) \neq \emptyset$; thus, e cannot be an edge in $H \setminus \varphi(C_i^G)$. If $v \notin e$ then since $e \cap T \neq \emptyset$, we also have $e \cap (T \setminus \{v\}) \neq \emptyset$. It follows that the probability Algorithm 5 returns a hitting set of size b or less given H is at least

$$P(b, H) \geq \bar{\gamma}_{m^G+1}^G P(b-1, H \setminus \{v\}) + \sum_{i=1}^{m^G} \bar{\gamma}_i^G P\left(b - |C_i^G|, H \setminus \varphi(C_i^G)\right)$$

$$\geq \bar{\gamma}_{m^G+1}^G P(b-1, k-1) + \sum_{i=1}^{m^G} \bar{\gamma}_i^G P\left(b - |C_i^G|, k - \mathbb{1}_{\|G\|<\Delta}\right)$$

$$\geq \bar{\gamma}_{m^G+1}^G p(b-1, k-1) + \sum_{i=1}^{m^G} \bar{\gamma}_i^G p\left(b - |C_i^G|, k - \mathbb{1}_{\|G\|<\Delta}\right) \geq p(b, k).$$

It remains to handle the case where $v \notin T$. Let F be the set of edges selected in Line 3 of the algorithm if $\deg(v) > \Delta$, and $F = \{e \in E | v \in e\}$ if $\deg(v) \leq \Delta$. Then $N = \mathsf{Ind}(v, F) = (V_{v,F}, E_{v,F})$. For any $e \in E_{v,F}$ it holds that $e \cup \{v\} \in F$; therefore, $e \cap T = (e \cup \{v\}) \cap T \neq \emptyset$. Thus, T contains a set $T_v \subseteq T$ such that T_v is a hitting set of N. W.l.o.g., we may assume that T_v is a minimal hitting set. Then $\varphi^{-1}(T_v)$ is a minimal vertex cover of G. Hence, there is $1 \leq j \leq m^G$ such that $\varphi^{-1}(T_v) = C_j^G$, and equivalently $T_v = \varphi(C_j^G)$.

The hypergraph $H \setminus S$ has a hitting set of size $|T \setminus (T \cap S)| \leq k - |T \cap S|$. For $S = \{v\}$ we have $|T \cap S| = |\emptyset| = 0$, and for $S = \varphi(C_i^G)$,

$$|T \cap S| \geq |T_v \cap S| = |\varphi(C_j^G) \cap \varphi(C_i^G)| = |C_j^G \cap C_i^G|.$$

Therefore,

$$P(b, H) \geq \bar{\gamma}_{m^G+1}^G P(b-1, H \setminus \{v\}) + \sum_{i=1}^{m^G} \bar{\gamma}_i^G P\left(b - |C_i^G|, H \setminus \varphi(C_i^G)\right)$$

$$\geq \bar{\gamma}_{m^G+1}^G \cdot P(b-1, k) + \sum_{i=1}^{m^G} \bar{\gamma}_i^G \cdot P\left(b - |C_i^G|, k - |C_i^G \cap C_j^G|\right)$$

$$\geq \bar{\gamma}_{m^G+1}^G \cdot p(b-1, k) + \sum_{i=1}^{m^G} \bar{\gamma}_i^G \cdot p\left(b - |C_i^G|, k - |C_i^G \cap C_j^G|\right) \geq p(b, k)$$

Hence, $P(b, H) \geq p(b, k)$ for any 3-hypergraph H with a hitting set of size k or less. We conclude that $P(b, k) \geq p(b, k)$. $\qquad \square$

Following the above analysis, an α-approximation algorithm for 3-Hitting Set can be derived by the same approach used for Vertex Cover. This leads to Algorithm 6.

It follows from Lemma 3 that Algorithm 6 yields an α-approximation for 3-Hitting Set with running time of $\frac{1}{p(\alpha k, k)}$. For any value of α, it is possible to optimize the value of $\bar{\gamma}^G$ for each $G \in \mathcal{G}_\Delta$ and evaluate the asymptotic behavior of $p(\alpha k, k)$ as k goes to infinity using Theorem 2.

Algorithm 6 α-HS

  Input: A 3-hypergraph H, a parameter k

1: Evaluate $r = p(\alpha k, k)$ using dynamic programming (p is defined in (11)).
2: **loop** r^{-1} times
3:   Execute 3HS(H).
4: Return the minimal hitting set found.

However, the size of $\mathcal{G}_\Delta$ grows rapidly as Δ increases, rendering the above computation less and less practical. With a little technical sophistication we were able to evaluate the running time of the algorithm with $\Delta = 7$ for various approximation ratios. Figure 1 shows the running times of the algorithm with $\Delta = 7$ as function of α. A list of running times for several approximation ratios is given in the table below. For $\alpha = 2$ the running time is $O^*(1.0659^k)$, yielding a significant improvement over the previous best result of $O^*(1.29^k)$ due to [9].

α	1.2	1.4	1.6	1.8	2.0	2.2	2.4	2.6	2.8
$\alpha - HS$	1.59^k	1.29^k	1.18^k	1.11^k	1.0659^k	1.039^k	1.021^k	1.0085^k	1.0026^k

4 Advanced Randomized Branching for Vertex Cover

In this section we give a parameterized approximation algorithm for Vertex Cover building on the exact $O^*(1.33^k)$ algorithm presented in [29]. That is, we analyze below a variant of the algorithm in which branching is replaced by selection of one of the branches *randomly*. The analysis shows that randomized branching in conjunction with faster parameterized algorithms can lead to faster parameterized approximation algorithms. We use below ideas presented in Section 2 and give the technical details for their implementation in a more advanced settings.

Consider Algorithm 7. It is easy to see that the algorithm always returns a cover of the input graph G.

Lemma 4. *Algorithm 7 has a polynomial running time.*

The following lemma lower bounds the probability that the algorithm returns a small cover.

Lemma 5. *Let $G \in \mathcal{G}_k$ ($\mathcal{G}_k$ is the set of graphs with vertex cover of size k or less), then the probability that Algorithm 7 returns a cover of size b or less is greater or equal to $p(b,k)$, where*

$p(b,k) = \min$

$$
\begin{cases}
p(b-1, k-1) & \\
p(b-2, k-2) & k \geq 2 \\
\gamma_d \cdot p(b-1, k-1) \;+\; (1-\gamma_d) \cdot p(b-d, k-1) & 5 \leq d < \Delta \\
\gamma_d \cdot p(b-1, k) \;+\; (1-\gamma_d) \cdot p(b-d, k-d) & 5 \leq d \leq \Delta, k \geq d \\
\gamma_\Delta \cdot p(b-1, k-1) \;+\; (1-\gamma_\Delta) \cdot p(b-\Delta, k) & \\
\lambda_{1,r} \cdot p(b-2, k-2) + (1-\lambda_{1,r}) \cdot p(b-r, k-2) & 3 \leq r \leq 7, k \geq 2 \\
\lambda_{1,r} \cdot p(b-2, k-1) + (1-\lambda_{1,r}) \cdot p(b-r, k-r) & 3 \leq r \leq 7, k \geq r \\
\lambda_{2,r} \cdot p(b-3, k-3) + (1-\lambda_{2,r}) \cdot p(b-r, k-1) & 3 \leq r \leq 4, k \geq 3 \\
\lambda_{2,r} \cdot p(b-3, k-1) + (1-\lambda_{2,r}) \cdot p(b-r, k-r) & 3 \leq r \leq 4, k \geq r \\
\lambda_3 \cdot p(b-3, k-3) + (1-\lambda_3) \cdot p(b-2, k) & k \geq 3 \\
\lambda_3 \cdot p(b-3, k-1) + (1-\lambda_3) \cdot p(b-2, k-2) & k \geq 2 \\
\delta_{r,1} \cdot p(b-3, k-3) + \delta_{r,2} \cdot p(b-4, k-1) + \delta_{r,3} \cdot p(b-r-1, k-3) & 5 \leq r \leq 7, k \geq 4 \\
\delta_{r,1} \cdot p(b-3, k-1) + \delta_{r,2} \cdot p(b-4, k-4) + \delta_{r,3} \cdot p(b-r-1, k-r) & 5 \leq r \leq 7, k \geq r \\
\delta_{r,1} \cdot p(b-3, k-2) + \delta_{r,2} \cdot p(b-4, k-4) + \delta_{r,3} \cdot p\left(b-r-1, k-1-\left\lceil\frac{r}{2}\right\rceil\right) & 5 \leq r \leq 7, k \geq 1+\left\lceil\frac{r}{2}\right\rceil \\
\delta_{r,1} \cdot p(b-3, k-2) + \delta_{r,2} \cdot p(b-4, k-2) + \delta_{r,3} \cdot p(b-r-1, k-r-1) & 5 \leq r \leq 7, k \geq r-1
\end{cases}
$$

(12)

Algorithm 7 BETTERVC

Input: An undirected graph $G = (V, E)$

Configuration Parameters: The algorithm uses several parameters that should be configured. These include $\Delta \in \mathbb{N}$, $\gamma_5, \gamma_6, \ldots, \gamma_\Delta \in (0, 1)$, $\lambda_{1,r} \in (0, 1)$ for every $3 \leq r \leq 7$, $\lambda_{2,r} \in (0, 1)$ for $3 \leq r \leq 4$, $\lambda_3 \in (0, 1)$ and $\delta_{r,1}, \delta_{r,2}, \delta_{r,3} \in \mathbb{R}_{\geq 0}$ with $\delta_{r,1} + \delta_{r,2} + \delta_{r,3} = 1$ for $r \in \{5, 6, 7\}$.

Notation. We use the term **branch over** $U_1, \ldots, U_r$ with probability $p_1, \ldots, p_r$ to denote the operation of returning BETTERVC$(G \setminus U_i) \cup U_i$ with probability p_i. The term **select** U denotes the operation of returning BETTERVC$(G \setminus U) \cup U$.

1: If the empty set is a cover of G return $\emptyset$.
2: If G is not connected, let G' be a connected component of G and $G'' = G - G'$. Return BETTERVC$(G') \cup$ BETTERVC(G'').
3: If G has a vertex v of degree 1, let u be its neighbor. Select u to the cover.
4: If G has a vertex v of degree $d \geq 5$ or more, let $U = N(v)$ if $d < \Delta$, and $U \subseteq N(v)$ with $|U| = \Delta$ otherwise. Branch over $\{v\}, U$ with probabilities $\gamma_d, 1 - \gamma_d$ ($\gamma_\Delta, 1 - \gamma_\Delta$ if $d \geq \Delta$).
5: If G is a regular graph, select an arbitrary edge $(v_1, v_2) \in E$. Evaluate $S_1 =$ BETTERVC$(G \setminus \{v_1\}) \cup \{v_1\}$ and $S_2 =$ BETTERVC$(G \setminus \{v_2\}) \cup \{v_2\}$. Return the smaller set between S_1 and S_2.
6: **if** G has a vertex v of degree 2, $N(v) = \{x, y\}$ **such that**:
7:      $(x, y) \in E$ **then** select $\{x, y\}$ to the cover.
8:      $\deg(x) = \deg(y) = 2$ and $N(x) = N(y) = \{z, v\}$ **then** select $\{z, v\}$ to the cover.
9:      None of the above holds. **Then** let $r = |N(x) \cup N(y)|$ and branch over $N(v), N(x) \cup N(y)$ with probabilities $\lambda_{1,r}, 1 - \lambda_{1,r}$
10: **if** G has a vertex v of degree 3, $N(v) = \{x, y, z\}$ **such that**:
11:      $(x, y) \in E$ **then** let $r = |N(z)|$ and branch over $N(v), N(z)$ with probability $\lambda_{2,r}, 1 - \lambda_{2,r}$.

12:      There is a vertex w, $w \notin N(v) \cup \{v\}$, $x, y \in N(w)$ **then** branch over $N(v), \{v, w\}$ with probabilities $\lambda_3, 1 - \lambda_3$.
13:      $\deg(x) = 4$ **then** let $r = |N(y) \cup N(z)|$ and branch over $N(v), N(x)$ and $\{x\} \cup N(y) \cup N(z)$ with probabilities $\delta_{r,1}, \delta_{r,2}, \delta_{r,3}$.

Figure 4: The performance of BetterVC. A dot at (α, c) means that the respective algorithm yields α-approximation with running time $O^*(c^k)$ or $O^*\left((c+\varepsilon)^k\right)$ for any $\varepsilon > 0$.

and $p(b, k) = 0$ for $b < 0$, and $p(b, k) = 1$ for $b \geq 0$ and $k \leq 0$.

The proofs of Lemmas 4 and 5 are given at the end of this section. The proof of Lemma 5 is a case by case analysis similar to the one done in [29]. The main difference between the analysis presented here and the analysis in [29] is that here we also count the reduction in the minimal cover size in a non-optimal branching step.

Let α-BetterVC be the algorithm which executes Algorithm 2 with Algorithm 7 as $\mathcal{A}$, and with p as the recurrence in Lemma 5. It follows from Lemma 5 that α-BetterVC is a random parameterized α-approximation algorithm for Vertex Cover, with running time $O^*\left(\frac{1}{p(\alpha k, k)}\right)$. As before, we arbitrarily select $\Delta = 100$. For every $1 < \alpha < 2$ and a set of configuration parameters, by Theorem 2 we can numerically evaluate (see Section 5 for details regarding the evaluation) a value M_α such that $p(\alpha k, k) > \exp(-M_\alpha - \varepsilon)$ for any $\varepsilon > 0$ and large enough k. Similarly, for every $1 < \alpha < 2$ we can optimize the configuration parameters so this value is minimized. Therefore, the running time of Algorithm α-BetterVC is $O^*(\exp(M_\alpha + \varepsilon)^k)$ for any $\varepsilon > 0$. Figure 4 shows $\exp(M_\alpha)$ as a function of α.

Note that the algorithm in [30] can be used along with our framework of randomized branching. However, due to its technical complexity, we preferred to use the algorithm in [29], which can be viewed as a simplified version of the same algorithm. In the discussion we describe the obstacles we encountered while attempting to obtain a randomized branching variants of faster algorithms.

4.1 Proofs

Proof of Lemma 4. To show the algorithm is polynomial, it suffices to show that the number of recursive calls is polynomial. We note that the only non-trivial part of the proof is the handling of regular graphs in Line 5. We use a simple potential function to handle this case. For $i = 2, 3, 4$, define $\Phi_i(G) = 1$ if G has a non-empty i-regular vertex induced subgraph and $\Phi_i(G) = 0$ otherwise. Also, define $\Phi(G) = \Phi_2(G) + \Phi_3(G) + \Phi_4(G)$.

We now prove by induction (on $|V|$) that the number of recursive calls initiated by the algorithm is at most $2(|V| - 1) \cdot 2^{\Phi(G)}$. If $|V| \leq 1$ the algorithm does not initiate recursive calls, and the claim holds. Each time a **Branch** or **Select** is used the size of $|V|$ decreases by at least one, $\Phi(G)$ does not increase, and only one recursive call is initiated, therefore the claim holds in these cases.

If G is not connected (Line 2) and is split into $G' = (V', E')$ and $G'' = (V'', E'')$ we note that $\Phi(G) \geq \Phi(G'), \Phi(G'')$; therefore, the number of recursive calls is bounded by

$$2 + 2(|V'| - 1) \cdot 2^{\Phi(G')} + 2(|V''| - 1) \cdot 2^{\Phi(G'')} \leq 2(|V| - 1) \cdot 2^{\Phi(G)}.$$

Finally, we need to handle the case in which G is an i-regular graph (Line 5). By the code structure, $i \in \{2, 3, 4\}$ and G is connected. In this case, two recursive calls are initiated, with $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ which are strict subgraphs of G. Since G is a connected i-regular graph, no vertex induced subgraph of G is also i-regular, thus $\Phi_i(G_1) = \Phi_i(G_2) = 0$ while $\Phi_i(G) = 1$. Thus, $\Phi(G_1), \Phi(G_2) \leq \Phi(G) - 1$. It follows that the number of recursive calls is bounded by

$$2 + 2(|V_1| - 1) \cdot 2^{\Phi(G_1)} + 2(|V_2| - 1) \cdot 2^{\Phi(G_2)}$$
$$\leq 2 + 2(|V| - 2) \cdot 2^{\Phi(G)-1} + 2(|V| - 2) \cdot 2^{\Phi(G)-1} \leq 2(|V| - 1) \cdot 2^{\Phi(G)}.$$

$\square$

Proof of Lemma 5. To prove the lemma we show by induction a slightly stronger claim. Given a collection of graphs $G_1, \ldots, G_t$, let $P(b, (G_1, \ldots, G_t))$ denote the probability that $\sum_{i=1}^{t} |\text{BETTERVC}(G_i)| \leq b$. Now, we claim that if the total size of minimal vertex covers of the graphs is at most k (formally, there are $S_1, \ldots, S_t$ where S_i is a vertex cover of G_i and $\sum_{i=1}^{t} |S_i| \leq k$) then $P(b, (G_1, \ldots, G_t)) \geq p(b, k)$. We prove the claim by induction over the lexicographical order of (b, M, ℓ), where M is the maximal number of vertices of a graph in $G_1, \ldots, G_t$, and ℓ is the number of graphs of maximal size.
Base Case 1: If $b < 0$ then clearly $P(b, (G_1, \ldots, G_t)) = 0 = p(b, k)$.
Base Case 2: For any $b \in \mathbb{N}$, if $M \leq 1$, then clearly $P(b, (G_1, \ldots, G_t)) = 1 \geq p(b, k)$.
Induction Step: Let $b \in \mathbb{N}$ and $G_1, \ldots, G_t$ with ℓ graphs of maximal size M and assume the claim holds for every (b', M', ℓ') lexicographically smaller than (b, M, ℓ). W.l.o.g assume that $G_1 = (V_1, E_1)$ and $|V_1| = M$. We consider the execution of $\text{BETTERVC}(G_1)$ and divide the analysis into cases depending on its execution path. We use two simple properties along the proof. If $\text{BETTERVC}(G_1)$ uses **branch** over $U_1, \ldots, U_r$ with probabilities $\mu_1, \ldots, \mu_r$ then

$$P(b, (G_1, \ldots, G_t)) = \sum_{j=1}^{r} \mu_j P(b - |U_j|, (G_1 \setminus U_j, G_2, \ldots, G_t))$$

And if the algorithm selects U into the cover then

$$P(b, (G_1, \ldots, G_t)) = P(b - |U|, (G_1 \setminus U, G_2, \ldots, G_t))$$

Case 1: The empty set is a cover of G_1. Therefore $|\text{BETTERVC}(G_1)| = 0$ and thus

$$\Pr\left[\sum_{i=1}^{t} |\text{BETTERVC}(G_i)| \leq b\right] = \Pr\left[\sum_{i=2}^{t} |\text{BETTERVC}(G_i)| \leq b\right] = P(b, (G_2, \ldots, G_t)) \geq p(b, k)$$

Where the last inequality follows from the induction claim, as either the maximal graph size in $G_2, \ldots, G_t$ is smaller than M, or the number of graphs of maximal size is less than ℓ.
Case 2: G_1 is not connected, then let G_1' and G_1'' be the two graphs considered in Line 2. Therefore,

$$\Pr\left[\sum_{i=1}^{t} |\text{BETTERVC}(G_i)| \leq b\right]$$
$$= \Pr\left[|\text{BETTERVC}(G_1')| + |\text{BETTERVC}(G_1'')| + \sum_{i=2}^{t} |\text{BETTERVC}(G_i)| \leq b\right]$$
$$= P(b, (G_1', G_1'', G_2, \ldots, G_t)) \geq p(b, k)$$

Note that since the number of vertices in both G_1' and G_1'' is strictly smaller than M the induction claim holds for b and $(G_1', G_1'', G_2, \ldots, G_t)$ from which the last inequality follows.

Case 3: The selection in Line 3 is executed. Then, G_1 has a vertex v of degree 1, and $N(v) = \{u\}$, and u is selected into the cover. Clearly, if G_1 has a vertex cover of size k_1 then $G_1 \setminus \{u\}$ has a vertex cover of size $k_1 - 1$. Therefore,

$$P(b, (G_1, \ldots, G_t)) = P(b - 1, (G_1 \setminus \{u\}, G_2, \ldots, G_t)) \geq p(b - 1, k - 1) \geq p(b, k)$$

the first inequality holds by the induction claim, the second inequality follows from (12).

Case 4: The algorithm uses the branching in Line 4. Let S_1 be a minimal cover of G_1. If $v \in S_1$, then $S_1 \setminus \{v\}$ is a vertex cover of $G_1 \setminus \{v\}$. Also, if $d < \Delta$, then $S_1 \setminus \{v\}$ is also a vertex cover of $G_1 \setminus U$. Therefore,

$$
\begin{aligned}
&P(b, (G_1, \ldots, G_t)) \\
=&\gamma_d \cdot P(b - 1, (G_1 \setminus \{u\}, G_2, \ldots, G_t)) + (1 - \gamma_d) \cdot P(b - d, (G_1 \setminus U, G_2, \ldots, G_t)) \\
\geq&\gamma_d \cdot p(b - 1, k - 1) + (1 - \gamma_d) \cdot p\left(b - d, k - \begin{cases} 1 & \text{if } d < \Delta \\ 0 & \text{if } d = \Delta \end{cases}\right) \geq p(b, k)
\end{aligned}
$$

The first inequality follows from the induction claim, the second is due to (12).

Otherwise, if $v \notin S_1$, then $U \subseteq S$. Clearly, $S_1 \setminus U$ is a vertex cover of $G_1 \setminus U$. Thus we get,

$$
\begin{aligned}
&P(b, (G_1, \ldots, G_t)) \\
=&\gamma_d \cdot P(b - 1, (G_1 \setminus \{u\}, G_2, \ldots, G_t)) + (1 - \gamma_d) \cdot P(b - d, (G_1 \setminus U, G_2, \ldots, G_t)) \\
\geq&\gamma_d \cdot p(b - 1, k) + (1 - \gamma_d) \cdot p(b - d, k - d) \geq p(b, k)
\end{aligned}
$$

As before, the first inequality is by the induction claim, and the second is due to (12).

As the claim holds whether $v \in S_1$ or $v \notin S_1$ we get that the induction claim hold for this case.

Case 5: Line 5 takes place. Let S_1 be a minimal vertex cover of G_1. As S_1 is a cover we have $v_1 \in S_1$ or $v_2 \in S_1$. W.l.o.g we can assume $v_1 \in S_1$. Clearly, $S_1 \setminus \{v_1\}$ is a cover of $G_1 \setminus \{v_1\}$. Now,

$$
\begin{aligned}
&\Pr\left[\sum_{i=1}^{t} |\text{BETTERVC}(G_i)| \leq b\right] \\
=&\Pr\left[1 + \min_{j=1,2} |\text{BETTERVC}(G_1 \setminus \{v_j\})| + \sum_{i=2}^{t} |\text{BETTERVC}(G_i)| \leq b\right] \\
\geq&\Pr\left[1 + |\text{BETTERVC}(G_1 \setminus \{v_1\})| + \sum_{i=2}^{t} |\text{BETTERVC}(G_i)| \leq b\right] \\
=&P(b - 1, (G_1 \setminus \{v_1\}, G_2, \ldots, G_t)) \geq p(b - 1, k - 1)
\end{aligned}
$$

The first inequality is since the event set in the third term is a subset of the event set of the second term. The second inequality follows from the induction claim, and the last inequality is due to (12).

Case 6: The algorithm executes Line 7. Let S_1 be an minimal vertex cover of G_1. Note that $|S_1 \cap \{x, y, v\}| \geq 2$ and $S_1 \setminus \{x, y, v\}$ is a vertex cover of $G_1 \setminus \{x, y\}$. Therefore,

$$P(b, (G_1, \ldots, G_t)) = P(b - 2, (G_1 \setminus \{x, y\}, G_2, \ldots, G_t)) \geq p(b - 2, k - 2) \geq p(b, k)$$

The first inequality follows from the induction claim, the second from (12).

Case 7: Line 8 is executed. Let S_1 be a minimal vertex cover of G_1. Clearly, $|S_1 \cap \{v, x, y, z\}| \geq 2$ and $S_1 \setminus \{x, a, b, d\}$ is a vertex cover of $G_1 \setminus \{z, v\}$. Therefore, as in the previous case,

$$P(b, (G_1, \ldots, G_t)) = P(b - 2, (G_1 \setminus \{z, v\}, G_2, \ldots, G_t)) \geq p(b - 2, k - 2) \geq p(b, k)$$

Case 8: Line 9 is executed. Since the conditions are not met for Lines 7 and 8 then $(x, y) \notin E_1$ and $|N(x) \cup N(y)| \geq 3$. As the graph does not have vertices of degree 5 or more, we also have $\deg(x), \deg(y) \leq 4$. We can now conclude $3 \leq r \leq 7$ (recall that $r = |N(x) \cup N(y)|$).

If there is a minimal vertex cover S_1 of G_1 such that $v \notin S_1$, then $x, y \in S_1$. Clearly, $S_1 \setminus \{x, y\}$ is a vertex cover of $G_1 \setminus \{x, y\} = G_1 \setminus N(v)$. Also, it is easy to see that $S_1 \setminus \{x, y\}$ is also a vertex cover of $G_1 \setminus (N(x) \cup N(y))$ (we remove vertices which do not belong to the graph). Therefore,

$$
\begin{aligned}
&P(b, (G_1, \ldots, G_t)) \\
=&\lambda_{1,r} \cdot P(b - 2, (G_1 \setminus N(v), G_2, \ldots, G_t)) + (1 - \lambda_{1,r}) \cdot P(b - r, (G_1 \setminus (N(x) \cup N(y)), G_2, \ldots, G_t)) \\
\geq&\lambda_{1,r} \cdot p(b - 2, k - 2) + (1 - \lambda_{1,r}) \cdot p(b - r, k - 2) \geq p(b, k)
\end{aligned}
$$

The first inequality follows from the induction claim. The second one is due to (12).

Otherwise, every minimal vertex cover of G_1 includes v. Let S_1 be a vertex cover of G_1. Clearly, $v \in S_1$. We note that $x, y \notin S_1$, since otherwise $S_1 \setminus \{v\} \cup \{x, y\}$ is a vertex cover of G_1 of the same size as S_1, in contradiction to our case. Therefore, $N(x) \cup N(y) \subseteq S_1$. Obviously $S_1 \setminus (N(x) \cup N(y))$ is a vertex cover of $G_1 \setminus (N(x) \cup N(y))$. We also note that $S_1 \setminus \{v\}$ is a cover of $G_1 \setminus N(v)$. Therefore,

$$
\begin{aligned}
&P(b, (G_1, \ldots, G_t)) \\
=&\lambda_{1,r} \cdot P(b - 2, (G_1 \setminus N(v), G_2, \ldots, G_t)) + (1 - \lambda_{1,r}) \cdot P(b - r, (G_1 \setminus (N(x) \cup N(y)), G_2, \ldots, G_t)) \\
\geq&\lambda_{1,r} \cdot p(b - 2, k - 1) + (1 - \lambda_{1,r}) \cdot p(b - r, k - r) \geq p(b, k)
\end{aligned}
$$

The first inequality follows from the induction claim. The second one is due to (12).

Case 9: Line 11 is executed. Since this line of code has been reached, then G_1 has only vertices of degree 3 and 4. Therefore $r = |N(z)| \in \{3, 4\}$.

If there is a minimal vertex cover S_1 of G_1 such that $v \notin S_1$, then $N(v) \subseteq S_1$, and $S_1 \setminus N(v)$ is a vertex cover of $G \setminus N(v)$. Also, it is easy to see that $S_1 \setminus \{z\}$ is a vertex cover of $G \setminus N(z)$ (after removing vertices which no longer belong to the graph). Therefore,

$$
\begin{aligned}
&P(b, (G_1, \ldots, G_t)) \\
=&\lambda_{2,r} \cdot P(b - 3, (G_1 \setminus N(v), G_2, \ldots, G_t)) + (1 - \lambda_{2,r}) \cdot P(b - r, (G_1 \setminus (N(z)), G_2, \ldots, G_t)) \\
\geq&\lambda_{2,r} \cdot p(b - 3, k - 3) + (1 - \lambda_{2,r}) \cdot p(b - r, k - 1) \geq p(b, k)
\end{aligned}
$$

The first inequality follows from the induction claim. The second one is due to (12).

Otherwise, every minimal vertex cover S_1 of G_1 has v in it. Let S_1 be a minimal vertex cover of G_1. Clearly, $v \in S_1$. If $|S_1 \cap \{x, y, z\}| \geq 2$ then $S_1 \cup \{x, y, z\} \setminus \{v\}$ is a vertex cover of G_1 of the same size, contradicting our assumption. Therefore, $|S_1 \cap \{x, y, z\}| \leq 1$. Since $x \in S_1$ or $y \in S_1$ (since $(x, y) \in E_1$) we have $z \notin S_1$, and $N(z) \subseteq S_1$. Also, note that $S_1 \setminus \{v\}$ is a cover of $G \setminus N(v)$ and $|S_1 \setminus \{v\}| \leq |S_1| - 1$. Therefore,

$$
\begin{aligned}
&P(b, (G_1, \ldots, G_t)) \\
=&\lambda_{2,r} \cdot P(b - 3, (G_1 \setminus N(v), G_2, \ldots, G_t)) + (1 - \lambda_{2,r}) \cdot P(b - r, (G_1 \setminus (N(z)), G_2, \ldots, G_t)) \\
\geq&\lambda_{2,r} \cdot p(b - 3, k - 1) + (1 - \lambda_{2,r}) \cdot p(b - r, k - r) \geq p(b, k)
\end{aligned}
$$

The first inequality follows from the induction claim. The second one is due to (12).

Case 10: Line 12 is executed.

If there is a minimal vertex cover S_1 such that $v \notin S_1$, then,

$$P(b, (G_1, \ldots, G_t))$$
$$= \lambda_3 \cdot P(b-3, (G_1 \setminus N(v), G_2, \ldots, G_t)) + (1 - \lambda_3) \cdot P(b-2, (G_1 \setminus \{v, w\}, G_2, \ldots, G_t))$$
$$\geq \lambda_3 \cdot p(b-3, k-3) + (1 - \lambda_3) \cdot p(b-2, k) \geq p(b, k)$$

The first inequality follows from the induction claim. The second one is due to (12).

Otherwise, every minimal vertex cover S_1 has v in it. Let S_1 be a minimal vertex cover of G_1. If $|S_1 \cap \{x, y, z\}| \geq 2$ we get get a contradiction to the assumption by removing v from S_1 and adding a vertex from x, y, z into it. Therefore $|S_1 \cap \{x, y, z\}| \leq 1$ and surely $w \in S_1$ (if $w \notin S_1$ then $x, y \in S_1$). We also note that $S_1 \setminus \{v\}$ is a vertex cover of $G \setminus N(v)$. Therefore,

$$P(b, (G_1, \ldots, G_t))$$
$$= \lambda_3 \cdot P(b-3, (G_1 \setminus N(v), G_2, \ldots, G_t)) + (1 - \lambda_3) \cdot P(b-2, (G_1 \setminus \{v, w\}, G_2, \ldots, G_t))$$
$$\geq \lambda_3 \cdot p(b-3, k-1) + (1 - \lambda_3) \cdot p(b-2, k-2) \geq p(b, k)$$

The first inequality follows from the induction claim. The second one is due to (12).

Case 11: Line 13 is executed. Since there are no edges between x, y, z and the vertices has no common neighbor beside v we have $r \in \{5, 6, 7\}$. We will further divide into sub-cases.

1. If there is a minimal vertex cover S_1 of G_1 such that $v \notin S_1$, then $N(v) \in S_1$. Clearly, $S_1 \setminus N(v)$ is a vertex cover of $G_1 \setminus N(v)$. Also, $S_1 \setminus \{x\}$ is a vertex cover of $G_1 \setminus N(x)$, and $S_1 \setminus N(v)$ is a vertex cover of $G \setminus (\{x\} \cup N(y) \cup N(z))$. Therefore

$$P(b, (G_1, \ldots, G_t))$$
$$= \delta_{r,1} \cdot P(b-3, (G_1 \setminus N(v), G_2, \ldots, G_t)) +$$
$$\delta_{r,2} \cdot P(b-4, (G_1 \setminus N(x), G_2, \ldots, G_t)) +$$
$$\delta_{r,3} \cdot P(b-r-1, (G_1 \setminus (\{x\} \cup N(y) \cup N(z)), G_2, \ldots, G_t))$$
$$\geq \delta_{r,1} \cdot p(b-3, k-3) + \delta_{r,2} \cdot p(b-4, k-1) + \delta_{r,3} \cdot p(b-r-1, k-3) \geq p(b, k)$$

Therefore, we may assume that v is in every minimal cover.

2. If there is a minimal cover S_1 of G_1 such that $x, y, z \notin S_1$. Then $N(x), N(y), N(z) \subseteq S_1$. Now, $S_1 \setminus N(x)$ is a vertex cover of $G_1 \setminus N(x)$, $S_1 \setminus \{v\}$ is a vertex cover of $G_1 \setminus N(v)$ and $S_1 \setminus (N(y) \cup N(z))$ is a vertex cover of $G_1 \setminus (\{x\} \cup N(y) \cup N(z))$. Therefore,

$$P(b, (G_1, \ldots, G_t))$$
$$= \delta_{r,1} \cdot P(b-3, (G_1 \setminus N(v), G_2, \ldots, G_t)) +$$
$$\delta_{r,2} \cdot P(b-4, (G_1 \setminus N(x), G_2, \ldots, G_t)) +$$
$$\delta_{r,3} \cdot P(b-r-1, (G_1 \setminus (\{x\} \cup N(y) \cup N(z)), G_2, \ldots, G_t))$$
$$\geq \delta_{r,1} \cdot p(b-3, k-1) + \delta_{r,2} \cdot p(b-4, k-4) + \delta_{r,3} \cdot p(b-r-1, k-r) \geq p(b, k)$$

3. If there is a minimal cover S_1 of G_1 such that $x \notin S_1$, but one of y, z is in S_1, w.l.o.g $y \in S_1$. Therefore $N(x), N(z) \subseteq S_1$, and we can tell that $S_1 \setminus N(x)$ is a vertex cover of $G_1 \setminus N(x)$, $S_1 \setminus \{v, y\}$ is a vertex cover of $G_1 \setminus N(v)$ and $S_1 \setminus N(z) \setminus \{y\}$ is a vertex cover of $G_1 \setminus (\{x\} \cup N(y) \cup N(z))$. Note that $N(z) \geq \lceil \frac{r}{2} \rceil$. Therefore

$$P(b, (G_1, \ldots, G_t))$$
$$= \delta_{r,1} \cdot P(b-3, (G_1 \setminus N(v), G_2, \ldots, G_t)) +$$
$$\delta_{r,2} \cdot P(b-4, (G_1 \setminus N(x), G_2, \ldots, G_t)) +$$
$$\delta_{r,3} \cdot P(b-r-1, (G_1 \setminus (\{x\} \cup N(y) \cup N(z)), G_2, \ldots, G_t))$$
$$\geq \delta_{r,1} \cdot p(b-3, k-2) + \delta_{r,2} \cdot p(b-4, k-4) + \delta_{r,3} \cdot p\left(b-r-1, k-1-\left\lceil \frac{r}{2} \right\rceil\right) \geq p(b, k)$$

22

4. If there is a minimal cover S_1 such that $x \notin S_1$ and $y, z \in S_1$, then $S_1 \cup \{x\} \setminus \{v\}$ is a minimal cover without v, and therefore the claim holds due to sub-case 1.

5. There is a minimal vertex cover S_1 such that $x, v \in S_1$. If $y \in S_1$ or $z \in S_1$, w.l.o.g $y \in S_1$, then $S_1 \cup \{z\} \setminus \{v\}$ is a minimal vertex cove of G_1 which does not include v. As this situation is already handled in sub-case 1, we can assume $y, z \notin S_1$ and therefore $N(y), N(z) \subseteq S_1$. Now, note that $S_1 \setminus \{x, v\}$ is a vertex cover of both $G_1 \setminus N(v)$ and $G_1 \setminus N(x)$. Therefore,

$$
\begin{aligned}
& P(b, (G_1, \ldots, G_t)) \\
={} & \delta_{r,1} \cdot P(b - 3, (G_1 \setminus N(v), G_2, \ldots, G_t)) + \\
& \delta_{r,2} \cdot P(b - 4, (G_1 \setminus N(x), G_2, \ldots, G_t)) + \\
& \delta_{r,3} \cdot P(b - r - 1, (G_1 \setminus (\{x\} \cup N(y) \cup N(z)), G_2, \ldots, G_t)) \\
\geq{} & \delta_{r,1} \cdot p(b - 3, k - 2) + \delta_{r,2} \cdot p(b - 4, k - 2) + \delta_{r,3} \cdot p(b - r - 1, k - r - 1) \geq p(b, k)
\end{aligned}
$$

$\square$

5   Numerical Methods

While our main contributions are theoretical, optimizing the parameter values and evaluating the running times of our algorithms required some numerical analysis. In this section we overview the methods and tools used for obtaining the numerical results.

Each of our algorithms consists of $R \in \mathbb{N}_+$ branching rules, where rule ℓ, $1 \leq \ell \leq R$, has r_ℓ branching *options* and h_ℓ branching *states* (mostly $r_\ell = h_\ell$, with the exception of Algorithm BETTERVC of Section 4). For each rule, the algorithm uses a distribution $\bar{\gamma}^\ell \in \mathbb{R}^{r_\ell}$ to randomly select a branching option. The vector $\bar{b}^\ell \in \mathbb{N}_+^{r_\ell}$ is the budget decrease incurred by selecting each option. Each rule is also associated with h_ℓ vectors $\bar{k}^{\ell,1}, \ldots, \bar{k}^{\ell,h_\ell} \in \mathbb{N}^{r_\ell}$ where the value $\bar{k}_i^{\ell,j}$ is the decrease in the parameter (coverage) when selecting the i-th option of rule ℓ while in state j. Using the above notation, the composite recurrence used to lower bound the success probability of the algorithm is the function $p_{\bar{\gamma}^1, \ldots, \bar{\gamma}^R}(b, k)$ defined by

$$
p_{\bar{\gamma}^1, \ldots, \bar{\gamma}^R}(b, k) = \min_{1 \leq \ell \leq R} \quad \min_{1 \leq j \leq h_\ell \text{ s.t. } \bar{k}^{\ell,j} \leq k} \quad \sum_{i=1}^{r_j} \bar{\gamma}_i^\ell \cdot p_{\bar{\gamma}^1, \ldots, \bar{\gamma}^R}(b - \bar{b}_i^\ell, k - \bar{k}_i^{\ell,j}), \tag{13}
$$

with the same initial conditions as in (2).

Consider, for example, VC3 (Algorithm 1). In this algorithm we have $R = 1$, $r_1 = h_1 = 2$; the algorithm has a single rule with two branching options: selecting the vertex v or its neighbors. The vectors $\bar{b}^1 = (1, 3)$ represents the budget decrease for each option. The vector $\bar{k}^{1,1} = (1, 0)$ indicates the decrease in the minimal cover size in the *state*: "v is in an optimal cover". Similarly, $\bar{k}^{1,2} = (0, 3)$ is the coverage decrease in the state: "v is not not in an optimal cover". Indeed, by using the above values in (13), we obtain the same recurrence as in (8).

For each of our algorithms and a given approximation ratio α, to obtain an optimal running time, we seek distributions $\bar{\gamma}^1, \ldots, \bar{\gamma}^R$ that maximize $\lim_{k \to \infty} \frac{1}{k} \log p_{\bar{\gamma}^1, \ldots, \bar{\gamma}^R}(\alpha k, k)$. It follows from Theorem 2 that

$$\max\left\{\lim_{k\to\infty}\frac{1}{k}\log p_{\bar{\gamma}^1,\ldots,\bar{\gamma}^R}(\alpha k,k)\,\middle|\,\begin{array}{c}\forall 1\leq\ell\leq R:\bar{\gamma}^\ell\in\mathbb{R}^{r_\ell}\\ \bar{\gamma}^1,\ldots,\bar{\gamma}^R\text{ are distributions}\end{array}\right\}\tag{14}$$

$$=\max\left\{-\max_{1\leq\ell\leq R}\max_{1\leq j\leq h_\ell}M^{\ell,j}\,\middle|\,\begin{array}{c}\forall 1\leq\ell\leq R:\bar{\gamma}^\ell\in\mathbb{R}^{r_\ell}\\ \bar{\gamma}^1,\ldots,\bar{\gamma}^R\text{ are distributions}\\ M^{\ell,j}\text{ is the }\alpha\text{-branching number of }(\bar{b}^\ell,\bar{k}^{\ell,j},\bar{\gamma}^\ell)\end{array}\right\}$$

$$=-\max_{1\leq\ell\leq R}\min\left\{\max_{1\leq j\leq h_\ell}\frac{D\left(\bar{q}^j\|\bar{\gamma}\right)}{\bar{k}^{\ell,j}\cdot\bar{q}^j}\,\middle|\,\begin{array}{c}\bar{\gamma},\bar{q}^1,\ldots,\bar{q}^{h_\ell}\in\mathbb{R}^{r_\ell}\text{ and are all distributions}\\ \forall 1\leq j\leq h_\ell:\alpha\bar{q}^j\cdot\bar{k}^{\ell,j}\geq\bar{q}^j\cdot\bar{b}^\ell\end{array}\right\}\tag{15}$$

Define a *rule opimization problem* as follows. The input is $\alpha\in\mathbb{R}_+$, $r,h\in\mathbb{N}$, $\bar{b}\in\mathbb{N}_+^r$ and h vectors $\bar{k}^1,\ldots,\bar{k}^h\in\mathbb{N}^r$. The objective is to find distributions $\bar{\gamma},\bar{q}^1,\ldots,\bar{q}^h\in\mathbb{R}_{\geq 0}^r$ such that, for any $1\leq j\leq h$, it holds that $\alpha\bar{q}^h\cdot\bar{k}^h\geq\bar{q}^h\cdot\bar{b}$, and $\max_{1\leq j\leq h}\frac{D(\bar{q}^j\|\bar{\gamma})}{\bar{q}^j\cdot\bar{k}^j}$ is minimized. Thus, the problem of optimizing the parameters $\bar{\gamma}^1,\ldots,\bar{\gamma}^R$ of a given algorithm, as given in (14), can be reduced to R separate rule optimization problems as in (15).

In the following we show how these rules optimization problems were solved. We first show that each of these problems is quasiconvex and discuss the methods used to solve the problems as such. We then consider a common special case which has a nearly closed form solution.

5.1 Quasiconvex Programming

A function $f:C\to R$ is *quasiconvex* if C is convex and, for any $\beta\in\mathbb{R}$, the *level-set* $\{x\in C|f(x)\leq\beta\}$ is convex. A *quasiconvex program* is the problem of finding the minimum of a quasiconvex function f over a convex set D (that is, $\min_{x\in D}f(x)$). Quasiconvex programming was first defined by Amenta et. al. [3], and was already used in the context of multivariate recurrences in [18].

We now show that the rule optimization problem is a quasicovex programming. It is well known that Kullback-Leibler divergence is convex (Theorem 2.7.2 cf. [13]); therefore, by Theorem 1 in [1], the functions $f_j(\bar{\gamma},\bar{q}^1,\ldots,\bar{q}^h)=\frac{D(q^j\|\gamma)}{k^j\cdot\bar{q}^j}$, $\forall 1\leq j\leq h$, are quasiconvex. Thus, $f(\bar{\gamma},\bar{q}^1,\ldots,\bar{q}^h)=\max_{1\leq j\leq h}f_j(\bar{\gamma},\bar{q}^1,\ldots,\bar{q}^h)$ is also quasiconvex. Furthermore, the constraints over $\bar{\gamma},\bar{q}^1,\ldots,\bar{q}^h$ defining the rule optimization problem are all linear; thus, the feasible region is convex.

We used the disciplined quasiconvex programming module of CVXPY [1], an open source python optimization package, to solve the rule optimization problems which did not fall into the category of simple rules (see Section 5.2). Specifically, the results for 3-Hitting Set (Section 3) were evaluated using this method. We encountered numerical accuracy issues when using CVXPY. In such cases, the returned solution was modified to make it a feasible solution. While such changes may harm the optimality of the solution, they can only increase the running times of our algorithms.

5.2 Simple Branching Rules

Many of the branching rules used for Vertex Cover have a specific structure that we call *simple*. We say a rule optimization problem is simple if $\alpha>1$, $r=h=2$, $\bar{b}=(b_1,b_2)$, $\bar{k}^1=(b_1,s_2)$ and $\bar{k}^2=(s_1,b_2)$ for $s_1<b_1$ and $s_2<b_2$. The single rule of VC3 is simple, and so are all the rules in the algorithms of Section 2. Also, many of the rules of BETTERVC (in Section 4) are simple.

As the case is two-dimensional, we use the notation $\bar{\gamma}=(\gamma,1-\gamma)$, $\bar{q}^1=(q^{(1)},1-q^{(1)})$ and $\bar{q}^2=(1-q^{(2)},q^{(2)})$. Let

$$C_1=\left\{q^{(1)}\in[0,1]\,\middle|\,\alpha\cdot(q^{(1)}b_1+(1-q^{(1)})s_2)\geq q^{(1)}b_1+(1-q^{(1)})b_2\right\}$$

$$C_2=\left\{q^{(2)}\in[0,1]\,\middle|\,\alpha\cdot(q^{(2)}b_2+(1-q^{(2)})s_1)\geq q^{(2)}b_2+(1-q^{(2)})b_1\right\}$$

and

$$f_1(\gamma) = \min_{q^{(1)} \in C_1} \frac{D\left(q^{(1)} \middle\| \gamma\right)}{q^{(1)}b_1 + (1 - q^{(1)})s_2} \quad , \quad f_2(\gamma) = \min_{q^{(2)} \in C_2} \frac{D\left(q^{(2)} \middle\| 1 - \gamma\right)}{q^{(2)}b_2 + (1 - q^{(2)})s_1}.$$

We use the common notation $D\left(x\middle\|y\right) = D\left((x, 1-x)\middle\|(y, 1-y)\right)$ when x and y are numbers. Thus, the rule optimization problem is

$$\min_{\gamma \in [0,1]} \max\left\{f_1(\gamma), f_2(\gamma)\right\}. \tag{16}$$

In the following we show that f_1 is monotonically decreasing, f_2 is monotonically increasing, and both can be evaluated exactly by a closed formula. Thus, the solution of (16) is at the point γ where $f_1(\gamma) = f_2(\gamma)$, which can be found using a binary search over the monotonic function $f_1(\gamma) - f_2(\gamma)$.

It can be easily observed that $C_1 = [c_1, 1]$ for c_1 that can be calculated exactly. For any $\gamma \geq c_1$ we have that $q^{(1)} = \gamma$ is a solution for the optimization problem of f_1, and therefore $f_1(\gamma) = 0$. Consider the case where $\gamma < c_1$. As shown in Section 5.1, the function $h_\gamma(q^{(1)}) = \frac{D\left(q^{(1)}\middle\|\gamma\right)}{q^{(1)}b_1 + (1-q^{(1)})s_2}$ is quasiconvex. Furthermore, h has a minimum at $q^{(1)} = \gamma$. Thus, the function is increasing in $[\gamma, 1] \supseteq C_1 = [c_1, 1]$. We get that

$$f_1(\gamma) = \min_{q^{(1)} \in C_1} \frac{D\left(q^{(1)}\middle\|\gamma\right)}{q^{(1)}b_1 + (1 - q^{(1)})s_2} = \min_{q^{(1)} \in C_1} h_\gamma(q^{(1)}) = h_\gamma(c_1) = \frac{D\left(c_1\middle\|\gamma\right)}{c_1 b_1 + (1 - c_1)s_2}.$$

A symmetric closed formula can be obtained for f_2.

It follows from the above that $f_1(\gamma) = 0$ for $\gamma \geq c_1$. For any fixed $q^{(1)} \in C_1$ the function $g_{q^{(1)}}(\gamma) = \frac{D\left(q^{(1)}\middle\|\gamma\right)}{q^{(1)}b_1 + (1-q^{(1)})s_2}$ is convex with a minimum at $g_{q^{(1)}}(q^{(1)}) = 0$, and therefore decreasing in $[0, q^{(1)}] \supseteq [0, c_1]$. Hence, for any $\gamma_1 < \gamma_2 \leq c_1$, we have

$$f_1(\gamma_1) = \min_{q^{(1)} \in C_1} g_{q^{(1)}}(\gamma_1) \geq \min_{q^{(1)} \in C_1} g_{q^{(1)}}(\gamma_2) = f(\gamma_2) \geq 0.$$

Thus, f_1 is decreasing. A symmetric argument can be used to show that f_2 is increasing. This implies that a binary search can be used to solve the rule optimization problem.

We note that some of the calculations outlined in this section could be replaced by a probabilistic interpretation of the rule and its states as negative binomial distributions.

6 Proof of Theorem 2

In this section we give a formal proof of Theorem 2. We first define the stochastic process used in the proof, and show its connection to the recurrence relation p in (2). While the definition of the process is abstract, it is driven from an intuitive interpretation of the algorithms presented in this paper, as demonstrated in Section 6.1.

6.1 Recurrence as a Random Process: an Example

Recall that in each recursive call of VC3 (Algorithm 1) either a vertex v or three of its neighbors are added to the cover. Given a graph G and a minimum size vertex cover S of G, the execution of VC3 can be associated with the following stochastic process $\tilde{X}_n$.

Let v_n be the vertex considered by the algorithm in the n-th recursive call (while the algorithm is finite, we assume it is infinite for this intuitive interpretation). In case $v_n \in S$, set $\tilde{X}_n = 1$ if v_n is selected by the algorithm, and $\tilde{X}_n = 2$ if the neighbors of v_n are selected. Similarly, in case $v_n \notin S$, set $\tilde{X}_n = 3$ if v_n is selected by the algorithm, and $\tilde{X}_n = 4$ otherwise.

We note that the algorithm "does not know" the values of $\tilde{X}_n$; nevertheless, we can use these values for the analysis.

Given $b \in \mathbb{Z}$ and $k \in \mathbb{N}$, we want to compute the probability that VC3 selects k vertices from the minimal cover S before it adds b vertices to the cover. This is equivalent to the probability that there is $n \in \mathbb{N}$ such that $\sum_{\ell=1}^n K_{\tilde{X}_\ell} \geq k$ and $\sum_{\ell=1}^n B_{\tilde{X}_\ell} \leq b$, where $B = (1,3,1,3)$ and $K = (1,0,0,3)$.

We note that $\tilde{X}_n$ is either drawn from $\mathcal{X}_1 = \{1,2\}$ or from $\mathcal{X}_2 = \{3,4\}$. The set from which $\tilde{X}_n$ is drawn depends on whether v_n is in S. The latter depends on the input graph G and the selections of VC3, $\tilde{X}_1, \ldots, \tilde{X}_{n-1}$ (we assume a specific minimal size vertex cover S is arbitrarily associated with every graph G).

Therefore, we can associate the graph G with a function $R : (\mathcal{X}_1 \cup \mathcal{X}_2)^* \to \{1,2\}$. For any $(a_1, \ldots, a_{n-1}) \in \mathcal{X}^*$ set $R(a_1, \ldots, a_{n-1}) = 1$ if $v_n \in S$ given $\tilde{X}_1 = a_1, \ldots, \tilde{X}_{n-1} = a_{n-1}$, and $R(a_1, \ldots, a_{n-1}) = 2$ otherwise (these includes also infeasible execution paths for the algorithm). That is, $\tilde{X}_n$ is drawn from $\mathcal{X}_j$, where $j = R(\tilde{X}_1, \ldots, \tilde{X}_{n-1})$.

Given R, we now define a stochastic process $(X_n)_{n=1}^\infty$ which has the same distribution as $(\tilde{X}_n)_{n=1}^\infty$. Let $(Y_n^1)_{n=1}^\infty$ and $(Y_n^2)_{n=1}^\infty$ be a two sequences of $i.i.d.$ random variables such that $\Pr(Y_n^1 = 1) = \gamma$, $\Pr(Y_n^1 = 2) = 1 - \gamma$, $\Pr(Y_n^2 = 3) = \gamma$ and $\Pr(Y_n^2 = 4) = 1 - \gamma$ ($\gamma \in (0,1)$ is the probability VC3 selects v). Now, set $X_n = Y_n^1$ if $R(X_1, \ldots, X_{n-1}) = 1$ and $X_n = Y_n^2$ otherwise. It can be easily verified that indeed $(\tilde{X}_n)_{n=1}^\infty$ and $(X_n)_{n=1}^\infty$ have the same distribution.

As we do not know R, our objective is to lower bound the probability there is $n \in \mathbb{N}$ such that $\sum_{\ell=1}^n K_{X_\ell} \geq k$ and $\sum_{\ell=1}^n B_{X_\ell} \leq b$, over a large set of functions R, which includes all the possible functions derived from graphs.

6.2 Proof the Theorem

Let $\alpha > 0$, $N \in \mathbb{N}_+$ and $r_j \in \mathbb{N}_+$ for $1 \leq j \leq N$. Also, let $\bar{b}^j \in \mathbb{N}_+^{r_j}$, $\bar{k}^j \in \mathbb{N}^{r_j}$ and $\bar{\gamma}^j \in \mathbb{R}_+^{r_j}$ for $1 \leq j \leq N$. We assume that $\bar{\gamma}^j$ is a distribution for any j, and there is $1 \leq j \leq N$ such that $\bar{k}^j \leq 1$. Let p be the composite recurrence of $\{(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j) | 1 \leq j \leq N\}$ as defined in (2). Finally, let M_j be the α-branching number of $(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)$ and $M = \max_{1 \leq j \leq N} M_j$. We assume that $M < \infty$.

We start with a few technical definitions. Let B (K, Γ) be the result of concatenating the vectors $\bar{b}^1, \bar{b}^2, \ldots, \bar{b}^N$ ($\bar{k}^1, \bar{k}^2, \ldots, \bar{k}^N$ and $\bar{\gamma}^1, \bar{\gamma}^2, \ldots, \bar{\gamma}^N$, respectively). Formally, set $s_j = \sum_{k=1}^j r_k$ (therefore, $s_0 = 0$) and $r = s_N$. For any $1 \leq j \leq N$ and $1 \leq i \leq r_j$ set $B_{s_{j-1}+i} = \bar{b}_i^j$, $K_{s_{j-1}+i} = \bar{k}_i^j$ and $\Gamma_{s_{j-1}+i} = \bar{\gamma}_i^j$. Also, define $\mathcal{X}_j = \{i \in \mathbb{N} | s_{j-1} < i \leq s_j\}$ and $\mathcal{X} = \bigcup_{j=1}^N \mathcal{X}_j = \{1, 2, \ldots, r\}$.

We say that $\Upsilon \in \mathbb{R}_{\geq 0}^r$ is an **extended distribution** if for any $1 \leq j \leq N$ it holds that $\sum_{i \in \mathcal{X}_j} \Upsilon_i = 1$; for example, Γ is an extended distribution. A **rules mapping** is a function[8] $R : \mathcal{X}^* \to \{1, 2, \ldots, N\}$.

Given an extended distribution Υ and a rules mapping R, we define a stochastic process $(X_n)_{n=1}^\infty$ where $X_n \in \mathcal{X}$ for any $n \in \mathbb{N}$. For $1 \leq j \leq N$, let $(Y_n^j)_{n=1}^\infty$ be $i.i.d.$ random variables where $\Pr(Y_n^j = i) = \Upsilon_i$ for $i \in \mathcal{X}_j$. We set $X_n = Y_n^j$ where $j = R(X_1, \ldots, X_{n-1})$. We use $\Pr_{R,\Upsilon}$ to denote the probability distribution function (formally, this is a probability measure) of the process defined by R and Υ. In particular, we use this distribution function with respect to two specific extended distributions: Γ and Q. We use Υ for showing generic results which apply to both Γ and Q.

Define $\kappa(a_1, \ldots, a_n) = \sum_{\ell=1}^n K_{a_\ell}$ and $\mu(a_1, \ldots, a_n) = \sum_{\ell=1}^n B_{a_\ell}$ to be the coverage and cost of $(a_1, \ldots, a_n) \in \mathcal{X}^n$.

---

[8] For a set A we use A^* to denote all vectors of elements in A. That is, $A = \bigcup_{n=0}^\infty A^n$. We use ϵ to denote the vector of dimension 0 ($\epsilon \in A^0$).

For $b, k \in \mathbb{R}_{\geq 0}$ define the event[9]

$$S^{b,k} = \{\exists n \in \mathbb{N} : \mu(X_1, \ldots, X_n) \leq b \text{ and } \kappa(X_1, \ldots, X_n) \geq k\}. \tag{17}$$

Finally, we say that a rules mapping R is k-**consistent** for $k \in \mathbb{N}$ if for any $\bar{a} \in \mathcal{X}^*$ such that $\kappa(\bar{a}) < k$ it holds that $K_i \leq k - \kappa(\bar{a})$ for any $i \in \mathcal{X}_{R(\bar{a})}$ (equivalently, $\bar{k}^{R(\bar{a})} \leq k - \kappa(\bar{a})$). The notion of k-consistent mapping corresponds to the requirement that the minimum in (2) is only taken over j such that $\bar{k}^j \leq k$. Let $\mathcal{R}$ be the set of all mappings and $\mathcal{R}(k)$ be the set of all k-consistent mappings.

The next lemma shows a strong connection between the process $(X_n)_{n=1}^\infty$ and p.

Lemma 6. *For any $b \in \mathbb{Z}$ and $k \in \mathbb{N}$ it holds that $p(b,k) = \inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma} \left(S^{b,k} \right)$.*

While the proof of the lemma is fairly straightforward, it requires multiple technical steps. We give the proof in Section 6.3.1. By Lemma 6, in order to evaluate the asymptotic behavior of $p(\alpha k, k)$, we can focus on the asymptotics of $\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}(S^{\alpha k, k})$.

Denote the **type** of $(a_1, \ldots, a_n) \in \mathcal{X}^n$ by $\mathcal{T}(a_1, \ldots, a_n) = T \in \mathbb{R}_{\geq 0}^r$, where $T_i = \frac{|\{\ell | a_\ell = i\}|}{n}$ is the frequency of each $i \in \mathcal{X}$ in $(a_1, \ldots, a_n)$. Equivalently, $\mathcal{T}(a_1, \ldots, a_n) = \frac{1}{n} \sum_{\ell=1}^n \bar{e}^{a_\ell}$, where $\bar{e}^d \in \mathbb{R}^r$ is the d-th unit vector.[10] It can be easily verified that $\kappa(a_1, \ldots, a_n) = n \cdot \mathcal{T}(a_1, \ldots, a_n) \cdot K$ and $\mu(a_1, \ldots, a_n) = n \cdot \mathcal{T}(a_1, \ldots, a_n) \cdot B$ for any $(a_1, \ldots, a_n) \in \mathcal{X}^n$, where $\cdot$ is the standard dot product of two vectors. While $(X_n)_{n=1}^\infty$ is not a sequence of *i.i.d.* random variables, several properties of types (see [13]) are preserved.

Recall that for two distributions $\bar{c}, \bar{d} \in \mathbb{R}^t$, the classic Kullback-Leibler divergence is given by $D\left(\bar{c} \| \bar{d}\right) = \sum_{i=1}^t \bar{c}_i \log \frac{\bar{c}_i}{\bar{d}_i}$ (we follow the standard convention $0 = 0 \log 0 = 0 \log \frac{0}{0}$). We define the **extended Kullback-Leibler divergence** for any $\Upsilon^1, \Upsilon^2 \in \mathbb{R}_{\geq 0}^r$ by

$$D_e\left(\Upsilon^1 \| \Upsilon^2\right) = \sum_{i \in \mathcal{X}} \Upsilon_i^1 \log \frac{\Upsilon_i^1}{\Upsilon_i^2} - \sum_{j=1}^N \lambda_j \log \lambda_j \qquad \text{where} \qquad \lambda_j = \sum_{i \in \mathcal{X}_j} \Upsilon_i^1.$$

There is a simple interpretation for $D_e\left(\Upsilon^1 \| \Upsilon^2\right)$ when Υ^1 is a type and Υ^2 is an extend distribution. In this case, $D_e\left(\Upsilon^1 \| \Upsilon^2\right) = \sum_{j=1}^N \lambda_j \cdot D_j + H(\lambda_1, \ldots, \lambda_N)$, where D_j is the Kullback-Liebler divergence between the distributions $\left(\frac{\Upsilon_i^1}{\lambda_j}\right)_{i \in \mathcal{X}_j}$ and $\left(\Upsilon_i^2\right)_{i \in \mathcal{X}_j}$ for $1 \leq j \leq N$, and H is the entropy function.

Lemma 7. *For any $R \in \mathcal{R}$ and extended distribution Υ the following hold.*

1. *Let $(a_1, \ldots a_n) \in \mathcal{X}^n$ and $T = \mathcal{T}(a_1, \ldots, a_n)$. If $\Pr_{R,\Upsilon}(\forall 1 \leq \ell \leq n : X_\ell = a_\ell) > 0$ then*

$$\Pr_{R,\Upsilon}(\forall 1 \leq \ell \leq n : X_\ell = a_\ell) = \exp\left(-n \sum_{i \in \mathcal{X}} T_i \log \frac{1}{\Upsilon_i}\right).$$

2. *Let $K_n = \left\{\left(\frac{n_1}{n}, \ldots, \frac{n_r}{n}\right) \middle| \forall 1 \leq i \leq r : n_i \in \mathbb{N} \text{ and } \sum_{i=1}^r n_i = n\right\}$. Then, for any $\bar{a} \in \mathcal{X}^n$, $\mathcal{T}(\bar{a}) \in K_n$. In particular, $|K_n| \leq (n+1)^r$.*

3. *Let $T \in K_n$, then*

$$\Pr_{R,\Upsilon}(\mathcal{T}(X_1, \ldots, X_n) = T) \leq \exp(-n D_e(T \| \Upsilon)).$$

---

[9]Our definitions implicitly assume the existence of a measurable space $(\Omega, \mathcal{F})$, such that Y_n^j and X_n are random variables with respect to $(\Omega, \mathcal{F})$ for any $n \geq 1$ and $1 \leq j \leq N$. In this terminology, $S^{b,k} \in \mathcal{F}$.

[10]$\bar{e}_i^d = 0$ for $i \neq d$ and $\bar{e}_d^d = 1$.

The proof of the lemma uses standard techniques in the method of types. A complete proof is given in Section 6.3.2.

Let $\bar{q}^j$ be the vector which solves (3) with respect to $(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)$. That is,

$$\bar{q}^j = \underset{\bar{q}\cdot\bar{b}^j \leq \alpha\bar{q}\cdot\bar{k}^j,\ \bar{q}\ \text{is a distribution}}{\arg\min} \frac{1}{\bar{q}\cdot\bar{k}^j} D\left(\bar{q}\big\|\bar{\gamma}^j\right). \tag{18}$$

Therefore, $\frac{1}{\bar{q}^j\cdot\bar{b}^j}D\left(\bar{q}^j\big\|\bar{\gamma}^j\right) = M_j$. As before, let Q be the concatenation of $\bar{q}^1, \bar{q}^2, \ldots, \bar{q}^N$. Formally, define $Q \in \mathbb{R}^r_{\geq 0}$ with $Q_{s_{j-1}+i} = \bar{q}^j_i$ for any $1 \leq j \leq N$ and $1 \leq i \leq r_j$.

The next lemma shows that a lower bound on the probability of events in $\Pr_{R,\Gamma}$ can be derived using a lower bound on the probability of the same events in $\Pr_{R,Q}$.

Lemma 8. *Let $A \subseteq \mathcal{X}^n$ such that $\mathcal{T}(\bar{a}) = T$ for any $\bar{a} \in A$. Then, for any $R \in \mathcal{R}$,*

$$\log\Pr_{R,\Gamma}((X_1, \ldots X_n) \in A) \geq \log\Pr_{R,Q}(X_1, \ldots X_n \in A) - n\sum_{i\in\mathcal{X}} T_i \log\frac{Q_i}{\Gamma_i}. \tag{19}$$

Note that the distribution function in (19) is $\Pr_{R,\Gamma}$ in the LHS and $\Pr_{R,Q}$ in the RHS.

Proof. Let $A' = \{(a_1, \ldots, a_n) \in A|\ \Pr_{R,Q}(\forall 1 \leq \ell \leq n : X_\ell = a_\ell) > 0\}$. Clearly,

$$\Pr_{R,Q}((X_1, \ldots, X_n) \in A) = \Pr_{R,Q}((X_1, \ldots, X_n) \in A').$$

For any $(a_1, \ldots, a_n) \in A'$, since $\Pr_{R,Q}(\forall 1 \leq \ell \leq n : X_\ell = a_\ell) > 0$, it holds that $a_\ell \in \mathcal{X}_{R(a_1,\ldots,a_{\ell-1})}$ and therefore $\Pr_{R,\Gamma}(\forall 1 \leq \ell \leq n : X_\ell = a_\ell) = \prod_{\ell=1}^n \Gamma_{a_\ell} > 0$. By Lemma 7,

$$\log\Pr_{R,\Gamma}((X_1, \ldots, X_n) \in A) \geq \log\Pr_{R,\Gamma}((X_1, \ldots, X_n) \in A')$$

$$= \log\left(\sum_{(a_1,\ldots,a_n)\in A'} \Pr_{R,\Gamma}(\forall 1 \leq \ell \leq n : X_\ell = a_\ell)\right)$$

$$= \log\left(\sum_{(a_1,\ldots a_n)\in A'} \exp\left(-n\sum_{i\in\mathcal{X}} T_i \log\frac{1}{\Gamma_i}\right)\right)$$

$$= \log\left(\sum_{(a_1,\ldots a_n)\in A'} \exp\left(-n\sum_{i\in\mathcal{X}} T_i \left(\log\frac{1}{Q_i} + \log\frac{Q_i}{\Gamma_i}\right)\right)\right)$$

$$= \log\left(\sum_{(a_1,\ldots a_n)\in A'} \exp\left(-n\sum_{i\in\mathcal{X}} T_i \log\frac{1}{Q_i}\right)\right) - n\sum_{i\in\mathcal{X}} T_i \log\frac{Q_i}{\bar{\gamma}_i}$$

$$= \log\left(\sum_{(a_1,\ldots a_n)\in A'} \Pr_{R,Q}(\forall 1 \leq \ell \leq n : X_\ell = a_\ell)\right) - n\sum_{i\in\mathcal{X}} T_i \log\frac{Q_i}{\Gamma_i}$$

$$= \log\left(\Pr_{R,Q}((X_1, \ldots, X_n) \in A)\right) - n\sum_{i\in\mathcal{X}} T_i \log\frac{Q_i}{\Gamma_i}$$

$\square$

The next technical result will be used in the proof of the subsequent lemma

Lemma 9. *For any type T and extended probability Υ, let $\lambda_j = \sum_{i\in\mathcal{X}_j} T_i$ for $1 \leq j \leq N$. Then it holds that*

$$\sum_{j=1}^N \sum_{i\in\mathcal{X}_j} |T_i - \lambda_j\Upsilon_i| \leq 2\sqrt{D_e\left(T\|\Upsilon\right)}$$

28

The lemma follows from a simple application of a known inequality relating ℓ_1-norms and Kullback-Leibler divergence along with Jensen's inequality. The proof is given in Section 6.3.2.

To utilize Lemma 8 we need to specify a subset $A \subseteq S^{\alpha k, k}$ such that all the elements in A have the same type and A has high probability. The next lemma shows a slightly relaxed claim.

Lemma 10. *For any $\varepsilon > 0$ there is $L > 0$ such that for any $k > L$ and rules mapping $R \in \mathcal{R}$, there is a length $n^{k,R} \in \mathbb{N}$ and a type $T^{k,R}$ such that:*

1.
$$\lim_{k \to \infty} \inf_{R \in \mathcal{R}} \frac{1}{k} \log \mathrm{Pr}_{R,Q} \left(\mathcal{T}(X_1, \dots, X_{n^{k,R}}) = T^{k,R} \right) = 0$$

2. *For any $k > L$, $R \in \mathcal{R}$ and $1 \le j \le N$ set $\lambda_j^{k,R} = \sum_{i \in \mathcal{X}_j} T_i^{k,R}$. Then,*
$$\lim_{k \to \infty} \sup_{R \in \mathcal{R}} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left| T_i^{k,R} - \lambda_j^{k,R} Q_i \right| = 0.$$

3. *For any $k \ge L$ and $R \in \mathcal{R}$ it holds that $n^{k,R} \cdot (K \cdot T^{k,R}) \ge k$, $n^{k,R} \cdot (B \cdot T^{k,R}) \le \alpha(1+\varepsilon)k$ and $n^{k,R} \le \alpha(1+\varepsilon)k$.*

Proof. Let $\varepsilon > 0$, $k \in \mathbb{N}$ and $R \in \mathcal{R}$. Also, let $s = \lceil k\alpha(1+\varepsilon) \rceil$. We note that $\mu(X_1, \dots, X_{s+1}) > \alpha(1+\varepsilon)k$ (since $B_i \ge 1$ for $i \in \mathcal{X}$). Thus,

$$1 = \sum_{n=1}^{s} \mathrm{Pr}_{R,Q} \left(\mu(X_1, \dots, X_n) \le \alpha(1+\varepsilon)k \text{ and } \mu(X_1, \dots, X_{n+1}) > \alpha(1+\varepsilon)k \right).$$

Hence, there is $n^{k,R} \le s$ such that

$$\mathrm{Pr}_{R,Q} \left(\mu(X_1, \dots, X_{n^{k,R}}) \le \alpha(1+\varepsilon)k \text{ and } \mu(X_1, \dots, X_{n^{k,R}+1}) > \alpha(1+\varepsilon)k \right) \ge \frac{1}{s+1}.$$

Since the type of $X_1, \dots, X_n$ is in K_n (see Lemma 7), we also have

$$\frac{1}{s+1} \le \sum_{T \in K_{n^{k,R}}} \mathrm{Pr}_{R,Q} \left(\begin{array}{ll} \mu(X_1, \dots, X_{n^{k,R}}) \le \alpha(1+\varepsilon)k & \text{and} \\ \mu(X_1, \dots, X_{n^{k,R}+1}) > \alpha(1+\varepsilon)k & \text{and} \\ \mathcal{T}(X_1, \dots, X_{n^{k,R}}) = T \end{array} \right)$$

Since $|K_{n^{k,R}}| \le (s+1)^r$, there is $T^{k,R} \in K_{n^{k,R}}$ such that

$$(\alpha(1+\varepsilon)k + 2)^{-(r+1)} \le \frac{1}{(s+1)^{r+1}} \le \mathrm{Pr}_{R,Q} \left(\begin{array}{ll} \mu(X_1, \dots, X_{n^{k,R}}) \le \alpha(1+\varepsilon)k & \text{and} \\ \mu(X_1, \dots, X_{n^{k,R}+1}) > \alpha(1+\varepsilon)k & \text{and} \\ \mathcal{T}(X_1, \dots, X_{n^{k,R}}) = T^{k,R} \end{array} \right) \quad (20)$$

Once we established how $n^{k,R}$ and $T^{k,R}$ are selected, it remains to show the properties in the lemma. The idea is that by our selection of the vectors $T^{k,R}$, the normalized vectors $\left(\frac{T_i^{k,R}}{\sum_{i' \in \mathcal{X}_j} T_{i'}^{k,R}} \right)_{i \in \mathcal{X}_j}$ cannot deviate significantly from $(Q_i)_{i \in \mathcal{X}_j} = \bar{q}^j$. This observation is used to derive the properties below. From equation (20) we have

$$0 \ge \inf_{R \in \mathcal{R}} \frac{1}{k} \log \mathrm{Pr}_{R,Q} \left(\mathcal{T}(X_1, \dots, X_{n^{k,R}}) = T^{k,R} \right)$$
$$\ge \inf_{R \in \mathcal{R}} \frac{1}{k} \log (\alpha(1+\varepsilon)k + 2)^{-(r+1)}$$
$$= \frac{-(r+1) \log (\alpha(1+\varepsilon)k + 2)}{k}$$

As the last term goes to 0 as k goes to infinity, it follows from the squeeze theorem that

$$\lim_{k \to \infty} \inf_{R \in \mathcal{R}} \frac{1}{k} \log \Pr_{R,Q} \left(\mathcal{T}(X_1, \ldots, X_{n^{k,R}}) = T^{k,R} \right) = 0.$$

Thus, we have shown Property 1.

For any $k \in \mathbb{N}$ and $R \in \mathcal{R}$, by Lemma 7 we have

$$(\alpha(1+\varepsilon)k+1)^{-(r+1)} \leq \Pr_{R,Q} \left(\mathcal{T}(X_1, \ldots, X_{n^{k,R}}) = T^{k,R} \right) \leq \exp \left(-n^{k,R} D_e \left(T^{k,R} \middle\| Q \right) \right).$$

Therefore,

$$D_e \left(T^{k,R} \middle\| Q \right) \leq \frac{(r+1)\log(\alpha(1+\varepsilon)k+2)}{n^{k,R}}.$$

Set $b_{\max} = \max_{i \in \mathcal{X}} B_i$, It follows from the definition of $n^{k,R}$ that $n^{k,R} \geq \frac{\alpha k}{b_{\max}}$. Thus,

$$D_e \left(T^{k,R} \middle\| Q \right) \leq \frac{b_{\max}(r+1)\log((\alpha+\varepsilon)k+2)}{\alpha k}.$$

Define $\lambda_j^{k,R} = \sum_{i \in \mathcal{X}_j} T_i^{k,R}$ for $1 \leq j \leq N$. It follows from Lemma 9 that

$$\sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left| T_i^{k,R} - \lambda_j^{k,R} Q_i \right| \leq 2 \cdot \sqrt{D_e(T^{k,R} \| Q)} \leq 2 \cdot \sqrt{\frac{b_{\max}(r+1)\log(\alpha(1+\varepsilon)k+2)}{\alpha k}}.$$

The right term approaches 0 as k goes to infinity, thus

$$\lim_{k \to \infty} \sup_{R \in \mathcal{R}} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left| T_i^{k,R} - \lambda_j^{k,R} Q_i \right| = 0. \tag{21}$$

Hence, we have shown Property 2. We use (21) to prove Property 3. For any $k \in \mathbb{N}$ and $R \in \mathcal{R}$, we have

$$\alpha \sum_{j=1}^{N} \lambda_j^{k,R} \sum_{i \in \mathcal{X}_j} Q_i K_i = \sum_{j=1}^{N} \lambda_j^{k,R} \alpha \sum_{i=1}^{r_j} \bar{q}_i^j \bar{k}_i^j \geq \sum_{j=1}^{N} \lambda_j^{k,R} \sum_{i=1}^{r_j} \bar{q}_i^j \bar{b}_i^j = \sum_{j=1}^{N} \lambda_j^{k,R} \sum_{i \in \mathcal{X}_j} Q_i B_i.$$

The inequality holds since $\alpha \bar{q}^j \cdot \bar{k}^j \geq \bar{q}^j \cdot \bar{b}^j$ for any $1 \leq j \leq N$, which follows from the definition of $\bar{q}^j$ in (18).

Hence, we have

$$\sum_{j=1}^{N} \lambda_j^{k,R} \sum_{i \in \mathcal{X}_j} Q_i(\alpha K_i - B_i) \geq 0.$$

Therefore,

$$n^{k,R} T^{k,R} \cdot (\alpha K - B) = n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} T_i^{k,R} \cdot (\alpha K_i - B_i)$$

$$= n^{k,R} \sum_{j=1}^{N} \lambda_j^{k,R} \sum_{i \in \mathcal{X}_j} Q_i \cdot (\alpha K_i - B_i) + n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} (T_i^{k,R} - \lambda_j^{k,R} Q_i) \cdot (\alpha K_i - B_i)$$

$$\geq -k \cdot (\alpha(1+\varepsilon)+1) \left(\max_{i \in \mathcal{X}} |(\alpha K_i - B_i)| \right) \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} |T_i^{k,R} - \lambda_j^{k,R} Q_i|$$

It follows from (21) that there is $L > 0$ such that for any $k > L$ and $R \in \mathcal{R}$,

$$(\alpha(1+\varepsilon) + 1)\left(\max_{i \in \mathcal{X}} |(\alpha K_i - B_i)|\right) \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} |T_i^{k,R} - \lambda_j^{k,R} Q_i| < \frac{\varepsilon \alpha}{2}.$$

Also, for any $k \in \mathbb{N}$ and $R \in \mathcal{R}$ it follows from the definition of $n^{k,R}$ and $T^{k,R}$ that

$$\alpha(1+\varepsilon)k - b_{\max} < n^{k,R} \cdot T^{k,R} \cdot B \leq \alpha(1+\varepsilon)k.$$

Combining the above we have

$$n^{k,R} T^{k,R} \cdot K \geq \frac{1}{\alpha}\left(n^{k,R} T^{k,R} \cdot B + n^{k,R} T^{k,R} \cdot (\alpha K - B)\right)$$
$$\geq \frac{\alpha(1+\varepsilon)}{\alpha} k - \frac{b_{\max}}{\alpha} - \frac{\varepsilon \alpha}{2\alpha} k = k + \frac{\varepsilon}{2} k - \frac{b_{\max}}{\alpha} \geq k,$$

where the last inequality holds for $k > \frac{2 \cdot b_{\max}}{\alpha \varepsilon}$. $\qquad\square$

The following lemma is derived by combining the results of Lemmas 8 and 10.

Lemma 11. *For any $\varepsilon > 0$,*

$$\liminf_{k \to \infty} \inf_{R \in \mathcal{R}} \frac{1}{k} \log \Pr_{R,\Gamma}\left(S^{\alpha(1+\varepsilon)k,k}\right) \geq -M(1+\varepsilon)$$

Proof. Let $\varepsilon > 0$. Also, let $(n^{k,R})_{k \in \mathbb{N}, R \in \mathcal{R}}$ and $(T^{k,R})_{k \in \mathbb{N}, R \in \mathcal{R}}$ be the numbers and types derived from Lemma 10. Define $\lambda_j^{k,R} = \sum_{i \in \mathcal{X}_j} T_i^{k,R}$. By Lemma 8, for any $k \in \mathbb{N}$ and $R \in \mathcal{R}$, it holds that

$$\frac{1}{k} \log \Pr_{R,\Gamma}\left(S^{\alpha(1+\varepsilon)k,k}\right) \geq \frac{1}{k} \log \Pr_{R,\Gamma}(\mathcal{T}(X_1, \ldots, X_{n^{k,R}}) = T^{k,R})$$
$$\geq \frac{1}{k} \log \Pr_{R,Q}(\mathcal{T}(X_1, \ldots, X_{n^{k,R}}) = T^{k,R}) - \frac{1}{k} n^{k,R} \sum_{i \in \mathcal{X}} T_i^{k,R} \log \frac{Q_i}{\Gamma_i}$$

.

By Lemma 10, the first term converges to 0 as k goes to infinity. Therefore,

$$\liminf_{k \to \infty} \inf_{R \in \mathcal{R}} \frac{1}{k} \log \Pr_{R,\Gamma}\left(S^{\alpha(1+\varepsilon)k,k}\right) \geq -\limsup_{k \to \infty} \sup_{R \in \mathcal{R}} \frac{1}{k} n^{k,R} \sum_{i \in \mathcal{X}} T_i^{k,R} \log \frac{Q_i}{\Gamma_i}$$

$$\geq -\limsup_{k \to \infty} \sup_{R \in \mathcal{R}} \frac{1}{k} n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left(\lambda_j^{k,R} Q_i + T_i^{k,R} - \lambda_j^{k,R} Q_i\right) \log \frac{Q_i}{\Gamma_i}$$

$$\geq -\limsup_{k \to \infty} \sup_{R \in \mathcal{R}} \frac{1}{k} n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \lambda_j^{k,R} Q_i \log \frac{Q_i}{\Gamma_i}$$

$$- \limsup_{k \to \infty} \sup_{R \in \mathcal{R}} \frac{1}{k} n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left|T_i^{k,R} - \lambda_j^{k,R} Q_i\right| \max_{i \in \mathcal{X}, Q_i > 0} \left|\log \frac{Q_i}{\Gamma_i}\right|$$

$$\geq -\limsup_{k \to \infty} \sup_{R \in \mathcal{R}} \frac{1}{k} n^{k,R} \sum_{j=1}^{N} \lambda_j^{k,R} \sum_{i \in \mathcal{X}_j} Q_i \log \frac{Q_i}{\Gamma_i}.$$

The third inequality uses the observation that, for large enough k, if $Q_i = 0$ then $T_i^{k,R} = 0$ (otherwise $\Pr_{R,Q}(\mathcal{T}(X_1, \ldots X_{n^{k,R}}) = T^{k,R}) = 0$). The last inequality follows from $n^{k,R} \leq \alpha(1+\varepsilon)k$ and Property 2 in Lemma 10.

31

Recall that $\bar{q}^j$ is found by solving the constrained optimization problem in (18). Therefore, by Kuhn-Tucker conditions, for any $1 \le j \le N$, either $\alpha \bar{k}^j \cdot \bar{q}^j = \bar{b}^j \cdot \bar{q}^j$, or $\bar{q}^j$ is a local minima of $h(\bar{q}) = \frac{D(\bar{q}\|\bar{\gamma}^j)}{\bar{k}^j \cdot \bar{q}}$. It follows from [1] that h is quasiconvex (see also Section 5) and therefore its only local minimum is at $\bar{q} = \bar{\gamma}^j$. Hence, for any $1 \le j \le N$, either $\alpha \bar{k}^j \cdot \bar{q}^j = \bar{b}^j \cdot \bar{q}^j$ or $D(\bar{q}^j\|\bar{\gamma}^j) = 0$. We get that

$$
\begin{aligned}
\frac{1}{k} n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \lambda_j^{k,R} Q_i \log \frac{Q_i}{\Gamma_i} &= \frac{1}{k} n^{k,R} \sum_{j=1}^{N} \lambda_j^{k,R} D(\bar{q}^j\|\bar{\gamma}^j) \\
&= \frac{1}{k} n^{k,R} \sum_{j=1}^{N} \lambda_j^{k,R} \frac{\bar{q}^j \cdot \bar{b}^j}{\alpha \bar{q}^j \cdot \bar{k}^j} D(\bar{q}^j\|\bar{\gamma}^j) \\
&\le \frac{1}{\alpha k} n^{k,R} \sum_{j=1}^{N} \lambda_j^{k,R} \bar{q}^j \cdot \bar{b}^j M \\
&= M \frac{1}{k\alpha} n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \lambda_j^{k,R} Q_i B_i \\
&= M \frac{1}{k\alpha} n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} T_i^{k,R} \cdot B_i + M \frac{1}{\alpha k} n^{k,R} \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left((\lambda_j^{k,R} Q_i - T_i^{k,R}) \cdot B_i \right) \\
&\le M \frac{n^{k,R} T^{k,R} \cdot B}{\alpha k} + \frac{n^{k,R}}{\alpha k} \max_{i \in \mathcal{X}} |B_i| \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left| \lambda_j^{k,R} Q_i - T_i^{k,R} \right| \\
&\le M(1+\varepsilon) + \frac{\alpha(1+\varepsilon)}{\alpha} \max_{i \in \mathcal{X}} |B_i| \sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} \left| \lambda_j^{k,R} Q_i - T_i^{k,R} \right|.
\end{aligned}
\tag{22}
$$

The first inequality follows from the definitions of α-branching numbers and M. The last inequality uses Property 3 in Lemma 10. In particular, $n^{k,R} \le \alpha(1+\varepsilon)k$ and $n^{k,R} \cdot T^{k,R} \cdot B \le \alpha(1+\varepsilon)k$. Combining the above, and using Property 2 in Lemma 10, we have,

$$
\liminf_{k \to \infty} \inf_{R \in \mathcal{R}} \frac{1}{k} \log \mathrm{Pr}_{R,\Gamma} \left(S^{(\alpha+\varepsilon)k,k} \right) \ge -M(1+\varepsilon).
$$

$\square$

Combining Lemma 11 with Lemma 6, we obtain the following (note that, by the definition of $S^{b,k}$, $S^{b,k} = S^{\lfloor b \rfloor, k}$).

Lemma 12. *For any $\varepsilon > 0$, $\liminf_{k \to \infty} \frac{1}{k} \log p(\lfloor \alpha(1+\varepsilon)k \rfloor, k) \ge -M(1+\varepsilon)$.*

While this claim is strictly weaker than the statement of Theorem 2, it suffices for deriving all of our algorithmic results.

To complete the proof of the theorem, we need to slightly improve the lower bound stated above to be independent of ε, and to derive a matching upper bound.

The claim in Lemma 10 can be strengthened to yield types $T^{k,R}$ and length $n^{k,R}$ such that $n^{k,R} \cdot T^{k,R} \cdot B \le \alpha k$ (omitting the ε term from Lemma 10). This stronger property, however, requires restricting the rules mappings considered to be k-consistent.

Lemma 13. *There is $L > 0$ such that, for any $k > L$ and rules mapping $R \in \mathcal{R}(k)$, there is a length $n^{k,R} \in \mathbb{N}$ and a type $T^{k,R}$ satisfying*

1.

$$
\lim_{k \to \infty} \inf_{R \in \mathcal{R}(k)} \frac{1}{k} \log \mathrm{Pr}_{R,Q} \left(\mathcal{T}(X_1, \ldots, X_{n^{k,R}}) = T^{k,R} \right) = 0
$$

2. For any $k > L$, $R \in \mathcal{R}(k)$ and $1 \le j \le N$ set $\lambda_j^{k,R} = \sum_{i \in \mathcal{X}_j} T_i^{k,R}$. Then

$$\lim_{k \to \infty} \sup_{R \in \mathcal{R}(k)} \sum_{j=1}^N \sum_{i \in \mathcal{X}_j} \left| T_i^{k,R} - \lambda_j^{k,R} Q_i \right| = 0.$$

3. For any $k \ge L$ and $R \in \mathcal{R}(k)$ it holds that $n^{k,R} \cdot (K \cdot T^{k,R}) \ge k$, $n^{k,R} \cdot (B \cdot T^{k,R}) \le \alpha k$ and $n^{k,R} = O(k)$.

The proof of Lemma 13 is highly technical and therefore deferred to Section 6.3.3. Using Lemma 13 we obtain the following.

Lemma 14.
$$\liminf_{k \to \infty} \frac{1}{k} \ \log \inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma} \left(S^{\alpha k, k} \right) \ge -M$$

The proof of Lemma 14 is essentially identical to the proof of Lemma 11. The main difference is in using the lengths and types ($n^{k,R}$ and $T^{k,R}$) generated by Lemma 13 instead of those of Lemma 10. Thus, we omit the proof.

An upper bound over $\inf_{R \in \mathcal{R}} \Pr_{R,\Gamma}(S^{b,k})$ can be easily derived using standard method of types arguments, as shown in Lemma 15. However, as we want to provide an upper bound on $p(\alpha k, k)$, it follows from Lemma 6 that an upper bound should be given for $\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}(S^{b,k})$. Although this small difference is fairly easy to overcome, it renders the proof more technically involved.

Let $1 \le j^* \le N$ be the index satisfying $M_{j^*} = M$, and define a function $F(\beta)$ such that $F(\beta)$ is the β-branching number of $(\bar{b}^{j^*}, \bar{k}^{j^*}, \bar{\gamma}^{j^*})$. Clearly, $F(\alpha) = M$, and it follows from standard calculus arguments that F is continuous on $[\alpha, \infty)$.

Lemma 15. *For any $\beta \ge \alpha$ and $k \in \mathbb{N}$ it holds that*

$$\inf_{R \in \mathcal{R}} \Pr_{R,\Gamma} \left(S^{\beta k, k} \right) \le (\beta k + 1)^{r+1} \cdot \exp(-F(\beta) \cdot k).$$

Proof. Define $R^* \in \mathcal{R}$ by $R^*(\bar{a}) = j^*$ for any $\bar{a} \in \mathcal{X}^*$. Then,

$$
\begin{aligned}
\Pr_{R^*,\Gamma} \left(S^{\beta k, k} \right) &= \Pr_{R^*,\Gamma} \left(\exists n : \mu(X_1, \ldots, X_n) \le \beta k \text{ and } \kappa(X_1, \ldots, X_n) \ge k \right) \\
&= \Pr_{R^*,\Gamma} \left(\exists 1 \le n \le \beta k : \mu(X_1, \ldots, X_n) \le \beta k \text{ and } \kappa(X_1, \ldots, X_n) \ge k \right) \\
&\le \sum_{n=1}^{\lfloor \beta k \rfloor} \Pr_{R^*,\Gamma} \left(\mu(X_1, \ldots, X_n) \le \beta k \text{ and } \kappa(X_1, \ldots, X_n) \ge k \right) \\
&\le \sum_{n=1}^{\lfloor \beta k \rfloor} \Pr_{R^*,\Gamma} \left(n \cdot B \cdot \mathcal{T}(X_1, \ldots, X_n) \le \beta k \text{ and } n \cdot K \cdot \mathcal{T}(X_1, \ldots, X_n) \ge k \right) \\
&= \sum_{n=1}^{\lfloor \beta k \rfloor} \sum_{T \in \{T' \in K_n | n \cdot B \cdot T' \le \beta k, n \cdot K \cdot T' \ge k\}} \Pr_{R^*,\Gamma} \left(\mathcal{T}(X_1, \ldots, X_n) = T \right)
\end{aligned}
$$

(23)

The second equality holds since $\mu(a_1, \ldots, a_n) \ge n$ for any $(a_1, \ldots, a_n) \in \mathcal{X}^n$. The second property of Lemma 7 is used in the last equality.

Now, let $1 \le n \le \beta k$, and consider $T \in \{T' \in K_n | n \cdot B \cdot T' \le \beta k, n \cdot K \cdot T' \ge k\}$. If there is $i \in \mathcal{X} \setminus \mathcal{X}_{j^*}$ such that $T_i > 0$, then

$$\Pr_{R^*,\Gamma} \left(\mathcal{T}(X_1, \ldots, X_n) = T \right) \le \Pr_{R^*,\Gamma} \left(\exists 1 \le \ell \le n : Y_\ell^{j^*} = i \right) = 0. \tag{24}$$

33

Now, consider the case where $T_i = 0$ for all $i \in \mathcal{X} \setminus \mathcal{X}_{j^*}$. Let $\bar{t} \in \mathbb{R}^{r_{j^*}}$ be defined by $\bar{t}_i = T_{s_{j^*-1}+i}$ for $1 \leq i \leq r_{j^*}$ (recall that $\mathcal{X}_{j^*} = \{s_{j^*-1} + 1, \ldots, s_{j^*} + r_{j^*}\}$). Also, let $\lambda_j = \sum_{i \in \mathcal{X}_j} T_i$ for $1 \leq j \leq N$. It follows that $\lambda_{j^*} = 1$ and $\lambda_j = 0$ for any $j \neq j^*$. By Lemma 7 we have,

$$
\begin{aligned}
\mathrm{Pr}_{R^*,\Gamma}\left(\mathcal{T}(X_1, \ldots, X_n) = T\right) &\leq \exp(-n D_e\left(T \| \Gamma\right)) \\
&= \exp\left(-n\left(\sum_{i \in \mathcal{X}} T_i \log \frac{T_i}{\Gamma_i} - \sum_{j=1}^N \lambda_j \log \lambda_j\right)\right) \\
&= \exp\left(-n\left(\sum_{i \in \mathcal{X}} T_i \log \frac{T_i}{\Gamma_i}\right)\right) \\
&= \exp\left(-n\left(\sum_{i=1}^{r_{j^*}} \bar{t}_i \log \frac{\bar{t}_i}{\bar{\gamma}_i^{j^*}}\right)\right) \\
&= \exp\left(-n D\left(\bar{t} \,\middle\|\, \bar{\gamma}^{j^*}\right)\right) \\
&\leq \exp\left(-k \frac{1}{\bar{t}^{j^*} \cdot \bar{k}^{j^*}} D\left(\bar{t} \,\middle\|\, \bar{\gamma}^{j^*}\right)\right) \\
&\leq \exp\left(-k \cdot F(\beta)\right)
\end{aligned} \tag{25}
$$

The second equality follows from $\lambda_{j^*} = 1$ and $\lambda_j = 0$ for $j \neq j^*$. The third equality follows from the definitions $\bar{t}$ and Γ, and the forth from the definition of Kullback-Leibler divergence. The second inequality follows from $n \cdot \bar{t} \cdot \bar{k}^{j^*} = n \cdot T \cdot K \geq k$ (note that the divergence is a non-negative function). For the last inequality a more involved argument is used. Note that

$$
\beta n \cdot \bar{t} \cdot \bar{k}^{j^*} = \beta n \cdot T \cdot K \leq \beta k \leq n \cdot T \cdot B = n \cdot \bar{t} \cdot \bar{b}^{j^*}.
$$

Therefore, $\bar{t} \cdot \bar{b}^{j^*} \leq \beta \bar{t} \cdot \bar{k}^{j^*}$. As $\bar{t}$ is a distribution, it follows from the definition of branching numbers (Definition 1) that $F(\beta) \leq \frac{1}{\bar{t}^{j^*} \cdot \bar{k}^{j^*}} D\left(\bar{t} \,\middle\|\, \bar{\gamma}^{j^*}\right)$.

Using (23), (24) and (25) together we obtain the following.

$$
\begin{aligned}
\mathrm{Pr}_{R^*,\Gamma}\left(S^{\beta k, k}\right) &\leq \sum_{n=1}^{\lfloor \beta k \rfloor} \sum_{T \in \left\{ T' \in K_n \,\middle|\, \begin{array}{l} n \cdot B \cdot T' \leq \beta k, \\ n \cdot K \cdot T' \geq k \end{array} \right\}} \mathrm{Pr}_{R^*,\Gamma}\left(\mathcal{T}(X_1, \ldots, X_n) = T\right) \\
&\leq \sum_{n=1}^{\lfloor \beta k \rfloor} \sum_{T \in \left\{ T' \in K_n \,\middle|\, \begin{array}{l} n \cdot B \cdot T' \leq \beta k, \\ n \cdot K \cdot T' \geq k \end{array} \right\}} \exp\left(-k F(\beta)\right) \\
&\leq (\beta k + 1)^{r+1} \exp(-k \cdot F(\beta)).
\end{aligned}
$$

The last inequality follows from $|K_n| \leq (n+1)^r$ (Lemma 7). $\qquad\square$

The next lemma shows how an upper bounds over $\inf_{R \in \mathcal{R}} \mathrm{Pr}_{R,\Gamma}(S^{b,k})$ can be used to derive an upper bound over $\inf_{R \in \mathcal{R}(k)} \mathrm{Pr}_{R,\Gamma}(S^{b,k})$.

Lemma 16. *For any $b \in \mathbb{Z}$ and $k \in \mathbb{N}$, it holds that*

$$
\inf_{R \in \mathcal{R}(k)} \mathrm{Pr}_{R,\Gamma}\left(S^{b,k}\right) \leq \inf_{R \in \mathcal{R}} \mathrm{Pr}_{R,\Gamma}\left(S^{b,k-k_{\max}}\right),
$$

where $k_{\max} = \max_{i \in \mathcal{X}} K_i$.

The proof of Lemma 16 is given in Section 6.3.1. We use the above lemmas to obtain an upper bound on $\limsup_{k\to\infty} \inf_{R\in\mathcal{R}(k)} \Pr_{R,\Gamma}(S^{\alpha k,k})$.

Lemma 17.
$$\limsup_{k\to\infty} \frac{1}{k} \log \inf_{R\in\mathcal{R}(k)} \Pr_{R,\Gamma}(S^{\alpha k,k}) \leq -M$$

Proof. Using Lemmas 16 and 15.

$$\limsup_{k\to\infty} \frac{1}{k} \log \inf_{R\in\mathcal{R}(k)} \Pr_{R,\Gamma}(S^{\alpha k,k}) = \limsup_{k\to\infty} \frac{1}{k} \log \inf_{R\in\mathcal{R}(k)} \Pr_{R,\Gamma}(S^{\lfloor \alpha k\rfloor,k})$$

$$\leq \limsup_{k\to\infty} \frac{1}{k} \log \inf_{R\in\mathcal{R}} \Pr_{R,\Gamma}(S^{\lfloor \alpha k\rfloor,k-k_{\max}})$$

$$= \limsup_{k\to\infty} \frac{1}{k} \log \inf_{R\in\mathcal{R}} \Pr_{R,\Gamma}(S^{\alpha k,k-k_{\max}})$$

$$\leq \limsup_{k\to\infty} \frac{1}{k} \log \left((\alpha k + 1)^{r+1} \cdot \exp\left(-F\left(\frac{\alpha k}{k - k_{\max}} \right) \cdot k \right) \right)$$

$$= \limsup_{k\to\infty} -F\left(\frac{\alpha k}{k - k_{\max}} \right)$$

$$= -F(\alpha)$$

$$= -M$$

The first and second equalities use the observation that $S^{\lfloor b\rfloor,k} = S^{b,k}$ by definition. The forth equality uses the continuity of F. $\square$

By Lemmas 17 and 14, it follows that

$$\lim_{k\to\infty} \frac{1}{k} \log \inf_{R\in\mathcal{R}(k)} \Pr_{R,\Gamma}(S^{\alpha k,k}) = -M.$$

Therefore, using Lemma 6, we have

$$\lim_{k\to\infty} \frac{1}{k} \log p(\lfloor \alpha k\rfloor, k) = \lim_{k\to\infty} \frac{1}{k} \log \inf_{R\in\mathcal{R}(k)} \Pr_{R,\Gamma}(S^{\alpha k,k}) = -M.$$

This completes the proof of Theorem 2. $\square$

6.3 Deferred Proofs

6.3.1 The Stochastic Process and Recurrence Relations

In this section we prove Lemmas 6 and 16. We start with some technical definitions and observations.

For any $b, k \in \mathbb{R}$ define $A^{b,k} = \emptyset$ if $b < 0$, $A^{b,k} = \{\epsilon\}$ if $k \leq 0$ and $b \geq 0$ ($\{\epsilon\}$ is the set which only includes the 0-dimensional vector), and in all other cases,

$$A^{b,k} = \left\{ (a_1, \ldots, a_n) \in \mathcal{X}^* \;\middle|\; \begin{array}{l} \mu(a_1, \ldots, a_n) \leq b \\ \kappa(a_1, \ldots, a_n) \geq k \\ \kappa(a_1, \ldots, a_{n-1}) < k \end{array} \right\}. \tag{26}$$

It follows from the definitions that

$$S^{b,k} = \{\exists (a_1, \ldots, a_n) \in A^{b,k} : (X_1, \ldots X_n) = (a_1, \ldots, a_n)\}. \tag{27}$$

The definition of $S^{b,k}$ using $A^{b,k}$ is useful because of the following properties. For any two vectors $(a_1, \ldots, a_n), (c_1, \ldots, c_m) \in A^{b,k}$, such that $(a_1, \ldots, a_n) \neq (c_1, \ldots, c_m)$, it holds that,

$$\{(X_1, \ldots, X_n) = (a_1, \ldots, a_n)\} \cap \{(X_1, \ldots, X_m) = (c_1, \ldots, c_m)\} = \emptyset.$$

Note that the sets above are events (element in $\mathcal{F}$). As for any $a_1, \ldots a_n \in \mathcal{X}_n$ with $n > b$ it holds that $\mu(a_1, \ldots, a_n) > b$, we conclude that all the vectors in $A^{b,k}$ are of dimension not greater than b. Therefore the sets $A^{b,k}$ are all finite. Finally, it is easy to verify the sets have a recursive structure. For any $k > 0$ and $b \in \mathbb{R}$, it follows from the definition that

$$A^{b,k} = \left\{ (a_1, \ldots, a_n) \in \mathcal{X}^* \,\middle|\, n \geq 1, (a_2, \ldots, a_n) \in A^{b-B_{a_1}, k-K_{a_1}} \right\}. \tag{28}$$

The following technical lemmas will be useful in obtaining later results. For any $R \in \mathcal{R}$ and $i \in \mathcal{X}$ define $R_i \in \mathcal{R}$ by $R_i(a_1, \ldots, a_n) = R(i, a_1, a_2, \ldots, a_n)$.

Lemma 18. *For any extended distribution Υ, $R \in \mathcal{R}$ and $(a_1, \ldots a_n) \in \mathcal{X}^n$ such that $n \geq 1$, it holds that*

$$\Pr_{R,\Upsilon} ((X_1, \ldots X_n) = (a_1, \ldots, a_n)) = \Pr_{R,\Upsilon}(X_1 = a_1) \cdot \Pr_{R_{a_1}, \Upsilon} ((X_1, \ldots, X_{n-1}) = (a_2, \ldots, a_n))$$

Proof. It follows from the definition of the stochastic process $(X_n)_{n=1}^\infty$ that

$$\begin{aligned}
\Pr_{R,\Upsilon} ((X_1, \ldots X_n) = (a_1, \ldots, a_n)) &= \Pr_{R,\Upsilon} \left(\forall 1 \leq \ell \leq n : Y_\ell^{R(a_1, \ldots, a_{\ell-1})} = a_\ell \right) \\
&= \Pr_{R,\Upsilon} \left(Y_1^{R(\epsilon)} = a_1 \right) \cdot \Pr_{R,\Upsilon} \left(\forall 2 \leq \ell \leq n : Y_\ell^{R(a_1, \ldots, a_{\ell-1})} = a_\ell \right) \\
&= \Pr_{R,\Upsilon} (X_1 = a_1) \cdot \prod_{\ell=2}^n \begin{cases} \Upsilon a_\ell & a_\ell \in \mathcal{X}_{R_{a_1}(a_2, \ldots, a_{\ell-1})} \\ 0 & \text{otherwise} \end{cases} \\
&= \Pr_{R,\Upsilon} (X_1 = a_1) \cdot \Pr_{R_{a_1}, \Upsilon} \left(\forall 1 \leq \ell \leq n-1 : Y_\ell^{R_{a_1}(a_2, \ldots, a_\ell)} = a_{\ell+1} \right) \\
&= \Pr_{R,\Upsilon} (X_1 = a_1) \cdot \Pr_{R_{a_1}, \Upsilon} ((X_1, \ldots, X_{n-1}) = (a_2, \ldots, a_n))
\end{aligned}$$

$\qquad\square$

This leads to the next lemma.

Lemma 19. *Let Υ be an extended distribution, $R \in \mathcal{R}$, $b \in \mathbb{Z}$ and $k \in \mathbb{N}_+$. Set $j = R(\epsilon)$. Then,*

$$\Pr_{R,\Upsilon} \left(S^{b,k} \right) = \sum_{i \in \mathcal{X}_j} \Upsilon_i \cdot \Pr_{R,\Upsilon} \left(S^{b-B_i, k-K_i} \right)$$

Proof.

$$\begin{aligned}
\Pr_{R,\Upsilon} \left(S^{b,k} \right) &= \Pr_{R,\Upsilon} \left(\exists (a_1, \ldots, a_n) \in A^{b,k} : (X_1, \ldots X_n) = (a_1, \ldots a_n) \right) \\
&= \sum_{(a_1, \ldots, a_n) \in A^{b,k}} \Pr_{R,\Upsilon} ((X_1, \ldots, X_n) = (a_1, \ldots a_n)) \\
&= \sum_{(a_1, \ldots, a_n) \in A^{b,k}} \Pr_{R,\Upsilon}(X_1 = a_1) \cdot \Pr_{R_{a_1}, \Upsilon} ((X_1, \ldots, X_{n-1}) = (a_2, \ldots a_n)) \\
&= \sum_{i \in \mathcal{X}} \sum_{(a_1, \ldots, a_n) \in A^{b-B_i, k-K_i}} \Pr_{R,\Upsilon}(X_1 = i) \cdot \Pr_{R_i, \Upsilon} ((X_1, \ldots, X_n) = (a_1, \ldots a_n)) \\
&= \sum_{i \in \mathcal{X}} \Pr_{R,\Upsilon}(X_1 = i) \cdot \sum_{(a_1, \ldots, a_n) \in A^{b-B_i, k-K_i}} \Pr_{R_i, \Upsilon} ((X_1, \ldots, X_n) = (a_1, \ldots a_n)) \\
&= \sum_{i \in \mathcal{X}} \Pr_{R,\Upsilon}(Y_1^j = i) \cdot \Pr_{R_i, \Upsilon} \left(\exists (a_1, \ldots, a_n) \in A^{b-B_i, k-K_i} : (X_1, \ldots, X_n) = (a_1, \ldots, a_n) \right) \\
&= \sum_{i \in \mathcal{X}_j} \Upsilon_i \cdot \Pr_{R_i, \Gamma} \left(S^{b-B_i, k-K_i} \right)
\end{aligned}$$

The first and last equalities follow from (27). The second and sixth equalities uses the observation that $A^{b,k}$ is finite. The third equality is due to Lemma 18. The forth equality is derived from (28). The last equality also uses the definition of Y_1^j. $\qquad\square$

Now we are ready for proving Lemmas 6 and 16.

Proof of Lemma 6. We prove the claim by induction on b. For $b < 0$, for any $R \in \mathcal{R}$ we have $\Pr_{R,\Gamma}(S^{b,k}) = 0$; therefore, $\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}(S^{b,k}) = 0 = p(b,k)$.

Let $b \in \mathbb{N}$ and assume the induction hypothesis holds for any smaller value of b. If $k = 0$ then for any $R \in \mathcal{R}(k)$ we have $\Pr_{R,\Gamma}(S^{b,k}) = 1$; therefore, $\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}(S^{b,k}) = 1 = p(b,k)$.

It remains to handle the case where $k > 0$. For any $1 \leq j \leq N$ for which $\bar{k}^j \leq k$ (recall that $\bar{k}^j \leq k$ if, for all $1 \leq i \leq r_j$, $\bar{k}_i^j \leq k$) define $R^j \in \mathcal{R}(k)$ such that $R^j(\epsilon) = j$ and $\Pr_{R^j,\Gamma}(S^{b,k}) = \inf_{R \in \mathcal{R}(k), R(\epsilon)=j} \Pr_{R,\Gamma}(S^{b,k})$. As for any $R \in \mathcal{R}(k)$ $\bar{k}^{R(\epsilon)} \leq k$, we have $\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}(S^{b,k}) = \min_{1 \leq j \leq N, \bar{k}^j \leq k} \Pr_{R^j,\Gamma}(S^{b,k})$. By the definition of R^j, it also holds that, for any $i \in \mathcal{X}_j$, $\Pr_{R_i^j,\Gamma}(S^{b-B_i,k-K_i}) = \inf_{R \in \mathcal{R}(k-K_i)} \Pr_{R,\Gamma}(S^{b-B_i,k-K_i})$ (for any $1 \leq j \leq N$ such that $\bar{k}^j \leq k$).

We use the above in the following equalities.

$$
\begin{aligned}
\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}\left(S^{b,k}\right) &= \min_{1 \leq j \leq N, \bar{k}^j \leq k} \Pr_{R^j,\Gamma}(S^{b,k}) \\
&= \min_{1 \leq j \leq N, \bar{k}^j \leq k} \sum_{i \in \mathcal{X}_j} \Gamma_i \cdot \Pr_{R_i^j,\Gamma}(S^{b-B_i,k-K_i}) &&\text{By Lemma 19} \\
&= \min_{1 \leq j \leq N, \bar{k}^j \leq k} \sum_{i \in \mathcal{X}_j} \Gamma_i \cdot \inf_{R \in \mathcal{R}(k-K_i)} \Pr_{R,\Gamma}(S^{b-B_i,k-K_i}) \\
&= \min_{1 \leq j \leq N, \bar{k}^j \leq k} \sum_{i \in \mathcal{X}_j} \Gamma_i \cdot p(b-B_i, k-K_i) &&\text{Induction Hypothesis} \\
&= \min_{1 \leq j \leq N, \bar{k}^j \leq k} \sum_{i=1}^{r_j} \bar{\gamma}_i^j \cdot p(b-\bar{b}_i^j, k-\bar{k}_i^j) = p(b,k)
\end{aligned}
$$

The fifth equality follows from the definitions of Γ, B and K. $\qquad\square$

Proof of Lemma 16. We prove the Lemma by induction on b. For $b < 0$ it holds that

$$
\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}\left(S^{b,k}\right) = 0 = \inf_{R \in \mathcal{R}} \Pr_{R,\Gamma}\left(S^{b,k-k_{\max}}\right).
$$

Let $b \geq 0$ and assume the claim holds for smaller values of b. If $k \leq k_{\max}$ then

$$
\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma}\left(S^{b,k}\right) \leq 1 = \inf_{R \in \mathcal{R}} \Pr_{R,\Gamma}\left(S^{b,k-k_{\max}}\right),
$$

and the claim holds.

We are left with the case where $k > k_{\max}$. Let $R^* \in \mathcal{R}$ such that

$$
\Pr_{R^*,\Gamma}\left(S^{b,k-k_{\max}}\right) = \inf_{R \in \mathcal{R}} \Pr_{R,\Gamma}\left(S^{b,k-k_{\max}}\right)
$$

and set $j = R^*(\epsilon)$. We also note that by the definition of R^*, for any $i \in \mathcal{X}_j$,

$$
\Pr_{R_i^*,\Gamma}\left(S^{b-B_i,k-k_{\max}-K_i}\right) = \inf_{R \in \mathcal{R}} \Pr_{R,\Gamma}\left(S^{b-B_i,k-k_{\max}-K_i}\right). \tag{29}
$$

Let $R^c \in \mathcal{R}(k)$ such that $R^c(\epsilon) = j$, and

$$
\Pr_{R^c,\Gamma}\left(S^{b,k}\right) = \inf_{R \in \mathcal{R}(k), R(\epsilon)=j} \Pr_{R,\Gamma}\left(S^{b,k}\right).
$$

Note that since $k > k_{\max}$ such R^c exists. It follows from the definition of R^c that for any $i \in \mathcal{X}_j$,

$$
\Pr_{R_i^c,\Gamma}\left(S^{b-B_i,k-K_i}\right) = \inf_{R \in \mathcal{R}(k-K_i)} \Pr_{R,\Gamma}\left(S^{b-B_i,k-k_{\max}-K_i}\right). \tag{30}
$$

37

This leads to the following inequalities.

$$\inf_{R \in \mathcal{R}(k)} \Pr_{R,\Gamma} \left(S^{b,k} \right) \le \Pr_{R^c,\Gamma} \left(S^{b,k} \right)$$

$$= \sum_{i \in \mathcal{X}_j} \Gamma_i \cdot \Pr_{R_i^c,\Gamma} \left(S^{b-B_i,k-K_i} \right) \qquad \text{Lemma 19}$$

$$= \sum_{i \in \mathcal{X}_j} \Gamma_i \cdot \inf_{R \in \mathcal{R}(k-K_i)} \Pr_{R,\Gamma} \left(S^{b-B_i,k-K_i} \right) \qquad \text{Eq. (30)}$$

$$\le \sum_{i \in \mathcal{X}_j} \Gamma_i \cdot \inf_{R \in \mathcal{R}} \Pr_{R,\Gamma} \left(S^{b-B_i,k-K_i-k_{\max}} \right) \qquad \text{Induction Hypothesis} \qquad (31)$$

$$\le \sum_{i \in \mathcal{X}_j} \Gamma_i \cdot \Pr_{R_i^*,\Gamma} \left(S^{b-B_i,k-K_i-k_{\max}} \right) \qquad \text{Eq. (29)}$$

$$= \Pr_{R^*,\Gamma}(S^{b,k-k_{\max}}) \qquad \text{Lemma 19}$$

$$= \inf_{R \in \mathcal{R}} \Pr_{R,\Gamma}(S^{b,k-k_{\max}})$$

Hence, we proved the induction step for this case. $\qquad\qquad\square$

6.3.2 Properties of Types

Proof of Lemma 7.

1. Since $\Pr_{R,\Upsilon}(\forall 1 \le \ell \le n : X_\ell = a_\ell) > 0$ we have that for any $1 \le \ell \le n$ it holds that $a_\ell \in \mathcal{X}_j$ with $j = R(a_1, \dots, a_{\ell-1})$. Therefore, by the definitions of the random process $X_n)$ and types, we have the following.

$$\Pr_{R,\Upsilon}(\forall 1 \le \ell \le n : X_\ell = a_\ell) = \Pr_{R,\Upsilon} \left(\forall 1 \le \ell \le n : Y_\ell^{R(a_1,\dots,a_{\ell-1})} = a_\ell \right)$$

$$= \Pi_{\ell=1}^n \Upsilon_{a_\ell}$$

$$= \exp \left(\sum_{i \in \mathcal{X}} n \cdot T_i \log \Upsilon_i \right)$$

$$= \exp \left(-n \sum_{i \in \mathcal{X}} \cdot T_i \log \frac{1}{\Upsilon_i} \right)$$

2. We note that the claim is essentially trivial. We give the detailed proof for completeness.

   It follows from the definition of types that for any $(a_1, \dots, a_n) \in \mathcal{X}^n$ and $T = \mathcal{T}(a_1, \dots, a_n)$ it holds that $T_i = \frac{|\{\ell | a_\ell = i\}|}{n}$ for any $i \in \mathcal{X}$. Since $\sum_{i \in \mathcal{X}} |\{\ell | \ a_\ell = i\}| = n$, we have $T \in K_n$ as required.

   Let $\phi : K_n \to \{0, \dots, n\}^r$ defined by $\phi(T) = nT$. It is easy to see that the image of ϕ is indeed a subset of $\{0, \dots, n\}^r$ and that ϕ is a one-to-one function. Therefore $|K_n| \le |\{0, \dots, n\}^r| = (n+1)^r$.

3. Let

   $$A = \{(a_1, \dots, a_n) \in \mathcal{X}^n | \mathcal{T}(a_1, \dots, a_n) = T, \ \forall 1 \le \ell \le n : a_\ell \in \mathcal{X}_{R(a_1,\dots,a_{\ell-1})}\}.$$

   We now start proving an upper bound for $|A|$. For $1 \le j \le N$ let $\varphi^j(a_1, \dots, a_n)$ be the result of removing from $(a_1, \dots, a_n)$ all entries which do not belong to $\mathcal{X}_j$. Formally, $\varphi^j : A \to \mathcal{X}_j$ is defined by $\varphi^j(a_1, \dots, a_n) = (a_{\ell_1}, \dots, a_{\ell_h})$, where $\{\ell_1, \dots, \ell_h\} = \{\ell | 1 \le \ell \le n, \ a_\ell \in \mathcal{X}_j\}$ and $\ell_1 < \ell_2 < \dots < \ell_h$. We further define $\varphi(\bar{a}) = \left(\varphi^1(\bar{a}), \varphi^2(\bar{a}), \dots, \varphi^N(\bar{a}) \right)$

with $\varphi : A \to \mathcal{X}_1^* \times \mathcal{X}_2^* \times \ldots \times \mathcal{X}_N^*$. In the following we bound image size of φ, and prove it is an injective function.[11]

Bounding the image. For $1 \leq j \leq N$ define $\lambda_j = \sum_{i \in \mathcal{X}_j} T_i$. The value λ_j can be viewed as the frequency of elements in $\mathcal{X}_j$ in a vector $\bar{a} \in \mathcal{X}^n$ such that $\mathcal{T}(\bar{a}) = T$. Following part 2 of the lemma, it holds that $n\lambda_j$ is integral, and it can be easily observed that $\varphi^j(\bar{a}) \in \mathcal{X}_j^{\lambda_j n}$ for any $1 \leq j \leq N$.

For $1 \leq j \leq N$ such that $\lambda_j \neq 0$ define $T^j \in \mathbb{R}_{\geq 0}^r$ by $T_i^j = \frac{1}{\lambda_j} T_i$ for $i \in \mathcal{X}_j$ and $T_i^j = 0$ for $i \in \mathcal{X} \setminus \mathcal{X}_j$. For $1 \leq j \leq N$ such that $\lambda_j = 0$ define $T^j = \bar{0} \in \mathbb{R}_{\geq 0}^r$. It is easy to verify that if $\lambda_j \neq 0$ then $\mathcal{T}(\varphi^j(\bar{a})) = T^j$ for any $\bar{a} \in A$. Define $C^j = \left\{ \bar{c} \in \mathcal{X}_j^{\lambda_j n} | \mathcal{T}(\bar{c}) = T^j \right\}$. It follows that $\mathrm{Im}(\varphi^j) \subseteq C^j$, and $\mathrm{Im}(\varphi) \subseteq C^1 \times C^2 \times \ldots \times C^N$. It is known that the number of vectors in $\mathcal{X}^{n'}$ of a given type T' is not greater than $\exp\left(-n' \sum_{i \in \mathcal{X}} T_i' \log T_i'\right)$ (Theorem 11.1.3 cf. [13]). Therefore $|C^j| \leq \exp\left(-\lambda_j n \sum_{i \in \mathcal{X}} T_i^j \log T_i^j\right)$. Hence,

$$|\mathrm{Im}(\varphi)| \leq |C^1| \cdot |C^2| \cdot \ldots \cdot |C^N| \leq \exp\left(-n \sum_{j=1}^{n} \lambda_j \sum_{i \in \mathcal{X}} T_i^j \log T_i^j\right). \qquad (32)$$

φ is an injection. Let $(a_1, \ldots, a_n), (d_1, \ldots, d_n) \in A$ with $\varphi(a_1, \ldots, a_n) = \varphi(d_1, \ldots, d_n)$. Assume by negation that $(a_1, \ldots, a_n) \neq (d_1, \ldots, d_n)$. Let ℓ be the minimal index such that $a_\ell \neq d_\ell$. By the definition of A and the choice of ℓ we have $a_\ell, d_\ell \in \mathcal{X}_j$ with $j = R(a_1, \ldots, a_{\ell-1}) = R(d_1, \ldots, d_{\ell-1})$. By the definitions of φ and φ^j it holds that $(a_{\ell_1}, \ldots, a_{\ell_h}) = \varphi^j(a_1, \ldots, a_n) = \varphi(d_1, \ldots, d_n) = (d_{\ell_1'}, \ldots, d_{\ell_{h'}'})$ where $\ell_1, \ldots, \ell_h$ and $\ell_1', \ldots, \ell_{h'}'$ are two monotonically increasing series. Since $(a_1, \ldots, a_n)$ and $(d_1, \ldots, d_n)$ are identical up to the $\ell - 1$ position and $a_\ell, d_\ell \in \mathcal{X}_j$, for some $1 \leq w \leq \ell$ it holds that $\ell_q = \ell_q'$ for $q < w$ and $\ell_w = \ell_w' = \ell$ Therefore $a_\ell = a_{\ell_w} = d_{\ell_w'} = d_\ell$. A contradiction. Thus φ is an injection.

Since φ is an injective function and by (32) we obtain the following.

$$|A| \leq \exp\left(-n \sum_{j=1}^{n} \lambda_j \sum_{i \in \mathcal{X}} T_i^j \log T_i^j\right) = \exp\left(-n \left(\sum_{i \in \mathcal{X}} T_i \log T_i - \sum_{j=1}^{N} \lambda_j \log \lambda_j\right)\right),$$

where the last transition follows from the definitions of λ_j and T_i^j.

---

[11] The *image* of a function $f : X \to Y$ is $\{f(x)| \ x \in X\}$ and denoted $\mathrm{Im}(f)$.

Using the first property of the lemma and the definition of A we have,

$$
\begin{aligned}
\Pr_{R,\Upsilon}(\mathcal{T}(X_1,\ldots,X_n) = T) &= \Pr_{R,\Upsilon}((X_1,\ldots,X_n) \in A) \\
&= \sum_{(a_1,\ldots,a_n)\in A} \Pr_{R,\Upsilon}(\forall 1 \le \ell \le n : X_\ell = a_\ell) \\
&\le \sum_{(a_1,\ldots,a_n)\in A} \exp\left(-n\sum_{i\in\mathcal{X}} T_i \log\frac{1}{\Upsilon_i}\right) \\
&= |A| \cdot \exp\left(-n\sum_{i\in\mathcal{X}} T_i \log\frac{1}{\Upsilon_i}\right) \\
&\le \exp\left(-n\left(\sum_{i\in\mathcal{X}} T_i \log T_i - \sum_{j=1}^{N} \lambda_j \log\lambda_j\right)\right) \cdot \exp\left(-n\sum_{i\in\mathcal{X}} T_i \log\frac{1}{\Upsilon_i}\right) \\
&= \exp\left(-n\sum_{i\in\mathcal{X}} T_i \log\frac{T_i}{\Upsilon_i} + n\sum_{j=1}^{N} \lambda_j \log\lambda_j\right) = \exp(-nD_e(T\|\Upsilon))
\end{aligned}
$$

$\square$

For the proof of Lemma 9 we use the next result (Lemma 11.6.1 cf. [13]).

Lemma 20. *For two distributions $\bar{v}^1, \bar{v}^2 \in \mathbb{R}^n_{\ge 0}$, it holds that*

$$
\frac{1}{2} \cdot \left(\sum_{i=1}^{n} |\bar{v}^1_i - \bar{v}^2_i|\right)^2 \le D\left(\bar{v}^1 \| \bar{v}^2\right).
$$

Proof of Lemma 9. We first define vectors $\bar{t}^1,\ldots,\bar{t}^N$ and $\bar{v}^1,\ldots,\bar{v}^N$, such that $\bar{t}^j, \bar{v}^j \in \mathbb{R}^{r_j}_{\ge 0}$, all the vectors are distributions, T is the concatenation of $\lambda_1\bar{t}^1,\ldots,\lambda_N\bar{t}^N$, and Υ is the concatenation of $\bar{v}^1,\ldots,\bar{v}^N$.

Formally, for $1 \le j \le N$ and $1 \le i \le r_j$ set $\bar{v}^j_i = \Upsilon_{s_{j-1}+i}$ (recall that $s_j = \sum_{k=1}^{j} r_k$). For $1 \le j \le N$ such that $\lambda_j \ne 0$ set $\bar{t}^j_i = \frac{1}{\lambda_j} T_{s_{j-1}+i}$ for any $1 \le i \le r_j$. For $1 \le j \le N$ such that $\lambda_j = 0$ set $\bar{t}^j$ to be an arbitrary distribution. By definition, for any $1 \le j \le N$ and $1 \le i \le r_j$ it holds that $\lambda_j\bar{t}^j_i = T_{s_{j-1}+i}$ and $\bar{v}^j_i = \Upsilon_{s_{j-1}+i}$.

Using the above notation we have the following.

$$
\begin{aligned}
D_e(T\|\Upsilon) &= \sum_{i\in\mathcal{X}} T_i \log\frac{T_i}{\Upsilon_i} - \sum_{j=1}^{N} \lambda_j \log\lambda_j \\
&= \sum_{j=1}^{N}\sum_{i=1}^{r_j} \lambda_j\bar{t}^j_i \log\frac{\lambda_j\bar{t}^j_i}{\bar{v}^j_i} - \sum_{j=1}^{N} \lambda_j\left(\sum_{i=1}^{r_j} \bar{t}^j_i\right)\log\lambda_j \\
&= \sum_{j=1}^{N} \lambda_j\sum_{i=1}^{r_j} \bar{t}^j_i \log\frac{\bar{t}^j_i}{\bar{v}^j_i} \\
&= \sum_{j=1}^{N} \lambda_j D\left(\bar{t}^j\|\bar{v}^j\right) \\
&\ge \sum_{j=1}^{N} \lambda_j\frac{1}{2}\left(\sum_{i=1}^{r_j}\left|\bar{t}^j_i - \bar{v}^j_i\right|\right)^2
\end{aligned}
\tag{33}
$$

Also, by Jensen's inequality (see, e.g., Theorem 2.6.2 in [13]), we have

$$\sum_{j=1}^{N} \lambda_j \left(\sum_{i=1}^{r_j} \left| \bar{t}_i^j - \bar{v}_i^j \right| \right)^2 \geq \left(\sum_{j=1}^{N} \lambda_j \sum_{i=1}^{r_j} \left| \bar{t}_i^j - \bar{v}_i^j \right| \right)^2 \tag{34}$$

Indeed, let $z_j = \sum_{i=1}^{r_j} |\bar{t}_i^j - \bar{v}_i^j|$, and consider W, Z to be random variables such that $\Pr(W = j) = \lambda_j$ and $Z = z_W$. Then

$$\sum_{j=1}^{N} \lambda_j \left(\sum_{i=1}^{r_j} \left| \bar{t}_i^j - \bar{v}_i^j \right| \right)^2 = E[Z^2] \geq (E[Z])^2 = \left(\sum_{j=1}^{N} \lambda_j \sum_{i=1}^{r_j} \left| \bar{t}_i^j - \bar{v}_i^j \right| \right)^2.$$

Using inequalities (33) and (34) we have

$$\sum_{j=1}^{N} \sum_{i \in \mathcal{X}_j} |T_i - \lambda_j \Upsilon_i| = \sum_{j=1}^{N} \sum_{i=1}^{r_j} |\lambda_j \bar{t}_i^j - \lambda_j \bar{v}_i^j|$$

$$= \sum_{j=1}^{N} \lambda_j \sum_{i=1}^{r_j} |\bar{t}_i^j - \bar{v}_i^j|$$

$$\leq \sqrt{\sum_{j=1}^{N} \lambda_j \left(\sum_{i=1}^{r_j} |\bar{t}_i^j - \bar{v}_i^j| \right)^2}$$

$$\leq 2\sqrt{D_e(T\|\Upsilon)}$$

$\square$

6.3.3 Proof of Lemma 13

In this section we give the proof for Lemma 13. Most of the proof is dedicated to showing the following limit.

$$\lim_{k \to \infty} \inf_{R \in \mathcal{R}(k)} \frac{1}{k} \log \Pr_{R,Q} \left(S^{\alpha k, k} \right) = 0. \tag{35}$$

Once the above limit is established, arguments similar to those in the proof of Lemma 10 are used to complete the proof.

One implications of Lemma 10 is that for any $\varepsilon > 0$ it holds that

$$\lim_{k \to \infty} \inf_{R \in \mathcal{R}} \frac{1}{k} \log \Pr_{R,Q} \left(S^{\alpha(1+\varepsilon)k, k} \right) = 0. \tag{36}$$

Our approach to show (35) uses the above limit. Conceptually, we focus in the analysis on events in which some fixed size prefix of the process provides a good cost to coverage ratio, namely, $\frac{\mu(X_1, \ldots, X_m)}{\kappa(X_1, \ldots, X_m)} \leq (\alpha - \eta)$ for some $\eta > 0$. Conditioned on such events, the event $S^{\alpha k, k}$ becomes equivalent to $S^{(\alpha+\varepsilon)k', k'}$ with respect to the process $(X_n)_{n=m+1}^{\infty}$. This allows us to use (36). While the proof is based on the above intuition, it is more involved, as we need to handle several corner cases.

We first partition the possible values of j, $1 \leq j \leq N$, as follows. Define the *strict* values by

$$\mathcal{S} = \{1 \leq j \leq N \mid \forall i \in \mathcal{X}_j : \text{if } Q_i > 0 \text{ then } \alpha K_i = B_i\}$$

and the *non-strict* values by

$$\mathcal{N} = \{1, \ldots, N\} \setminus \mathcal{S} = \{1 \leq j \leq N \mid \exists i \in \mathcal{X}_j : Q_i > 0 \text{ and } \alpha K_i > B_i\}.$$

The last equality follows from the definition of $\bar{q}^j$ in (18), as well as $Q_{s_{j-1}+i} = \bar{q}_i^j$ for $1 \leq j \leq N$ and $1 \leq i \leq r_j$.

With a slight abuse of notation, we also use $\mathcal{N}$ as a function from $\mathcal{X}^*$ to $\left(\bigcup_{j \in \mathcal{N}} \mathcal{X}_j \right)^*$. Given $(a_1, \ldots, a_n) \in \mathcal{X}^*$, we define $\mathcal{N}(a_1, \ldots, a_n)$ to be $(a_1, \ldots, a_n)$ after removing entries in $\bigcup_{j \in \mathcal{S}} \mathcal{X}_j$. Formally, $\mathcal{N}(a_1, \ldots, a_n) = (a_{\ell_1}, \ldots, a_{\ell_m})$, where $\ell_1 < \ell_2 < \ldots < \ell_m$ and $\{\ell| \ a_\ell \in \bigcup_{j \in \mathcal{N}} \mathcal{X}_j\} = \{\ell_1, \ldots, \ell_m\}$.

For every $j \in \mathcal{N}$ set $d_j \in \mathcal{X}_j$ such that $\alpha K_{d_j} > B_{d_j}$ and $Q_{d_j} > 0$. For $j \in \mathcal{S}$ set $d_j \in \mathcal{X}_j$ such that $Q_{d_j} > 0$. It follows from the above that there is $\eta > 0$ such that $B_{d_j} \leq (\alpha - \eta) K_{d_j}$ for every $j \in \mathcal{N}$.

Our analysis focuses on events in which $(X_1, \ldots, X_n)$ (for some n) are in the following sets. For $k, \nu \in \mathbb{N}_+$ define

$$M^{\nu,k} = \left\{ (a_1, \ldots, a_n) \in \mathcal{X}^* \ \middle| \ \begin{array}{ll} \forall 1 \leq \ell \leq n : Q_{a_\ell} > 0 & \text{and} \\ \mathcal{N}(a_1, \ldots, a_n) \subseteq \{d_j | j \in \mathcal{N}\}^* & \text{and} \\ \kappa(a_1, \ldots, a_{n-1}) < k & \text{and} \\ \kappa(\mathcal{N}(a_1, \ldots, a_{n-1})) < \nu & \text{and} \\ (\kappa(a_1, \ldots, a_n) \geq k \text{ or } \kappa(\mathcal{N}(a_1, \ldots, a_n)) \geq \nu) \end{array} \right\}$$

Further, define $M^{\nu,k} = \{\epsilon\}$ for $\nu \in \mathbb{Z}$ if $\nu \leq 0$ or $k \leq 0$. It is easy to show that for every $(a_1, \ldots, a_n) \in M^{\nu,k}$ it holds that $n \leq k$; therefore, $M^{\nu,k}$ is always finite. A useful property of vectors in $M^{v,k}$ is that their costs can be bounded by their coverage. That is, for every $\bar{a} \in M^{b,k}$ it holds that

$$\mu(\bar{a}) \leq \alpha \kappa(\bar{a}) - \eta \kappa(\mathcal{N}(\bar{a})) \tag{37}$$

Similar to the sets $A^{b,k}$ (see Section 6.3.1), the sets $M^{\nu,k}$ admit a recursive structure. It follows from the definitions that, for every $\nu, k \in \mathbb{N}_+$,

$$\begin{aligned} M^{\nu,k} = &\left\{ (d_j, a_1, \ldots, a_n) \ \middle| \ j \in \mathcal{N}, \ (a_1, \ldots, a_n) \in M^{\nu - K_{d_j}, k - K_{d_j}} \right\} \\ &\cup \left\{ (i, a_1, \ldots, a_n) \ \middle| \ j \in \mathcal{S}, \ i \in \mathcal{X}_j, \ Q_i > 0, \ (a_1, \ldots, a_n) \in M^{\nu, k - K_i} \right\}. \end{aligned} \tag{38}$$

Also, similar to $A^{b,k}$, for any two vectors $(a_1, \ldots, a_n), (c_1, \ldots, c_m) \in M^{\nu,k}$ such that $(a_1, \ldots, a_n) \neq (c_1, \ldots, c_m)$, we have that

$$\{(X_1, \ldots, X_n) = (a_1, \ldots, a_n)\} \cap \{(X_1, \ldots, X_m) = (c_1, \ldots, c_m)\} = \emptyset. \tag{39}$$

Let $Q_{\min} = \min_{i \in \mathcal{X}: Q_i > 0} Q_i$.

Lemma 21. *For every $k \in \mathbb{Z}$, $\nu \in \mathbb{Z}$ and $R \in \mathcal{R}$ it holds that*

$$\sum_{(a_1, \ldots, a_n) \in M^{\nu,k}} \Pr_{R,Q}((X_1, \ldots, X_n) = (a_1, \ldots, a_n)) \geq (Q_{\min})^{\max\{\nu, 0\}}.$$

Proof. We prove the claim by induction on k. For $k \leq 0$ it holds that $M^{\nu,k} = \{\epsilon\}$, therefore

$$\sum_{(a_1, \ldots, a_n) \in M^{\nu,k}} \Pr_{R,Q}((X_1, \ldots, X_n) = (a_1, \ldots, a_n)) = 1 \geq (Q_{\min})^{\max\{\nu, 0\}}.$$

For $k > 0$, let $\nu \in \mathbb{Z}$ and $R \in \mathcal{R}$. Assume $\nu > 0$ and let $j = R(\epsilon)$. If $j \in \mathcal{N}$ then, following

(38) and Lemma 18, we have

$$\sum_{(a_1,\ldots,a_n)\in M^{\nu,k}} \Pr_{R,Q}((X_1,\ldots,X_n)=(a_1,\ldots,a_n))$$

$$\geq \sum_{(a_1,\ldots,a_n)\in M^{\nu-K_{d_j},k-K_{d_j}}} \Pr_{R,Q}((X_1,\ldots,X_{n+1})=(d_j,a_1,\ldots,a_n))$$

$$= Q_{d_j} \sum_{(a_1,\ldots,a_n)\in M^{\nu-K_{d_j},k-K_{d_j}}} \Pr_{R_{d_j},Q}((X_1,\ldots,X_n)=(a_1,\ldots,a_n))$$

$$\geq Q_{d_j}\cdot Q_{\min}^{\max\{\nu-K_{d_j},0\}} \geq Q_{\min}^{\max\{\nu,0\}}$$

The second inequality uses the induction hypothesis. By the definition of d_j, it holds that $K_{d_j}\geq 1$.

If $j\in\mathcal{S}$ then, using again (38) and Lemma 19, we have

$$\sum_{(a_1,\ldots,a_n)\in M^{\nu,k}} \Pr_{R,Q}((X_1,\ldots,X_n)=(a_1,\ldots,a_n))$$

$$\geq \sum_{i\in\mathcal{X}_j:\ Q_i>0}\ \sum_{(a_1,\ldots,a_n)\in M^{\nu,k-K_i}} \Pr_{R,Q}((X_1,\ldots,X_{n+1})=(i,a_1,\ldots,a_n))$$

$$= \sum_{i\in\mathcal{X}_j:\ Q_i>0} Q_i \sum_{(a_1,\ldots,a_n)\in M^{\nu,k-K_i}} \Pr_{R_i,Q}((X_1,\ldots,X_n)=(a_1,\ldots,a_n))$$

$$\geq \sum_{i\in\mathcal{X}_j:\ Q_i>0} Q_i\cdot Q_{\min}^{\max\{\nu,0\}} \geq Q_{\min}^{\max\{\nu,0\}}$$

The second inequality uses the induction hypothesis. The last equality uses the fact that $\sum_{i\in\mathcal{X}_j:\ Q_i>0} Q_i = \sum_{i\in\mathcal{X}_j} Q_i = 1$.

It remains to handle the case where $\nu\leq 0$. However, in this case $M^{\nu,k}=\{\epsilon\}$; thus,

$$\sum_{(a_1,\ldots,a_n)\in M^{\nu,k}} \Pr_{R,Q}((X_1,\ldots,X_n)=(a_1,\ldots,a_n)) = 1 \geq (Q_{\min})^{\max\{\nu\},0}\ .$$

$\square$

Lemma 22. *For every $k\in\mathbb{N}$ and $R\in\mathcal{R}(k)$ it holds that $\Pr_{R,Q}(S^{\alpha k,k})\geq Q_{min}^k$.*

Proof. Let $(a_1,\ldots,a_n)\in M^{k,k}$. It follows from the definition of $M^{\nu,k}$ that $\kappa(a_1,\ldots,a_n)\geq k$ and $\kappa(a_1,\ldots,a_{n-1})<k$. Let $j=R(a_1,\ldots,a_{n-1})$ and assume $\Pr_{R,Q}((X_1,\ldots,X_n)=(a_1,\ldots,a_n))>0$. Then, $a_n\in\mathcal{X}_j$. As R is k-consistent, it follows that $K_{a_n}\leq k-\kappa(a_1,\ldots,a_{n-1})$. Hence,

$$\kappa(a_1,\ldots,a_n)=\kappa(a_1,\ldots,a_{n-1})+K_{a_n}\leq k.$$

Thus, we have that

$$\kappa(a_1,\ldots,a_n)=k.$$

Using (37), we also have

$$\mu(a_1,\ldots,a_n)\leq\alpha\mu(a_1,\ldots,a_n)=\alpha k.$$

.

We conclude that if $(a_1,\ldots,a_n)\in M^{k,k}$ and $\Pr_{R,Q}((X_1,\ldots,X_n)=(a_1,\ldots,a_n))>0$ then $\mu(a_1,\ldots,a_n)\leq\alpha k$ and $\kappa(a_1,\ldots,a_n)\geq k$. It follows that

$$Pr_{R,Q}(S^{\alpha k,k})\geq\sum_{(a_1,\ldots,a_n)\in M^{k,k}} Pr_{R,Q}\left((X_1,\ldots X_n)=(a_1,\ldots,a_n)\right)\geq(Q_{\min})^k,$$

where the last inequality is due to Lemma 21.

$\square$

Lemma 23.

$$\liminf_{k\to\infty} \inf_{R\in\mathcal{R}(k)} \frac{1}{k}\log\mathrm{Pr}_{R,Q}\left(S^{\alpha k,k}\right) = 0$$

Proof. We use a more compact notation for the proof of the lemma. Given $\bar{a} = (a_1,\ldots,a_n) \in \mathcal{X}^n$ define $|\bar{a}| = n$. We use $\bar{a} \oplus \bar{c}$ to denote the concatenation of two vectors $\bar{a} = (a_1,\ldots,a_n) \in \mathcal{X}^n$ and $\bar{c} = (c_1,\ldots,c_m) \in \mathcal{X}^m$. That is, $\bar{a} \oplus \bar{c} = (a_1,\ldots,a_n,c_1,\ldots,c_m)$.

We generalize the definition of R_i for $R \in \mathcal{R}$ as given in Section 6.3.1. Given $\bar{c} \in \mathcal{X}^*$, define $R_{\bar{c}} \in \mathcal{R}$ by $R_{\bar{c}}(\bar{a}) = R(\bar{c} \oplus \bar{a})$. It is easy to verify that if $R \in \mathcal{R}(k)$ then $R_{\bar{c}} \in \mathcal{R}(k-\kappa(\bar{c}))$. By iteratively applying Lemma 19, it is easy to show that for any $\bar{a},\bar{c} \in \mathcal{X}^*$ and $R \in \mathcal{R}$,

$$\mathrm{Pr}_{R,Q}((X_1,\ldots,X_{|\bar{a}\oplus\bar{c}|}) = \bar{c}\oplus\bar{a}) = \mathrm{Pr}_{R,Q}((X_1,\ldots,X_{|\bar{c}|}) = \bar{c}) \cdot \mathrm{Pr}_{R_{\bar{c}},Q}((X_1,\ldots,X_{|\bar{a}|}) = \bar{a}). \quad (40)$$

Let $\varepsilon > 0$. We use the sets $A^{b,k}$ as defined in Section 6.3.1 (see (26)). We first show that, for $k \in \mathbb{N}$,

$$\left\{\bar{c}\oplus\bar{a}\middle|\ \bar{c}\in M^{\varepsilon k,k}, \kappa(c)\le k, \bar{a}\in A^{(\alpha+\varepsilon\eta)(k-\kappa(\bar{c})),k-\kappa(\bar{c})}\ \right\} \subseteq A^{\alpha k,k} \quad (41)$$

Let $\bar{c} = (c_1,\ldots,c_m) \in M^{\varepsilon k,k}$ with $\kappa(\bar{c}) \le k$ and $\bar{a} = (a_1,\ldots,a_n) \in A^{(\alpha+\varepsilon\eta)(k-\kappa(\bar{c})),k-\kappa(\bar{c})}$.

If $\kappa(\bar{c}) \ge k$ then $\kappa(\bar{c}) = k$ and $\bar{a} = \epsilon$. By (37) it holds that $\mu(\bar{c}) \le \alpha k$, and by the definition of $M^{\varepsilon k,k}$, $\kappa(c_1,\ldots,c_{m-1}) < k$. Therefore, $\bar{c} = \bar{c}\oplus\bar{a} \in A^{\alpha k,k}$.

If $\kappa(\bar{c}) < k$ then $\kappa(\mathcal{N}(\bar{c})) \ge \varepsilon k$. Therefore, using (37), it holds that

$$\mu(\bar{c}) \le \alpha\kappa(\bar{c}) - \eta\kappa(\mathcal{N}(\bar{c})) \le \alpha\kappa(\bar{c}) - \eta\varepsilon k.$$

Hence,

$$\mu(\bar{c}\oplus\bar{a}) \le \alpha\kappa(\bar{c}) - \eta\varepsilon k + (\alpha+\varepsilon\eta)(k-\kappa(\bar{c})) \le \alpha k.$$

Also, by the definitions, we have that $\kappa(\bar{c}\oplus\bar{a}) \ge k$ and $\kappa(c_1,\ldots,c_m,a_1,\ldots,a_{n-1}) < k$. Therefore (41) holds.

Finally, we note that if $(c_1,\ldots,c_m) = \bar{c} \in M^{\varepsilon k,k}$, $\kappa(\bar{c}) \ge k$ and $\mathrm{Pr}_{R,Q}((X_1,\ldots,X_{|c|}) = \bar{c}) > 0$, for some $R \in \mathcal{R}(k)$, then $\kappa(\bar{c}) = k$. If the conditions for the claim hold then we have $\kappa(c_1,\ldots,c_{m-1}) < k$ and $c_m \in \mathcal{X}_j$ for $j = R(c_1,\ldots,c_{m-1})$. As R is k-consistent, we have that $K_{c_m} \le k - \kappa(c_1,\ldots,c_{m-1})$; therefore, $\kappa(c_1,\ldots,c_m) \le k$.

It follows that, for any $k \in \mathbb{N}$ and $R \in \mathcal{R}(k)$,

$$\mathrm{Pr}_{R,Q}\left(S^{\alpha k,k}\right) = \mathrm{Pr}_{R,Q}\left(\exists \bar{a}\in A^{\alpha k,k}: (X_1,\ldots,X_{|\bar{a}|}) = \bar{a}\right)$$

$$\ge \mathrm{Pr}_{R,Q}\left(\exists\ \begin{matrix} \bar{c}\in M^{\varepsilon k,k}, \kappa(\bar{c})\le k \\ \bar{a}\in A^{(\alpha+\varepsilon\eta)(k-\kappa(\bar{c})),k-\kappa(\bar{c})} \end{matrix} : \begin{matrix} (X_1,\ldots,X_{|\bar{c}|}) = \bar{c} \\ (X_{|\bar{c}|+1},\ldots,X_{|\bar{c}|+|\bar{a}|}) = \bar{a} \end{matrix}\right)$$

$$= \sum_{\bar{c}\in M^{\varepsilon k,k}, \kappa(\bar{c})\le k}\ \sum_{\bar{a}\in A^{(\alpha+\varepsilon\eta)(k-\kappa(\bar{c})),k-\kappa(\bar{c})}} \mathrm{Pr}_{R,Q}\left((X_1,\ldots,X_{|\bar{c}|}) = \bar{c}\right)\mathrm{Pr}_{R_{\bar{c}},Q}\left((X_1,\ldots,X_{|\bar{a}|}) = \bar{a}\right)$$

$$= \sum_{\bar{c}\in M^{\varepsilon k,k}, \kappa(\bar{c})\le k} \mathrm{Pr}_{R,Q}\left((X_1,\ldots,X_{|\bar{c}|}) = \bar{c}\right)\sum_{\bar{a}\in A^{(\alpha+\varepsilon\eta)(k-\kappa(\bar{c})),k-\kappa(\bar{c})}} \mathrm{Pr}_{R_{\bar{c}},Q}\left((X_1,\ldots,X_{|\bar{a}|}) = \bar{a}\right)$$

$$= \sum_{\bar{c}\in M^{\varepsilon k,k}} \mathrm{Pr}_{R,Q}\left((X_1,\ldots,X_{|\bar{c}|}) = \bar{c}\right) \cdot \mathrm{Pr}_{R_{\bar{c}},Q}\left(S^{(\alpha+\varepsilon\eta)(k-\kappa(\bar{c})),k-\kappa(\bar{c})}\right)$$

The first and last equality use (27). The second equality is by (40) and the observation that the events considered are disjoint. By (36), there is $L > 0$ such that, for any $k > L$ and $R' \in \mathcal{R}$,

$$\mathrm{Pr}_{R',Q}\left(S^{(\alpha+\varepsilon\eta)k,k}\right) \ge \exp(-k\varepsilon).$$

Also, by Lemma 22, for any $k \in \mathbb{N}$, $k \le L$ and $R' \in \mathcal{R}(k)$, it holds that

$$\mathrm{Pr}_{R',Q}\left(S^{(\alpha+\varepsilon\eta)k,k}\right) \ge \mathrm{Pr}_{R',Q}\left(S^{\alpha k,k}\right) \ge \exp\left(L\log Q_{\min}\right).$$

Combining the above, we have that

$$\liminf_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\Pr_{R,Q}\left(S^{\alpha k,k}\right)$$

$$\geq\liminf_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\left(\sum_{\bar{c}\in M^{\varepsilon k,k}}\Pr_{R,Q}\left((X_1,\ldots,X_{|\bar{c}|})=\bar{c}\right)\cdot\Pr_{R_{\bar{c}},Q}\left(S^{(\alpha+\varepsilon\eta)(k-\kappa(\bar{c})),k-\kappa(\bar{c})}\right)\right)$$

$$\geq\liminf_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\left(\sum_{\bar{c}\in M^{\varepsilon k,k}}\Pr_{R,Q}\left((X_1,\ldots,X_{|\bar{c}|})=\bar{c}\right)\cdot\exp(-\varepsilon k)\right)$$

$$=-\varepsilon+\liminf_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\left(\sum_{\bar{c}\in M^{\varepsilon k,k}}\Pr_{R,Q}\left((X_1,\ldots,X_{|\bar{c}|})=\bar{c}\right)\right)$$

$$\geq-\varepsilon+\liminf_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\left(Q_{\min}^{\varepsilon k}\right)=-\varepsilon-\varepsilon\log\frac{1}{Q_{\min}}.$$

Since the last inequality holds for any $\varepsilon>0$, we have

$$\liminf_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\Pr_{R,Q}\left(S^{\alpha k,k}\right)\geq0.$$

Also, as

$$\limsup_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\Pr_{R,Q}\left(S^{\alpha k,k}\right)\leq0,$$

we conclude that

$$\lim_{k\to\infty}\ \inf_{R\in\mathcal{R}(k)}\frac{1}{k}\log\Pr_{R,Q}\left(S^{\alpha k,k}\right)=0.$$

$\square$

Proof of Lemma 13. Let $k\in\mathbb{N}_+$ and $R\in\mathcal{R}(k)$. By the definition of $S^{\alpha k,k}$, and since $\mu(a_1,\ldots,a_n)\geq n$ for any $(a_1,\ldots,a_n)\in\mathcal{X}^n$, we have

$$\Pr_{R,Q}\left(S^{\alpha k,k}\right)\leq\sum_{n=1}^{\lfloor\alpha k\rfloor}\Pr_{R,Q}\left(\mu(X_1,\ldots,X_n)\leq\alpha k\text{ and }\kappa(X_1,\ldots,X_n)\geq k\right).$$

Therefore, there is $1\leq n^{k,R}\leq\alpha k$ for which

$$\frac{\Pr_{R,Q}\left(S^{\alpha k,k}\right)}{\alpha k}\leq\Pr_{R,Q}\left(\mu(X_1,\ldots,X_{n^{k,R}})\leq\alpha k\text{ and }\kappa(X_1,\ldots,X_{n^{k,R}})\geq k\right).$$

We can write the above in terms of types:

$$\frac{\Pr_{R,Q}\left(S^{\alpha k,k}\right)}{\alpha k}\leq\Pr_{R,Q}\left(\mu(X_1,\ldots,X_{n^{k,R}})\leq\alpha k\text{ and }\kappa(X_1,\ldots,X_{n^{k,R}})\geq k\right)$$

$$=\Pr_{R,Q}\left(n\cdot\mathcal{T}(X_1,\ldots,X_{n^{k,R}})\cdot B\leq\alpha k\text{ and }n\cdot\mathcal{T}(X_1,\ldots,X_{n^{k,R}})\cdot K\geq k\right)$$

$$=\sum_{\substack{T\in K_{n^{k,R}}:\\n^{k,R}T\cdot K\geq k,\\n^{k,R}T\cdot B\leq\alpha k}}\Pr_{R,Q}\left(\mathcal{T}(X_1,\ldots,X_{n^{k,R}})=T\right).$$

Since $|K_{n^{k,R}}|\leq(n^{k,R}+1)^r\leq(\alpha k+1)^r$ and $\Pr_{R,Q}\left(S^{\alpha k,k}\right)>0$ (by Lemma 22), there is $T^{k,R}\in K_{n^{k,R}}$ such that $n^{k,R}T^{k,R}\cdot K\geq k$, $n^{k,R}T^{k,R}\cdot B\leq\alpha k$, and

$$\frac{\Pr_{R,Q}\left(S^{\alpha k,k}\right)}{(\alpha k+1)^{r+1}}\leq\Pr_{R,Q}\left(\mathcal{T}(X_1,\ldots,X_{n^{k,R}})=T^{k,R}\right).$$

Once $n^{k,R}$ and $T^{k,R}$ are defined, it remains to show the three properties in the lemma. Property 3 holds by the definition.

From Lemma 23, we have

$$\liminf_{k\to\infty} \inf_{R\in\mathcal{R}(k)} \frac{1}{k} \log \Pr_{R,Q}\left(\mathcal{T}(X_1,\ldots,X_{n^{k,R}}) = T^{k,R}\right)$$

$$\geq \liminf_{k\to\infty} \inf_{R\in\mathcal{R}(k)} \frac{1}{k} \log\left(\frac{\Pr_{R,Q}(S^{\alpha k,k})}{(\alpha k + 1)^{r+1}}\right) = 0.$$

To show Property 1, we note that since

$$\limsup_{k\to\infty} \inf_{R\in\mathcal{R}(k)} \frac{1}{k} \log \Pr_{R,Q}\left(\mathcal{T}(X_1,\ldots,X_{n^{k,R}}) = T^{k,R}\right) \leq 0$$

we have

$$\lim_{k\to\infty} \inf_{R\in\mathcal{R}(k)} \frac{1}{k} \log \Pr_{R,Q}\left(\mathcal{T}(X_1,\ldots,X_{n^{k,R}}) = T^{k,R}\right) = 0, \tag{42}$$

as required.

We now show Proerty 2. For any $1 \leq j \leq N$, $k \in \mathbb{N}$ and $R \in \mathcal{R}(k)$, define $\lambda_j^{k,R} = \sum_{i\in\mathcal{X}_j} T_i^{k,R}$. Then using Lemmas 9 and 7, we have

$$\limsup_{k\to\infty} \sup_{R\in\mathcal{R}(k)} \sum_{j=1}^{N} \sum_{i\in\mathcal{X}_j} \left|T_i^{k,R} - \lambda_j^{k,R} Q_i\right| \leq \limsup_{k\to\infty} \sup_{R\in\mathcal{R}(k)} 2\sqrt{D_e\left(T^{k,R}\|Q\right)}$$

$$\leq \limsup_{k\to\infty} \sup_{R\in\mathcal{R}(k)} 2\sqrt{\frac{1}{n^{k,R}} \cdot \log\left(\Pr_{R,Q}\left(\mathcal{T}(X_1,\ldots,X_{n^{k,R}}) = T^{k,R}\right)\right)}$$

$$\leq \limsup_{k\to\infty} \sup_{R\in\mathcal{R}(k)} 2\sqrt{-\frac{k_{\max}}{k} \cdot \log\left(\Pr_{R,Q}\left(\mathcal{T}(X_1,\ldots,X_{n^{k,R}}) = T^{k,R}\right)\right)} = 0,$$

where the last equality is since $n^{k,R}T \cdot K \geq k$. It follows that $n^{k,R} \geq \frac{k}{k_{\max}}$ (recall $k_{\max} = \max_{i\in\mathcal{X}} K_i$). The last equality is by (42). This completes the proof of the lemma. $\square$

7 Discussion

In this paper we introduced a new technique for obtaining parameterized approximation algorithms leading to significant improvements in running times over existing algorithms. The analysis of our algorithms required the development of a mathematical machinery for the analysis of a wide class of two-variable recurrence relations. Following the above results, several issues remain open:

- From theoretical perspective, it is desirable to obtain deterministic variants of our algorithms. Derandomizing our technique is left for future work.

- Sanov's theorem also falls into the category of Large Deviation Theory. The theorem has multiple extensions and variations from the viewpoint of theoretical probability theory. One of the most general of these is Gartner-Ellis theorem [22, 17] (see a unified claim in [24]). By using this theorem, some specific steps within the proof of Theorem 2 may be skipped. We keep these steps to make the proof clearer and more accessible to readers outside the above areas.

- The most similar work we found on recurrence relations with two or more variables is due to Eppstein [18]. The recurrence relations considered in [18] are different from the relations considered in this paper. However, intuitively, the two classes of recurrences

seem to be related. It would be interesting to establish such a relation, which may lead to a statement similar to the one of Theorem 2 for the recurrence relations considered in [18].

- We considered a family of two-variable recurrence relations. It seems possible to extend the results to multivariate relations, such as

$$p(b_1, \ldots, b_m, k) = \min_{1 \leq j \leq N : \bar{k}^j \leq k} \sum_{i=1}^{r_j} \bar{\gamma}_i^j \cdot p(b_1 - B_{i,1}^j, \ldots, b_m - B_{i,m}^j, k - \bar{k}_i^j),$$

with the initial conditions $p(b_1, \ldots, b_m, k) = 0$ if $b_\ell < 0$ for some $1 \leq \ell \leq m$, and $p(b_1, \ldots, b_m, 0) = 1$ if $b_1, \ldots, b_m \geq 0$. However, we are not aware of any interesting applications for such generalizations.

- In the definition of composite recurrence as given in (2), only values of j such that $\bar{k}^j \leq k$ are considered in the min operation. One way to eliminate this requirement is to define a relation $q : \mathbb{Z} \times \mathbb{Z} \to [0, 1]$ by,

$$q(b, k) = \min_{1 \leq j \leq N} \sum_{i=1}^{r_j} \bar{\gamma}_i^j \cdot q(b - \bar{b}_i^j, k - \bar{k}_i^j)$$
$$q(b, k) = 0 \qquad\qquad\qquad\qquad \forall b < 0, k \in \mathbb{Z}$$
$$q(b, 0) = 1 \qquad\qquad\qquad\qquad \forall b \geq 0, k \geq 0$$

An analog of Theorem 2 can be established for q stating that

$$\forall \varepsilon > 0 : \liminf_{k \to \infty} \frac{1}{k} \log q((\alpha + \varepsilon)k, k) \geq -M \quad \text{and} \quad \limsup_{k \to \infty} \frac{1}{k} \log q(\alpha k, k) \leq -M,$$

where $M = \max_{1 \leq j \leq N} M_j$, and M_j is the α-branching number of $(\bar{b}^j, \bar{k}^j, \bar{\gamma}^j)$ (as in Definition 1).

- Often the analysis of branching algorithms uses complex recurrence relations which involve two functions or more to obtain improved bounds on running times. Examples for such analyses can be found in [11] and [20]. When transformed to the context of randomized branching, the analyses yield recurrence relations in two functions, such as

$$p(b, k) = \min \begin{cases} 0.5 \cdot p(b - 1, k - 1) + 0.5 \cdot q(b - 2, k) \\ 0.5 \cdot p(b - 1, k) + 0.25 \cdot q(b - 2, k) + 0.25 \cdot q(b - 2, k - 2) \end{cases}$$

$$q(b, k) = \min \begin{cases} 0.5 \cdot p(b - 1, k - 1) + 0.5 \cdot q(b - 3, k) \\ 0.5 \cdot p(b - 1, k) + 0.25 \cdot q(b - 3, k) + 0.25 \cdot q(b - 3, k - 3) \end{cases}$$

We are currently unable to analyze the asymptotic behavior of such relations. Such an analysis however, is likely to lead to an improved parameterized approximation for small values of α (for both Vertex Cover and 3-Hitting Set), as the algorithms in [11] and [20] have better running times as exact algorithms, compared to the running time of our algorithms for approximation ratios approaching 1.

Currently, the parameterized algorithm for Vertex Cover with the best running time is due to [12]. We were unable to obtain a randomized branching variant for this algorithm. One reason is that an incorrect branching can lead to an unbounded increase in the mininmal vertex cover size.

- We showed the application of randomized branching to Vertex Cover and to 3-Hitting Set. While we believe that the technique can be used for other problems, such as *Feedback Vertex Set*, *Total Vertex Cover* and *Edge Dominating Set*, we leave such results for future works.

  Initial experiments for Feedback Vertex Set led to a parameterized random 1.5-approximation with running time $O^* \left(\left(\frac{16}{9} \right)^k \right) = O^*(1.778^k)$. The algorithm is a fairly naive and builds on the randomized $O^*(4^k)$ algorithm of [5].

References

[1] A. Agrawal and S. Boyd. Disciplined quasiconvex programming. *Optimization Letters*, 2020.

[2] N. Alon and J. H. Spencer. *The Probabilistic Method*. Wiley Publishing, 4th edition, 2016.

[3] N. Amenta, M. Bern, and D. Eppstein. Optimal point placement for mesh smoothing. *Journal of Algorithms*, 30(2):302 – 322, 1999.

[4] N. Bansal, P. Chalermsook, B. Laekhanukit, D. Nanongkai, and J. Nederlof. New tools and connections for exponential-time approximation. *Algorithmica*, 81(10):3993–4009, 2019.

[5] A. Becker, R. Bar-Yehuada, and D. Geiger. Random algorithms for the loop cutset problem. In *UAI-99*, pages 49–56, Stockholm, Sweden,, 1999.

[6] R. Beigel and D. Eppstein. 3-coloring in time o(1.3289n). *Journal of Algorithms*, 54(2):168 – 204, 2005.

[7] E. Bonnet, B. Escoffier, E. J. Kim, and V. T. Paschos. On subexponential and fpt-time inapproximability. *Algorithmica*, 71(3):541–565, Mar 2015.

[8] N. Bourgeois, B. Escoffier, and V. T. Paschos. Approximation of max independent set, min vertex cover and related problems by moderately exponential algorithms. *Discrete Appl. Math.*, 159(17):1954–1970, 2011.

[9] L. Brankovic and H. Fernau. Parameterized approximation algorithms for hitting set. In *WAOA 2011*, pages 63–76, Saarbrcken, Germany, 2012.

[10] L. Brankovic and H. Fernau. A novel parameterised approximation algorithm for minimum vertex cover. *Theoretical Computer Science*, 511:85 – 108, 2013. Exact and Parameterized Computation.

[11] J. Chen, I. A. Kanj, and W. Jia. Vertex cover: Further observations and further improvements. *Journal of Algorithms*, 41(2):280 – 301, 2001.

[12] J. Chen, I. A. Kanj, and G. Xia. Improved upper bounds for vertex cover. *Theoretical Computer Science*, 411(40):3736 – 3756, 2010.

[13] T. M. Cover and J. A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, New York, NY, USA, second edition, 2006.

[14] I. Csiszar. The method of types [information theory]. *IEEE Transactions on Information Theory*, 44(6):2505–2523, Oct 1998.

[15] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer Publishing Company, Incorporated, 1st edition, 2015.

[16] A. Drucker, J. Nederlof, and R. Santhanam. Exponential Time Paradigms Through the Polynomial Time Lens. In *ESA 2016*, volume 57, pages 36:1–36:14, Aarhus, Denmask, 2016.

[17] R. S. Ellis. Large deviations for a general class of random vectors. *The Annals of Probability*, 12(1):1–12, 1984.

[18] D. Eppstein. Quasiconvex analysis of multivariate recurrence equations for backtracking algorithms. *ACM Trans. Algorithms*, 2(4):492–509, Oct. 2006.

[19] M. R. Fellows, A. Kulik, F. A. Rosamond, and H. Shachnai. Parameterized approximation via fidelity preserving transformations. *J. Comput. Syst. Sci.*, 93:30–40, 2018.

[20] H. Fernau. Parameterized algorithmics for d-hitting set. *International Journal of Computer Mathematics*, 87(14):3157–3174, 2010.

[21] F. V. Fomin, F. Grandoni, and D. Kratsch. A measure & conquer approach for the analysis of exact algorithms. *J. ACM*, 56(5):25:1–25:32, 2009.

[22] J. Grtner. On large deviations from the invariant measure. *Theory of Probability & Its Applications*, 22(1):24–39, 1977.

[23] W. Hoeffding. Asymptotically optimal tests for multinomial distributions. *Ann. Math. Statist.*, 36(2):369–401, 04 1965.

[24] F. d. Hollander. *Larege Deviations*. American Mathematical Society, USA, 2000.

[25] S. Khot and O. Regev. Vertex cover might be hard to approximate to within 2-ϵ. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008.

[26] D. Lokshtanov, F. Panolan, M. S. Ramanujan, and S. Saurabh. Lossy kernelization. In *STOC 2017*, pages 224–237, New York, NY, USA, 2017.

[27] D. Lokshtanov, M. S. Ramanujan, and S. Saurabh. Linear time parameterized algorithms for subset feedback vertex set. *ACM Trans. Algorithms*, 14(1):7:1–7:37, 2018.

[28] P. Manurangsi and L. Trevisan. Mildly exponential time approximation algorithms for vertex cover, balanced separator and uniform sparsest cut. In *APPROX/RANDOM'18*, pages 20:1–20:17, 2018.

[29] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

[30] R. Niedermeier and P. Rossmanith. Upper bounds for vertex cover further improved. In C. Meinel and S. Tison, editors, *STACS 99*, pages 561–570, 1999.

[31] I. N. Sanov. On the probability of large deviations of random variables. *Matematicheskii Sbornik*, 42:11–44. In Russian. English translation in: Selected Translations in Mathematical Statistics and Probability I, pages 213–244, 1961.

[32] M. Wahlström. *Algorithms, Measures and Upper Bounds for Satisfiability and Related Problems*. PhD thesis, Department of Computer and Information Science, Linköpings University, Sweden, 2007.