# Efficient Computation of Representative Weight Functions with Applications to Parameterized Counting

Daniel Lokshtanov[*] Saket Saurabh[†] Meirav Zehavi[‡]

**Abstract**

In this paper we prove an analogue of the classic Bollobás lemma for *approximate counting*. In fact, we match an analogous result of Fomin *et al.* [JACM 2016] for *decision*. This immediately yields, for a number of fundamental problems, parameterized approximate counting algorithms with the same running times as what is obtained for the decision variant using the representative family technique of Fomin *et al.* [JACM 2016]. For example, we devise an algorithm for approximately counting (a factor $(1 \pm \epsilon)$ approximation algorithm) $k$-paths in an $n$-vertex directed graph (#$k$-Path) running in time $\mathcal{O}((2.619^k + n^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot (n+m))$. This improves over an earlier algorithm of Brand *et al.* [STOC 2018] that runs in time $\mathcal{O}(4^k \cdot k^{\mathcal{O}(1)} \cdot \frac{1}{\epsilon^2} \cdot (n+m))$.

Additionally, we obtain an approximate counting analogue of the efficient computation of representative families for product families of Fomin *et al.* [TALG 2017], again essentially matching the running time for decision. This results in an algorithm with running time $\mathcal{O}((3.841^k + |I|^{o(1)}) \cdot \frac{1}{\epsilon^6} \cdot |I|)$ for computing a $(1+\epsilon)$ approximation of the sum of the coefficients of the multilinear monomials in a degree-$k$ homogeneous $n$-variate polynomial encoded by a monotone circuit (#Multilinear Monomial Detection). When restricted to monotone circuits (rather than polynomials of non-negative coefficients), this improves upon an earlier algorithm of Pratt [FOCS 2019] that runs in time $4.075^k \cdot \frac{1}{\epsilon^2} \log \frac{1}{\epsilon} \cdot n^{\mathcal{O}(1)}$.

[*] University of California, Santa Barbara, USA. `daniello@ucsb.edu`

[†] The Institute of Mathematical Sciences, HBNI, Chennai, India, and University of Bergen, Norway. `saket@imsc.res.in`

[‡] Ben-Gurion University, Beersheba, Israel. `meiravze@bgu.ac.il`

# 1 Introduction and Overview

The seminal paper of Valiant on counting problem [Val79] showed that although Perfect Matching is solvable in polynomial time, #Perfect Mathicng is unlikely to be. This paper has since sparked vast interest in the study of counting problems. In this paper, we consider counting problems from the lens of Parameterized Complexity [CFK$^+$15, DF13, FG06]. Our objective is twofold.

- Devise a general purpose algorithmic tool for parameterized counting problems.
- Use this tool to design state-of-the-art algorithms for several counting problems, including #$k$-Path.

The subfield of Parameterized Counting Complexity was initiated by Flum and Grohe [FG04], as early as 2002. Thus, this subfield has been around for the last 18 years, but until recently it has remained largely unexplored, with exceptions that are few and far between [AR02, Kou08, KW16b]. The last few years have seen a flurry of activities in this area resulting in the development of new tools and settlement of some old problems [Cur13, CM14, CDM17, CX15, BDH18, RW20, Bra19, DLM20]. We refer to the survey by Curticapean [Cur18] for a detailed exposition to parameterized counting problems.

As is the case with classical complexity, most of the natural counting problems are #W[1]-hard [FG04] in the realm of Parameterized Complexity, which means that they are unlikely to be solvable in time $f(k)n^{\mathcal{O}(1)}$ for any computable function $f$ of $k$. A problem admitting an algorithm with running time $f(k)n^{\mathcal{O}(1)}$ is called *fixed parameter tractable (FPT)* and the running time of the form $f(k)n^{\mathcal{O}(1)}$ is called FPT-time. For example, Flum and Grohe [FG04] showed that counting $k$-sized distinct (simple) paths in an undirected or directed graph (#$k$-Path) is #W[1]-hard [FG04], although the decision version can be solved in FPT-time. In fact, until this day #$k$-Path is considered the most classical example of a problem solvable in FPT-time but which is #W[1]-hard. Further, Flum and Grohe [FG04] conjectured the same for counting $k$-sized matchings (#$k$-Matching) even on bipartite graphs. Curticapean [Cur13] and Curticapean and Marx [CM14] settled the parameterized complexity of #$k$-Matching by showing that the problem is #W[1]-hard.

The intractability of counting problems leads to the question of *approximately counting* in FPT-time. In particular, there is a long history of FPT-approximation schemes (FPT-ASs), that is, $f(k,\epsilon^{-1})n^{\mathcal{O}(1)}$-time algorithms that approximate the number of certain combinatorial objects in the given input. Specifically, an FPT-AS for the #$k$-Path problem has been around for almost two decades [AR02] and is one of the fundamental problems driving the field of parameterized counting problems [AR02, ADH$^+$08, AG10, BDH18, BLSZ19]. Recently, an approach based on representative families has been successful in the design of FPT-time algorithms for a wide-range of problems including $k$-Path (the decision version of #$k$-Path), thus it is natural to consider a counting notion analogous to this notion. However, even just the existence of "small" representative families for counting purposes has not been known. In this paper, we develop a new technology that both asserts their existence and shows how to compute them efficiently.

## 1.1 Representative Functions (or Counters) and Applications

Our starting point is the notion of *representative families* [Mon85, Mar09]. Let $U$ be a universe and let $\mathcal{S} = \{S_1, \ldots, S_t\}$ be a family of subsets of $U$ of size $p$. A subfamily $\widehat{\mathcal{S}} \subseteq \mathcal{S}$ is $q$-*representative* for $\mathcal{S}$ if for every set $Y \subseteq U$ of size at most $q$, if there is a set $X \in \mathcal{S}$ disjoint from $Y$, then there exists a set $X \in \widehat{\mathcal{S}}$ disjoint from $Y$. By the classical combinatorial result of Bollobás, every family of sets of size $p$ has a $q$-*representative family* with at most $\binom{p+q}{p}$

Table 1: History of $\#k$-Path

| Ref. | Time | Technique | Det. | Extension |
|---|---|---|---|---|
| [AR02] | $k^{\mathcal{O}(k)}n^{\mathcal{O}(1)}$ | Karp-Luby | No | Treewidth $\mathcal{O}(1)$ |
| [ADH+08] | $(2e)^k n^{\mathcal{O}(1)}$ | Color-Coding | No | No Extension |
| [AG09] | $(2e)^{k+o(k)}n^{\mathcal{O}(1)}$ | Color-Coding | Yes | Treewidth $\mathcal{O}(1)$ |
| [BDH18] | $4^k n^{\mathcal{O}(1)}$ | Exterior Algebra | No | Pathwidth $\mathcal{O}(1)$ |
| [Pra19] | $4.075^k n^{\mathcal{O}(1)}$ | Waring Rank | No | Treewidth $\mathcal{O}(1)$ |
| [BLSZ19] | $4^{k+o(k)}n^{\mathcal{O}(1)}$ | Divide & Color | Yes | Treewidth $\mathcal{O}(1)$ |
| **This Paper** | $2.619^k n^{\mathcal{O}(1)}$ | **Representative Counters** | **No** | **Treewidth $\mathcal{O}(1)$** |

sets [Bol65]. Given a family $\mathcal{S}$ of sets of size $p$, and an integer $q$, an efficient algorithm computing a $q$-representative family $\widehat{\mathcal{S}} \subseteq \mathcal{S}$ was given in [Mar06, Mar09, FLPS16]. The fact that $\widehat{\mathcal{S}}$ can be efficiently computed from $\mathcal{S}$ (and its generalizations to representative matroids) has found numerous applications in Parameterized and Exact Algorithms [Mon85, Mar09, FLPS16, SZ16, FGPS19, FLPS17, KW12, KS17, Mar06].

In this paper we prove an analogue of this result for *approximate counting*. More precisely, a function $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ where $\mathcal{P} \subseteq \binom{U}{p}$ (such a function is called a *counter*) is said to $(\epsilon, q)$-*represent* a function $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ with respect to $\mathcal{Q} \subseteq \binom{U}{q}$ if for every set $Q \in \mathcal{Q}$, the following condition is satisfied: $\sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \mathfrak{C}(P) \simeq (1 \pm \epsilon) \cdot \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P)$. We prove that, when $\mathcal{P}$ and $\mathcal{Q}$ are "nice" (where the definition of "nice" is just the product of a minor technicality that can be ignored at the moment and to which we will return later), given any function $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$, a function $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ that $(\epsilon, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$ and whose support (denoted by $\mathsf{supp}$) size is $\binom{k}{p} \cdot 2^{o(k)} \cdot \frac{1}{\epsilon^2} \cdot n^{o(1)}$ where $k = p + q$ and $n = |U|$, can be computed with success probability arbitrarily close to 1 and in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot (\frac{k}{q})^q \cdot 2^{o(k)} \cdot \frac{1}{\epsilon^2} \cdot n^{1+o(1)})$. We demonstrate how the efficient construction of representative functions can be a powerful tool in designing parameterized algorithms for counting problems.

### 1.1.1 Applications

The $k$-Path problem (on both directed and undirected graphs) is among the most extensively studied parameterized problems [CFK+15, FLSZ19]. This problem has played a pivotal role in the development of Parameterized Complexity and has led to several new tools and techniques in the area such as *color-coding* [AYZ95], *divide & color* [CKL+09b], *algebraic methods* [KW16b, BKKZ17, Wil09] and *representative families* [Mon85, Mar09, FLPS16]. After a long sequence of works in the past three decades, the current best known parameterized algorithms for $k$-Path have running times $1.657^k n^{\mathcal{O}(1)}$ (randomized, polynomial space, undirected only) [BHKK17, Bjö14] (extended in [BKKZ17]), $2^k n^{\mathcal{O}(1)}$ (randomized, polynomial space) [Wil09], $2.554^k n^{\mathcal{O}(1)}$ (deterministic, exponential space) [Tsu19, Zeh15, FLPS16, SZ16], and $4^{k+o(k)}n^{\mathcal{O}(1)}$ (deterministic, polynomial space) [CKL+09a].

Similarly to $k$-Path, the counting analogue $\#k$-Path plays a significant role in the development of the field of parameterized counting. More than 15 years ago, Arvind and Raman [AR02] utilized the classic method of *color coding* [AYZ95] and *Karp-Luby approximate counting* technique to design a *randomized* exponential-space FPT-AS for $\#k$-Path with running time $k^{\mathcal{O}(k)}n^{\mathcal{O}(1)}$ whenever $\epsilon^{-1} \leq k^{\mathcal{O}(k)}$. A few years afterwards, the development and use of applications in computational biology to detect and analyze *network motifs* have already become common practice [SIKS06, SSRS06, SI06, DSG+08, HWZ08]. Roughly speaking, a network motif is a small pattern whose number of occurrences in a given network is substantially larger than its number of occurrences in a random network. Due to their tight relation to network motifs, #$k$-Path and other cases of the #Subgraph Isomorphism problem became highly relevant to the study of gene transcription networks, protein-protein interaction (PPI) networks, neural networks and social networks [MSOI$^+$02]. In light of these developments, Alon et al. [ADH$^+$08] revisited the method of color coding to attain a running time whose dependency on $k$ is *single-exponential* rather than *slightly super-exponential*. Specifically, they designed a *simple randomized* $\mathcal{O}((2e)^k m\epsilon^{-2})$-time exponential-space FPT-AS for #$k$-Path, which they employed to analyze PPI networks of unicellular organisms. In particular, their algorithm has running time $2^{\mathcal{O}(k)}m$ whenever $\epsilon^{-1} \leq 2^{\mathcal{O}(k)}$. The first *deterministic* FPT-AS for #$k$-Path was found in 2007 by Alon and Gutner [AG10]; this algorithm has an exponential space complexity and running time $2^{\mathcal{O}(k\log\log k)}m\log n$ whenever $\epsilon^{-1} = 2^{o(\log k)}$. Shortly afterwards, Alon and Gutner [AG09] improved upon their previous work, and designed a deterministic exponential-space FPT-AS for #$k$-Path with running time $(2e)^{k+\mathcal{O}(\log^3 k)}m\log n$ whenever $\epsilon^{-1} = k^{\mathcal{O}(1)}$. For close to a decade, this algorithm has remained the state-of-the-art. In 2016, Koutis and Williams [KW16a] made the following conjecture.

**Conjecture:** #$k$-Path admits an FPT-AS with running time $2^k(\frac{1}{\epsilon})^{\mathcal{O}(1)}n^{\mathcal{O}(1)}$.

After a decade, in 2018, Brand et al. [BDH18] provided a speed-up towards the resolution of this conjecture. Specifically, they gave an algebraic *randomized* $\mathcal{O}(4^k m\epsilon^{-2})$-time exponential-space algorithm. This was followed up by Björklund et al. [BLSZ19] who gave a *deterministic algorithm* with almost similar running time. However, this algorithm is still far away from resolving the conjecture of Koutis and Williams [KW16a].

As our first application we give an algorithm for #$k$-Path that runs in time $\mathcal{O}((2.619^k + n^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot (n+m))$. This results brings the gap between the known algorithm and the conjecture close. While on a superficial level, we make use of the notion of parsimonious universal families also present in [BLSZ19], our new result is centred around the efficient computation of representative counter functions (a concept introduced in this paper), which requires to develop a whole new machinery in general, and sampling primitives in particular, on which we elaborate in Section 1.3.

The #$k$-Path problem is a special case of the #$k$-Subgraph Isomorphism problem, where for a given $n$-vertex graph $G$ and a given $k$-vertex graph $F$, the objective is to count the number of distinct subgraphs of $G$ that are isomorphic to $F$. In addition to #$k$-Path, parameterized counting algorithms for two other variants of #$k$-Subgraph Isomorphism, when $F$ is a tree, and more generally, a graph of treewidth at most $t$, were studied in the literature. The algorithm of Björklund et al. [BLSZ19] can be extended for these cases with running time similar to that for #$k$-Path. Independently, Pratt [Pra19] obtained an algorithm for these cases as an application of his algorithm for a more general problem, called #Multilinear Detection, which we discuss in more detail in the following subsection.

In particular, we obtain Theorem 1.1 ahead as an application of our first tool. Before we state it, let us give the definitions of the problems it addresses. In $q$-Set $p$-Packing we are given a universe $U$, a family $\mathcal{F}$ of subsets of size $q$ of $U$, and $p \in \mathbb{N}$. Then, the objective is to determine whether there exist at least $p$ pairwise-disjoint sets in $\mathcal{F}$. In $q$-Dimensional $p$-Matching, we are given a universe $U$, a partition $(U_1, U_2, \ldots, U_q)$ of $U$, a family $\mathcal{F}$ of subsets of size $q$ of $U$ where each subset contains exactly one element from each part $U_i$, and $p \in \mathbb{N}$. Then, the objective is to determine whether there exist at least $p$ pairwise-disjoint sets in $\mathcal{F}$. In Graph Motif, we are given a graph $G$ where each vertex is assigned a set of colors, a multiset of colors $M$, and $k \in \mathbb{N}$ (the sought motif size). Then, the objective is to determine whether there exist a subtree $T$ of $G$ on $k$ vertices and a coloring of the vertices in $T$ (each by a color from its set) so that no color is used more times than its number of occurrences in $M$.

**Theorem 1.1.** *For any* $0 < \epsilon < 1$*, the* #$k$-Path*,* #$q$-Set $p$-Packing *with* $k = qp$*,* #$q$-Dimensional $p$-Matching *with* $k = (q-1)p$ *and* #Graph Motif *with* $k$ *being twice the sought motif size problems can be approximated with factor* $(1 \pm \epsilon)$ *and success probability at least* $\frac{9}{10}$ *in time* $\mathcal{O}((2.619^k + |I|^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot |I|)$*, where* $k$ *is the parameter and* $|I|$ *is the input size. Moreover, for any* $0 < \epsilon < 1$*, the* #$k$-Tree *(or, more generally,* #Subgraph Isomorphism *where the treewidth of pattern graph is bounded by a fixed constant) can be approximated with factor* $(1 \pm \epsilon)$ *and success probability at least* $\frac{9}{10}$ *in time* $2.619^k \cdot \frac{1}{\epsilon^2} \cdot |I|^{\mathcal{O}(1)}$*.*

## 1.2 Representation for Product Functions (or Counters) and Applications

Let $\mathcal{P} \subseteq \binom{U}{p}$. Given two functions $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ where $\mathcal{P}_1 \subseteq \binom{U}{p_1}$, $\mathcal{P}_2 \subseteq \binom{U}{p_2}$ and $p_1 + p_2 = p$, the *product* $\mathfrak{C}_1 \times \mathfrak{C}_2$ *(with respect to* $\mathcal{P}$*)* is the function $\mathfrak{C}_1 \times \mathfrak{C}_2 : \mathcal{P} \to \mathbb{N}_0$ defined as follows: For each $P \in \mathcal{P}$,

$$(\mathfrak{C}_1 \times \mathfrak{C}_2)(P) = \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2:\\ P_1 \cap P_2 = \emptyset, P_1 \cup P_2 = P}} \mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2).$$

We prove that, given that $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}$ and $\mathcal{Q} \subseteq \binom{U}{q}$ are “nice”, given any two functions $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$, a function $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ that $(\epsilon, q)$-represents $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$ with respect to $\mathcal{Q}$ and whose support size is $\binom{k}{p} \cdot 2^{o(k)} \cdot \frac{1}{\epsilon^2} \cdot n^{o(1)}$ where $k = p + q$, can be computed with success probability arbitrarily close to 1 and in time

$$\mathcal{O}\left(\left(3.841^k + |\mathsf{supp}(\mathfrak{C}_1)| \cdot (\frac{k}{q+p_2})^{q+p_2} + |\mathsf{supp}(\mathfrak{C}_2)| \cdot (\frac{k}{q+p_1})^{q+p_1}\right) \cdot 2^{o(k)} \cdot \frac{1}{\epsilon^2} \cdot n^{1+o(1)}\right).$$

A more exact expression of the upper bound on the time complexity that precisely describes the dependence on the sizes of the supports of $\mathfrak{C}_1$ and $\mathfrak{C}_2$ rather than the term $3.841^k$ is given in the paper. However, the crux here is that the time complexity to compute the output function can be substantially smaller than even just the time to explicitly write up the function $\mathfrak{C}_1 \times \mathfrak{C}_2$ that it represents (even if both $\mathfrak{C}_1 \times \mathfrak{C}_2$ have already been reduced to have support size $\binom{k}{p_1}$ and $\binom{k}{p_1}$)! For example, if both $p_1$ and $p_2$ are close to $k/2$, then the support of their product is already $4^k$.

Our main application is a randomized algorithm for the #Multilinear Monomial Detection problem, mentioned in the previous subsection. In this problem, the objective is to compute a $(1+\epsilon)$ approximation of the sum of the coefficients of the multilinear monomials in a degree-$k$ homogeneous $n$-variate polynomial encoded by an arithmetic circuit with nonnegative coefficients (i.e., a *monotone circuit*). Recently, Pratt [Pra19] developed a randomized $(1+\epsilon)$-approximation algorithm for this problem with time complexity $\mathcal{O}(4.075^k \cdot \frac{1}{\epsilon^2} \log \frac{1}{\epsilon} \cdot s(C)^{\mathcal{O}(1)})$. In fact, the result of Pratt [Pra19] is for a notably more general—it deals with the #Multilinear Monomial Detection problem extended to only requiring the polynomial to have nonnegative coefficients, thus allowing the arithmetic circuit to have negative coefficients, though not the polynomial that it encodes. Improving upon this result for the case of monotone circuits, we get the following.

**Theorem 1.2.** *For any* $0 < \epsilon < 1$ *, the* #Multilinear Monomial Detection *problem (on monotone circuits) can be approximated with factor* $(1 \pm \epsilon)$ *and success probability at least* $\frac{9}{10}$ *in time* $\mathcal{O}((3.841^k + s(C)^{o(1)}) \cdot \frac{1}{\epsilon^6} \cdot s(C))$*.*

The decision version of #Multilinear Monomial Detection is the central problem in the algebraic approach of Koutis and Williams for designing fast parameterized algorithms [Kou08, KW16b, Wil09]. Here, the objective is to decide whether there exists a multilinear monomial of degree-$k$ with non-zero coefficient (rather than to compute the sum of coefficients of such monomilas). Let $s(C)$ denote the size of $C$. Williams [Wil09] gave a *randomized* algorithm solving $k$-Multilinear Monomial Detection in time $2^k \cdot s(C)^{\mathcal{O}(1)}$ (over monotone circuits). The only known algorithm for the problem when there is no restriction on circuits is by Brand et al. [BDH18], who gave an algorithm with running time $4.32^k \cdot s(C)^{\mathcal{O}(1)}$ (with exponential space complexity). (Recently, further (yet unpublished) developments were given in the preprint [BP20].) Afterwards, Arvind et al. [ACDM19] obtained an algorithm with the same running time and with polynomial space complexity. The algorithms based on the algebraic method of Koutis-Williams provide a dramatic improvement for a number of fundamental problems. See the survey by Koutis and Williams [KW16a] for further details. The idea behind the approach is to translate a given problem into the language of algebra by reducing it to the problem of deciding whether a constructed polynomial has a multilinear monomial of degree $k$.

We note that #$k$-Subgraph Isomorphism can be reduced to the #Multilinear Monomial Detection problem and thus one can obtain an algorithm (that is efficient when the sought graph is of constant treewidth) for it as an application of Theorem 1.2. In fact, #$k$-Subgraph Isomorphism reduces to #Multilinear Monomial Detection on specials circuits where we can obtain a faster algorithm. This is what is exploited in the proof of Theorem 1.1. More precisely, the aforementioned special circuits are "$d$-skewed circuits" (mostly, for $d = \mathcal{O}(1)$), where every multiplication gate has at most one child whose polynomial can consist of more than $d$ monomials. Specifically, we have the following theorem, where we are particularly interested in the case where $\ell = 0$. This theorem is also our intermediate step to derive Theorem 1.1.

**Theorem 1.3.** *For any $0 < \epsilon < 1$ and $\ell \in \mathbb{N}_0$, the* #Multilinear Monomial Detection *problem on $2^{o(k)} s(C)^\ell$-skewed circuits can be approximated with factor $(1 \pm \epsilon)$ and success probability at least $\frac{9}{10}$ in time $\mathcal{O}((2.619^k + s(C)^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot s(C)^{\ell+1})$.*

## 1.3 Our Methods

In this subsection, we give a short overview of our proof and the tools we develop along the way, which are of broader interest. Most of the overview concerns our computation of representative counters (in the simpler, "non-product" setting), and it is structured as follows. We first extend the definition of special families of sets called *parsimonious universal families* to have so called *membership queries*. We then present an alternative definition of representation that assumes the presence of such extended families, and argue that this alternative definition is equivalent to our original definition of representation (which is also the one used for applications) up to a minor error. After that, we turn to show how to compute representative counters according to this alternative definition. To this end, we present a sampling procedure, which, given a counter and a parsimonious universal family, samples (while using the membership queries of the family) some sets from the support of the counter and assigns new values to them. We then argue that this procedure, with high probability, yields a representative counter. After this, we still have the task of computing a parsimonious universal family with membership queries. Unfortunately, we do not know how to efficiently compute such a *small* family with *efficient* membership queries. So, we define a weaker but more technical notion based on a partition of the universe. We then present a sampling procedure that, with high probability, succeeds in computing the required family. Afterwards, we argue that the weaker notion suffices for our purposes. Lastly in this overview, we briefly address the case of product counters, where computation of a representative counter is substantially more technically involved. After that, we also briefly address our applications.

**Representation via Similarity with Respect to an Approximately Parsimonious Universal Family.** Our computation of representative functions assumes access to so called *approximately parsimonious families* with efficient *membership procedures*, defined as follows. Let $n, p, q \in \mathbb{N}$ and $0 < \epsilon < 1$. Let $U$ be a universe of size $n$, and let $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. A family $\mathcal{F} \subseteq 2^U$ is an *$\epsilon$-parsimonious $(n, p, q)$-universal family with respect to* $(\mathcal{P}, \mathcal{Q})$ if there exists $T = T(n, p, q, \epsilon) > 0$, called a *correction factor*, such that for each pair of disjoint sets $P \in \mathcal{P}$ and $Q \in \mathcal{Q}$, it holds that $(1 - \epsilon) \cdot T \leq |\mathcal{F}[P, Q]| \leq (1 + \epsilon) \cdot T$ where $\mathcal{F}[P, Q] = \{F \in \mathcal{F} : P \subseteq F, Q \cap F = \emptyset\}$. A *$\widehat{T}$-membership query procedure* is a procedure that given any set $P \in \mathcal{P}$ as input, outputs the subfamily $\{F \in \mathcal{F} : P \subseteq F\}$ in time $\mathcal{O}(\widehat{T})$. The development of a procedure to compute small parsimonious families with efficient membership query procedures is part of our paper, and it is of independent interest. We briefly discuss this part later in the overview.

Our first key insight towards the computation of representative functions is to project the notion of similarity between functions to approximately parsimonious families. To this end, we extend the domains of functions as follows. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a function. The *extender* $\mathfrak{C}_{\text{ext}} : 2^U \to \mathbb{N}_0$ is defined as follows. For any set $F \subseteq U$, let $\mathfrak{C}_{\text{ext}}(F) \triangleq \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \mathfrak{C}(P)$. Notice that for any set $P \in \mathcal{P} \subseteq \binom{U}{p}$, we have that $\mathfrak{C}_{\text{ext}}(P) = \mathfrak{C}(P)$. Then, we say that a function $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ and a function $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ are *$(\epsilon, \mathcal{F})$-similar* (with respect to a family $\mathcal{F} \subseteq 2^U$), if for every set $F \in \mathcal{F}$, $(1 - \epsilon) \cdot \mathfrak{C}_{\text{ext}}(F) \leq \widehat{\mathfrak{C}}_{\text{ext}}(F) \leq (1 + \epsilon) \cdot \mathfrak{C}_{\text{ext}}(F)$. Roughly, the advantage of the definition of similarity compared to representation is that representation requires to explicitly consider every set $P \in \mathcal{P}$ and every set $Q \in \mathcal{Q}$, while similarity requires to explicitly consider only every set $P \in \mathcal{P}$ (indirectly via the definition of the extenders of the functions). The consideration of every set $P \in \mathcal{P}$ is not as "intimidating" as the consideration of every set $Q \in \mathcal{Q}$, as among the sets in $\mathcal{P}$ we only care about those that belong to the support of the function which we want to represent and that is part of the input, while about the sets in $\mathcal{Q}$ we do not know anything and they are not part of the input (so, it may be that $|\mathcal{Q}|$ is of a prohibitive magnitude of $\binom{n}{q}$)! Specifically, we prove the following lemma.

**Lemma 1.1.** *Let $n, p, q \in \mathbb{N}$, $0 < \epsilon < 1$ and $0 < \delta < 1$. Let $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathcal{F} \subseteq 2^U$ be an $\epsilon$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ and $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ be $(\delta, \mathcal{F})$-similar. Then, $\widehat{\mathfrak{C}}$ $(4\epsilon + \delta, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$.*

**Sampling the Support of the Input Function and Re-Adjusting the Assigned Values.** Our computation of representative functions is done in a sampling procedure defined as follows. (Some explanation of the intuition behind it is given ahead.)

**Definition 1.1** (**$(\mathfrak{C}, \mathcal{F})$-Counter Sampling**). *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$. Let $p, L \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$. Then, $(\mathfrak{C}, \mathcal{F}, L)$-*counter sampling *is the randomized procedure that constructs $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ as follows. For any set $P \in \mathcal{P}$, define*

$$\mathsf{assoc}_{\mathfrak{C}, \mathcal{F}, L}(P) \triangleq \min_{F \in \mathcal{F} : P \subseteq F} \mathfrak{C}_{\text{ext}}(F),$$

$$\mathsf{prob}_{\mathfrak{C}, \mathcal{F}, L}(P) \triangleq \min(1, L \cdot \frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C}, \mathcal{F}, L}(P)}), \text{and}$$

$$\mathsf{count}_{\mathfrak{C}, \mathcal{F}, L}(P) \triangleq \frac{\mathfrak{C}(P)}{\mathsf{prob}_{\mathfrak{C}, \mathcal{F}, L}(P)}.$$

*Then, for any set $P \in \mathcal{P}$, set $\widehat{\mathfrak{C}}(P)$ to $\mathsf{count}_{\mathfrak{C}, \mathcal{F}, L}(P)$ with probability $\mathsf{prob}_{\mathfrak{C}, \mathcal{F}, L}(P)$ and to $0$ with probability $1 - \mathsf{prob}_{\mathfrak{C}, \mathcal{F}, L}(P)$.*[1]

[1] Since $U \in \mathcal{F}$, there exists $F \in \mathcal{F}$ such that $P \subseteq F$, hence $\mathsf{assoc}_{\mathfrak{C}, \mathcal{F}, L}(P)$ is well defined.

The support of any function that can be potentially output is contained in the support of the input function. Essentially, with this sampling procedure we aim to discard as many sets as possible from the support of the input function (so that the output function will have small support), while modifying the values of the sets that are kept so as to "make amends" for all those sets we dropped—so that we obtain a representative function. Intuitively, each set $P \in \mathcal{P}$ is associated, among the sets in $\mathcal{F}$ that contain it and hence whose value is effected by the value of $P$ (as assigned by the function and its extender), with a set $F$ having minimum value. Thus, $P$ is associated with a set $F$ for which $P$ is most significant among all sets in $\mathcal{F}$—that is, in which the fraction of the value of $P$ from the entire value of $F$ is largest. In a sense, this means that the value of $F$ is most "vulnerable" in case $P$ will be dropped from the support of the function. Next, the probability of keeping $P$ in the support is chosen to be proportional to its fraction of value within $F$—the larger $\mathfrak{C}(P)$ is, the larger is the probability to choose it, but at the same time, the larger $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)$ is (which means that the set $F$ associated with $P$, and hence all other sets in $\mathcal{F}$ as well, are less vulnerable to $P$ being dropped out), the smaller is the probability to choose $P$. The factor $L$ (whose exact value will be determined later) is meant to boost up the probability to be larger than just the fraction of the value of $P$ within $F$ (else we may drop "too many" sets from the support, and hence the output function will not represent the input function). Due to this boosting factor, we also need to trim down the boosted fraction to be 1 so that it will indeed represent a probability. Lastly, the new value of $P$ when decided to be kept in the support, is chosen in a way as to ensure that its expected value (being the probability to keep it times its new value when it is kept) will be equal to its original value.

We prove our main theorem, stated below, by utilizing our sampling procedure.

**Theorem 1.4.** *Let $U$ be a universe. Let $0 < \epsilon < 1$, $p, q, c \in \mathbb{N}_0$, $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathcal{F} \subseteq 2^U$ be a $\frac{1}{5}\epsilon$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$, equipped with a $T$-membership query procedure. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$. Then, a function $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ such that*

1. *for every $P \in \mathcal{P}$, $E[\widehat{\mathfrak{C}}(P)] = \mathfrak{C}(P)$, and*
2. *with success probability at least $1 - \frac{1}{c}$, $\widehat{\mathfrak{C}}$ $(\epsilon, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$ and satisfies $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq \mathcal{O}((\frac{1}{\epsilon})^2 |\mathcal{F}| \log c \log(c|\mathcal{F}|))$,*

*can be computed in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot T)$.*

**Computation of Approximate Universal Families with Membership Query Procedures for Balancedly Split Sets.** While it is quite easy to compute a "small" approximate universal family using the probabilistic method, this will yield a family $\mathcal{F}$ where given a set $P \in \binom{U}{p}$, the computation of all sets in the family that contain $P$ may entail iterating over all sets in $\mathcal{F}$! Even if the family $\mathcal{F}$ is just of size $\binom{k}{p}$ (which is clearly a lower bound on the size of $\mathcal{F}$ even if it is not required to be parsimonious, but just required to satisfy $|\mathcal{F}[P, Q]| \geq 1$ for all $P, Q$), this is much too costly for us. Instead, our construction can be viewed as a *data structure* that enables to perform efficient membership queries after initialization (in which it computes an approximately parsimonious family $\mathcal{F}$ in a fashion tailored to make its queries efficient later). The construction is done only for "nice" sets, that is, sets that are roughly balancedly split across some partitioned universe (this is made precise immediately). This is merely a technicality, because a problem can be reduced in a black box fashion to a version considering only nice sets as solution, as we will explain later in this overview.

We now make the exposition of this part more precise. To this end, we need a few notations. Let $k, p \in \mathbb{N}$. A tuple $\overline{\mathbf{U}} = (U_1, , \ldots, U_{\sqrt{k}})$ where $U_1, U_2, \ldots, U_{\sqrt{k}}$ are pairwise-disjoint universes is called a *$\sqrt{k}$-partitioned universe.* Moreover, a function $f : \{1, \ldots, \sqrt{k}\} \to \{0, \ldots, 2\sqrt{k}\}$

satisfying $\sum_{i=1}^{\sqrt{k}} f(i) = k$ is called a *$k$-splitting function.* Lastly, a pair $(f, g)$ of a $k$-splitting function $f : \{1, \ldots, t\} \to \{0, \ldots, 2\sqrt{k}\}$ and a function $g : \{1, \ldots, \sqrt{k}\} \to \{0, \ldots, \sqrt{k}\}$ satisfying $g \leq f$ (i.e., for every $i \in \{1, \ldots, \sqrt{k}\}$, $g(i) \leq f(i)$) and $\sum_{i=1}^{\sqrt{k}} g(i) = p$, is called a *$(k,p)$-splitting function pair.* Then, $P \in \binom{U}{p}$ is *$(\overline{\mathbf{U}}, f, g)$-balancedly split* if for every $i \in \{1, \ldots, \sqrt{k}\}$, it holds that $|P \cap U_i| = g(i)$. Further, $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} \subseteq \binom{U}{p}$ denotes the collection of all $(\overline{\mathbf{U}}, f, g)$-balancedly split sets. Moreover, $Q \in \binom{U}{k-p}$ is *complementary $(\overline{\mathbf{U}}, f, g)$-balancedly split* if for every $i \in \{1, \ldots, \sqrt{k}\}$, it holds that $|Q \cap U_i| = f(i) - g(i)$. Further, $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g} \subseteq \binom{U}{k-p}$ denotes the collection of all complementary $(\overline{\mathbf{U}}, f, g)$-balancedly split sets.

We give the following construction for $\mathcal{F}$ equipped with a procedure to query membership.

**Definition 1.2 (Parsimonious Universal Family Sampling).** *Let $k, p \in \mathbb{N}$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_{\sqrt{k}})$ be a partitioned universe with $U = \bigcup_{i=1}^{\sqrt{k}} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. Then, $(\overline{\mathbf{U}}, f, g, \epsilon, c, d)$-*universal family sampling *is the randomized procedure that constructs a family $\mathcal{F} \subseteq 2^U$ as follows. For $i \in \{1, 2, \ldots, \sqrt{k}\}$ and $j \in \{1, 2, \ldots, s_i\}$ with*

$$s_i = \frac{(d \cdot f(i))^{f(i)}}{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}} \cdot \frac{1}{\widehat{\epsilon}^2} \cdot 10k \cdot \ln(nc),$$

*where $\widehat{\epsilon} = \frac{\ln(1+\epsilon)}{\sqrt{k}}$, construct a set $F_{i,j} \subseteq U_i$ as follows: Each element in $U_i$ is inserted independently with probability $\frac{g(i)}{d \cdot f(i)}$ into $F_{i,j}$. For $i \in \{1, 2, \ldots, \sqrt{k}\}$, denote $\mathcal{F}_i = \{F_{i,j} : j \in \{1, 2, \ldots, s_i\}\}$. Then, $\mathcal{F} = \{F_{1,j_1} \cup F_{2,j_2} \cup \cdots \cup F_{\sqrt{k},j_{\sqrt{k}}} : F_{1,j_1} \in \mathcal{F}_1, F_{i,j_2} \in \mathcal{F}_2, \ldots, F_{i,j_{\sqrt{k}}} \in \mathcal{F}_t\}$.*

**Definition 1.3 (Membership Query Procedure).** *Given $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$, the procedure* MEMBERSHIP *naively computes $\mathcal{F}'_i = \{F_{i,j_i} \in \mathcal{F}_i : P \cap U_i \subseteq F_{i,j_i}\}$ (for every $i \in \{1, \ldots, \sqrt{k}\}$) by iterating over every set in $\mathcal{F}_i$; then, the output is $\{F_{1,j_1} \cup F_{2,j_2} \cup \cdots \cup F_{\sqrt{k},j_{\sqrt{k}}} : F_{1,j_1} \in \mathcal{F}'_1, F_{2,j_2} \in \mathcal{F}'_2, \ldots, F_{\sqrt{k},j_{\sqrt{k}}} \in \mathcal{F}'_{\sqrt{k}}\}$.*

We prove the following theorem.[2]

**Theorem 1.5.** *Let $k, p \in \mathbb{N}$ with $p \leq k$, and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_{\sqrt{k}})$ be a partitioned universe with $U = \bigcup_{i=1}^{\sqrt{k}} U_i$ of size $n \leq c$, and let $(f, g)$ be a splitting function pair. With probability at least $1 - \frac{1}{c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, 2, f, g, \epsilon, c, d)$-universal family sampling, computed in time $\mathcal{O}(|\mathcal{F}|n)$, satisfies the following conditions.*

1. $|\mathcal{F}| \leq \frac{(dk)^k}{p^p(dk-p)^{k-p}} \cdot (\frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc))^{\sqrt{k}}$.

2. *$\mathcal{F}$ is an $\epsilon$-parsimonious $(n, p, k-p)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}, \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g})$.*

3. MEMBERSHIP *is a $T$-membership query procedure where*

$$T = \left((dk)^{\sqrt{k}} + (\frac{dk}{dk-p})^{k-p}(\frac{dk}{p})^{p-p'}\right) \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)\right)^{\sqrt{k}}.$$

[2] We remark that the precise upper bounds given in this theorem, in particular the terms $\frac{(dk)^k}{p^p(dk-p)^{k-p}}$ and $(\frac{dk}{dk-p})^{k-p}(\frac{dk}{p})^{p-p'}$, cannot be loosened if one is to obtain the running times guaranteed for our applications.

**Reduction of a Problem to a Partitioned Version.** Lastly, we give a black box reduction, which may also be of independent interest. Roughly, this reduction shows that if one can approximately count solutions that are balancedly split (which can be an easier task, as in our case), then one can approximately count all solutions. Formally, let $\Pi$ be a problem whose input consists, among possibly other components, of a universe $U$ of size $n$ and $k \in \mathbb{N}$, and whose solutions are subsets (resp. ordered subsets) of $U$ of size $k$. Such a problem $\Pi$ is said to be *splittable*. The *split version of* $\Pi$ is defined as follows. Its input consists of the same components as the input of $\Pi$, and in addition, of a $\sqrt{k}$-partitioned universe $\overline{\mathbf{U}}$ and a $k$-splitting function $f$, and whose solutions are all the subsets (resp. ordered subsets) of $U$ that are both solutions of $\Pi$ and are $(\overline{\mathbf{U}}, f, f)$-balancedly split.

**Lemma 1.2.** *Let $\Pi$ be a splittable problem such that the number of solutions of its split version can be approximately counted with multiplicative error $(1 \pm \alpha)$ in time $T \geq n$ with success probability at least $1 - \frac{1}{c'}$. Then, for any $c \in \mathbb{N}$ such that $(4\sqrt{k})^{\sqrt{k}} \cdot \frac{1}{\beta^2} k \ln(nc) \cdot \frac{1}{c'} \leq \frac{1}{2c}$ and $0 < \beta < 1$, the number of solutions of $\Pi$ can be approximately counted with multiplicative error $(1 \pm \alpha)(1 \pm \beta)$ in time $\mathcal{O}((4\sqrt{k})^{\sqrt{k}} \cdot T \cdot \frac{1}{\beta^2} k \ln(nc))$ with success probability at least $1 - \frac{1}{c}$.*

**Extension to Product Functions.** The computation of a representative function for a product function is technically involved. Among the main difficulties being faced here is the fact that we cannot even iterate over the support of the input product function (since that in itself is too costly) and decide for each set in the support whether to insert it to the support of the output function (with some probability and new assigned value). Instead, we pre-determine how many sets to pick up, and devise a somewhat complex mechanism that allows us to efficiently sample sets from the support according to some distribution without ever computing the support! In particular, we now have two approximately parsimonious families rather than one (where one is meant to separate between sets in $\mathcal{P}$ and sets in $\mathcal{Q}$, and the other is meant to separate between sets in $\mathcal{P}_1$ and sets in $\mathcal{P}_2$), and the sampling is done in three stages after some critical preprocessing to efficiently determine (in part) the probability distributions used in these stages. The first stage involves sampling a set $P_1$ from the support of $\mathfrak{C}_1$, the second (which depends on the outcome of the first) involves sampling a pair of sets from our approximately parsimonious families, and the third (which depends on the outcome of the first and second) involves sampling a set $P_2$ disjoint from $P_1$ from the support of $\mathfrak{C}_2$, so as to pick up $P_1 \cup P_2$. We defer further technical details on the extension to product functions to Section 5.

**Applications.** Our algorithm for #Multilinear Detection on skewed circuits is based on dynamic programming over the nodes of the input circuit. For each node, we store a counter that assigns to each monomial (encoded by the set containing its variables) of the polynomial of the subcircuit rooted at the current node its coefficient with "small error". (More precisely, for each node together with a combination of other arguments, we store one such counter, but for the sake of simplicity of this overview, we ignore these other arguments here.) When we consider a node, we have already computed the aforementioned counters for all its outgoing neighbours. So, as the circuit is skewed, we can explicitly compute the counter for the current node, and then compute a representative counter for it and store the representative counter instead of it (else, even though the circuit is skewed, after several levels just writing the polynomial via a counter explicitly may take time $\binom{n}{k}$). When we reach the root, we can solve the problem.

On general (monotone) circuits we cannot write the polynomial (and hence the counter) of a node that results from the multiplication of the polynomials stored for its outgoing neighbours (within the desired time complexity) even after their sized have already been reduced by representation. So, instead, here we use our computation for product counters that sidesteps this. Having attained algorithms for #Multilinear Detection on skewed and general circuits, all our other applications, including the algorithm for #$k$-Path, follow just by using reductions known in the literature and observing that they are parsimonious.

### 1.4 Additional Related Works

The algorithms by Alon et al. [ADH$^+$08] and Alon and Gutner [AG10, AG09], just like our algorithms, extend to approximate counting of graphs of bounded treewidth. (This remark is also made by Alon and Gutner [AG10, AG09].) In what follows, we briefly review works related to exact counting and decision from the viewpoint of Parameterized Complexity. Since these topics are not the focus of our work, the survey is illustrative rather than comprehensive.

The problem of counting the number of subgraphs of a graph $G$ that are isomorphic to a graph $H$—that is, #Subgraph Isomorphism with Pattern $H$—admits a dichotomy: If the vertex cover number of $H$ is bounded, then it is FPT [WW13], and otherwise it is #W[1]-hard [CM14]. The #W[1]-hardness of #$k$-Path, originally shown by Flum and Grohe [FG04], follows from this dichotomy. By using the "meet in the middle" approach, the #$k$-Path problem and, more generally, #Subgraph Isomorphism with Pattern $H$ where $H$ has bounded *pathwidth* and $k$ vertices, was shown to admit an $n^{\frac{k}{2}+\mathcal{O}(1)}$-time algorithm [BHKK09]. Later, Björklund et al. [BKK17] showed that $\frac{k}{2}$ is not a barrier (which was considered to be the case at that time) by designing an $n^{0.455k+\mathcal{O}(1)}$-time algorithm. A breakthrough that resulted in substantially faster running times took place: Curticapean et al. [CDM17] showed that #Subgraph Isomorphism with Pattern $H$ is solvable in time $\ell^{\mathcal{O}(\ell)}n^{0.174\ell}$ where $\ell$ is the number of edges in $H$; in particular, this algorithm solves #$k$-Path in time $k^{\mathcal{O}(k)}n^{0.174k}$. Recently, Arvind et al. [ACDM19] obtained an algorithm for #Multilinear Monomial Detection with time complexity $n^{k/2+\mathcal{O}(\log k)}$. Also recently, Dell et al. [DLM20] gave "black box" results for turning algorithms which decide whether or not a witness exists into algorithms to approximately count the number of witnesses (with overheads of $k^{\mathcal{O}(k)}$ that are prohibitive for our settings).

## 2 Preliminaries

Let $U$ be a universe, and let $p, q \in \mathbb{N}_0$ be non-negative integers. Then, let $\binom{U}{p}$ be the collection of subsets of $U$ of size exactly $p$, and denote $\binom{U}{\leq p} = \bigcup_{i=0}^{p} \binom{U}{i}$. Given two subsets $P, Q$ of $U$ and a family $\mathcal{F} \subseteq 2^U$ of subsets of $U$, denote $\mathcal{F}[P,Q] \triangleq \{F \in \mathcal{F} : P \subseteq F, Q \cap F = \emptyset\}$. Given a function $f : U \to \mathbb{R}$, let $\mathsf{supp}(f) = \{u \in U : f(u) \neq 0\}$ denote the support of $f$. Given two functions $f : U \to \mathbb{R}$ and $g : U \to \mathbb{R}$ such that for every $a \in A$, it holds that $g(a) \leq f(a)$, we denote $g \leq f$.

A central notion in our proofs is of parsimonious universal families, defined as follows.

**Definition 2.1** (**$\epsilon$-Parsimonious Universal Family, Generalization of [BLSZ19]**). *Let $n, p, q \in \mathbb{N}$ and $0 < \epsilon < 1$. Let $U$ be a universe of size $n$, and let $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. A family $\mathcal{F} \subseteq 2^U$ is an* $\epsilon$-parsimonious $(n,p,q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$ *if there exists $T = T(n,p,q,\epsilon) > 0$, called a* correction factor, *such that for each pair of disjoint sets $P \in \mathcal{P}$ and $Q \in \mathcal{Q}$, it holds that $(1-\epsilon) \cdot T \leq |\mathcal{F}[P,Q]| \leq (1+\epsilon) \cdot T$.*

The special case of Definition 2.1 where $\mathcal{P} = \binom{U}{p}$ and $\mathcal{Q} = \binom{U}{q}$ is the definition of an $\epsilon$-parsiminious universal family in [BLSZ19]. For parsimonious universal families, the following proposition is known, based on a straightforward sampling argument.

**Proposition 2.1** ([BLSZ19]). *Let $c \in \mathbb{N}$ be a fixed constant. Let $n, p, q \in \mathbb{N}$ and $0 < \epsilon < 1$, and denote $k = p + q$. Let $U$ be a universe of size $n$. An $\epsilon$-parsimonious $(n,p,q)$-universal family*

*$\mathcal{F} \subseteq 2^U$ of size $t = \mathcal{O}\left(\frac{k^k}{p^p q^q} \cdot k \log n \cdot \frac{1}{\epsilon^2}\right)$, can be computed with success probability at least $1 - 1/n^{ck}$ in time $\mathcal{O}(t \cdot n)$.*

We will need more sophisticated parsimonious universal families, constructed in a manner to enable having an efficient "membership query" procedure—that is, a procedure that given any set $P \in \binom{U}{p}$, outputs all the sets in the family that contain $P$. We will address the computation of such families and procedures in Section 3.2. Formally, they are defined as follows.

**Definition 2.2 (Membership Query Procedure).** *Let $n, p, q \in \mathbb{N}$ and $0 < \epsilon < 1$. Let $U$ be a universe of size $n$, and let $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathcal{F} \subseteq 2^U$ be an $\epsilon$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$. A $T$-*membership query procedure* is a procedure that given any set $P \in \mathcal{P}$ as input, outputs the subfamily $\{F \in \mathcal{F} : P \subseteq F\}$ in time $\mathcal{O}(T)$.*

For the case of so called product counters (defined in Section 5), we need to further generalize the notion of a membership procedure as well as consider disjointness procedures. Because our computation easily extends to the more general notion of membership procedure, we already give the definition here, so that in Section 3.2 we can directly present the computation for the more general notion and avoid repetition.

**Definition 2.3 (General Membership and Disjointness Query Procedures).** *Let $n, p, q \in \mathbb{N}$ and $0 < \epsilon < 1$. Let $U$ be a universe of size $n$. Let $\mathcal{P} \subseteq \binom{U}{p}$, $\mathcal{P}' \subseteq \{P' \subseteq P : P \in \mathcal{P}\}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathcal{F} \subseteq 2^U$ be an $\epsilon$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$. A $T$-*membership query procedure with respect to $\mathcal{P}'$ is a procedure that given any set $P' \in \mathcal{P}'$ as input, outputs the subfamily $\{F \in \mathcal{F} : P' \subseteq F\}$ in time $\mathcal{O}(T)$.*

*Additionally, A $T$-*disjointness query procedure is a procedure that given any set $Q \in \mathcal{Q}$ as input, outputs the subfamily $\{F \in \mathcal{F} : Q \cap F = \emptyset\}$ in time $\mathcal{O}(T)$.*

We will also make use of the following well known inequality to bound probabilities.

**Proposition 2.2 (Chernoff Bound).** *Let $X_1, \ldots, X_\ell$ be independent random variables bounded by the interval $[0, 1]$. Let $X = \sum_{i=1}^{\ell} X_i$. For any $\epsilon \geq 0$, $Pr(|X - E[X]| > \epsilon E[X]) \leq 2e^{-\frac{\epsilon^2 E[X]}{2}}$.*

Lastly, we need to define the notion of an arithmetic circuit. An *arithmetic circuit* $C$ over a commutative ring $R$ is a simple labelled directed acyclic graph whose internal nodes are labeled by + or × and whose leaves (in-degree zero nodes) are labeled from $X$ where $X = \{x_1, x_2, ..., x_n\}$ is a set of variables. There is a node of out-degree zero, called the *root* node or the output gate. The size of $C$, denoted by $s(C)$, is the number of nodes, $s_{\mathsf{V}}(C)$, plus the number of arcs, $s_{\mathsf{A}}(C)$, in the digraph.

## 3 Representative Counters

In this section, we will be working with counters, defined as follows.

**Definition 3.1 (Counter).** *Let $U$ be a universe. Let $p \in \mathbb{N}_0$, and let $\mathcal{P} \subseteq \binom{U}{p}$. A function $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ is called a* counter. *A counter is encoded as a collection of pairs, where each pair consists of an element $P \in \mathsf{supp}(\mathfrak{C})$ and its value $\mathfrak{C}(P)$.*

The main objective of this section is to compute representative counters, defined as follows.

**Definition 3.2** **((Approximate) Representative Counter).** *Let $U$ be a universe. Let $\alpha \leq 1, \beta \geq 1$, and let $p, q \in \mathbb{N}_0$. Let $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. A counter $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ is said to* $(\alpha, \beta, q)$-represent *a counter $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ with respect to $\mathcal{Q}$ if for every set $Q \in \mathcal{Q}$, the following condition is satisfied.*

$$\alpha \cdot \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \mathfrak{C}(P) \leq \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P) \leq \beta \cdot \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \mathfrak{C}(P).$$

*When $\alpha = 1 - \epsilon$ and $\beta = 1 + \epsilon$ for some $0 < \epsilon < 1$, $\widehat{\mathfrak{C}}$ is said to* $(\epsilon, q)$-represent $\mathfrak{C}$.

Further, we will need the representative counter to be, in expectation, not just similar, but identical to the given counter.

**Definition 3.3** **((Exact) Representative Counter in Expectation).** *Let $U$ be a universe. Let $p \in \mathbb{N}_0$, and let $\mathcal{P} \subseteq \binom{U}{p}$. A sampled counter $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ is said to* represent in expectation *a counter $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ if for every set $P \in \mathcal{P}$, the following condition is satisfied.*

$$E_{\widehat{\mathfrak{C}}}[\widehat{\mathfrak{C}}(P)] = \mathfrak{C}(P).$$

We will first show how to efficiently compute representative counters under the assumption that we can compute parsimonious universal families equipped with efficient membership query procedures. Next, we will show how to compute a parsimonious universal family equipped with efficient membership query procedure for specific choices of $(\mathcal{P}, \mathcal{Q})$. We remark that in what follows, we implicitly suppose that the counter to represent has non-empty support, because otherwise representation is trivial.

### 3.1 Computation of Representative Counters of Small Support

We first extend the notion of a counter to also assign values to sets of size larger than $p$.

**Definition 3.4** **(Domain Extension).** *Let $U$ be a universe. Let $p \in \mathbb{N}_0$, and let $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. The* extender $\mathfrak{C}_{\text{ext}} : 2^U \to \mathbb{N}_0$ *is defined as follows. For any set $F \subseteq U$, define*

$$\mathfrak{C}_{\text{ext}}(F) \triangleq \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \mathfrak{C}(P).$$

Notice that for any set $P \in \mathcal{P}$, we have that $\mathfrak{C}_{\text{ext}}(P) = \mathfrak{C}(P)$. Now, we present an alternative (to Definition 3.2) notion of similarity between counters, based on a given family $\mathcal{F}$ (that will, when used ahead, be a parsimonious universal family). In particular, it makes similarly, in a sense, be more focused, considering only sets in $\mathcal{F}$ rather than all possible choices of $P \in \mathcal{P}$ and $Q \in \mathcal{Q}$ in order to measure similarity. Being more focused, working with this definition for the *computation of representative counters* will also yield efficiency. Notice that this definition does not replace Definition 3.2—the *usage of representative counters* for applications will require Definition 3.2.

**Definition 3.5** **($(\epsilon, \mathcal{F})$-Similarly).** *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$. Let $0 < \epsilon < 1$, and let $p \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ and $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ be two counters. We say that $\mathfrak{C}$ and $\widehat{\mathfrak{C}}$ are* $(\epsilon, \mathcal{F})$-similar *if for every set $F \in \mathcal{F}$, $(1 - \epsilon) \cdot \mathfrak{C}_{\text{ext}}(F) \leq \widehat{\mathfrak{C}}_{\text{ext}}(F) \leq (1 + \epsilon) \cdot \mathfrak{C}_{\text{ext}}(F)$.*

We now prove that for the sake of efficient computation of representative counters, we can indeed work with the new definition.

**Lemma 3.1.** *Let $n, p, q \in \mathbb{N}$, $0 < \epsilon < 1$ and $0 < \delta < 1$. Let $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathcal{F} \subseteq 2^U$ be an $\epsilon$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ and $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ be $(\delta, \mathcal{F})$-similar counters. Then, $\widehat{\mathfrak{C}}$ $(4\epsilon + \delta, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$.*

*Proof.* To prove that $\widehat{\mathfrak{C}}$ $(\epsilon, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$, consider some set $Q \in \mathcal{Q}$. First, observe that

$$\begin{aligned}(*) \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} |\mathcal{F}[P,Q]| \cdot \mathfrak{C}(P) &= \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} \sum_{F \in \mathcal{F}[P,Q]} \mathfrak{C}(P) \\ &= \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \sum_{P \in \mathcal{P}: P \subseteq F} \mathfrak{C}(P) \\ &= \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \mathfrak{C}_{\text{ext}}(F).\end{aligned}$$

Let $T$ be the correction factor of $\mathcal{F}$. Then, for any set $P \in \mathcal{P}$, we have that $(1-\epsilon)T \leq |\mathcal{F}[P,Q]| \leq (1+\epsilon)T$. On the one hand, this implies that

$$\begin{aligned}\text{(I)} \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} \mathfrak{C}(P) &= \frac{1}{(1-\epsilon)T} \cdot \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} (1-\epsilon)T \cdot \mathfrak{C}(P) \\ &\leq \frac{1}{(1-\epsilon)T} \cdot \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} |\mathcal{F}[P,Q]| \cdot \mathfrak{C}(P) \\ &= \frac{1}{(1-\epsilon)T} \cdot \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \mathfrak{C}_{\text{ext}}(F).\end{aligned}$$

Here, the last equality was derived from equality (*). Symmetrically, we have that

$$\text{(II)} \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P) \leq \frac{1}{(1-\epsilon)T} \cdot \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \widehat{\mathfrak{C}}_{\text{ext}}(F).$$

On the other hand, this implies that

$$\begin{aligned}\text{(III)} \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} \mathfrak{C}(P) &= \frac{1}{(1+\epsilon)T} \cdot \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} (1+\epsilon)T \cdot \mathfrak{C}(P) \\ &\geq \frac{1}{(1+\epsilon)T} \cdot \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} |\mathcal{F}[P,Q]| \cdot \mathfrak{C}(P) \\ &= \frac{1}{(1+\epsilon)T} \cdot \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \mathfrak{C}_{\text{ext}}(F).\end{aligned}$$

Again, the last equality was derived from equality (*). Symmetrically, we have that

$$\text{(IV)} \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P) \geq \frac{1}{(1+\epsilon)T} \cdot \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \widehat{\mathfrak{C}}_{\text{ext}}(F).$$

Because $\mathfrak{C}$ and $\widehat{\mathfrak{C}}$ are $(\delta, \mathcal{F})$-similar, for any set $F \in \mathcal{F}$, we have that $(1-\delta) \cdot \mathfrak{C}_{\text{ext}}(F) \leq \widehat{\mathfrak{C}}_{\text{ext}}(F) \leq (1+\delta) \cdot \mathfrak{C}_{\text{ext}}(F)$. On the one hand, combined with inequalities (II) and (III), this implies that

$$\begin{aligned}\sum_{P \in \mathcal{P}: P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P) &\leq \frac{1}{(1-\epsilon)T} \cdot \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \widehat{\mathfrak{C}}_{\text{ext}}(F) \\ &\leq \frac{(1+\delta)}{(1-\epsilon)T} \cdot \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \mathfrak{C}_{\text{ext}}(F) \\ &= \frac{(1+\delta)(1+\epsilon)}{(1-\epsilon)} \cdot \frac{1}{(1+\epsilon)T} \cdot \sum_{F \in \mathcal{F}: Q \cap F = \emptyset} \mathfrak{C}_{\text{ext}}(F) \\ &\leq \frac{(1+\delta)(1+\epsilon)}{(1-\epsilon)} \cdot \sum_{P \in \mathcal{P}: P \cap Q = \emptyset} \mathfrak{C}(P).\end{aligned}$$

On the other hand, combined with inequalities (I) and (IV), this implies that

$$\begin{aligned}\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\widehat{\mathfrak{C}}(P) &\geq \frac{1}{(1+\epsilon)T}\cdot\sum_{F\in\mathcal{F}:Q\cap F=\emptyset}\widehat{\mathfrak{C}}_{\text{ext}}(F)\\ &\geq \frac{(1-\delta)}{(1+\epsilon)T}\cdot\sum_{F\in\mathcal{F}:Q\cap F=\emptyset}\mathfrak{C}_{\text{ext}}(F)\\ &= \frac{(1-\delta)(1-\epsilon)}{(1+\epsilon)}\cdot\frac{1}{(1-\epsilon)T}\cdot\sum_{F\in\mathcal{F}:Q\cap F=\emptyset}\mathfrak{C}_{\text{ext}}(F)\\ &\geq \frac{(1-\delta)(1-\epsilon)}{(1+\epsilon)}\cdot\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\mathfrak{C}(P).\end{aligned}$$

Overall, we have that

$$\frac{(1+\delta)(1-\epsilon)}{(1+\epsilon)}\cdot\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\widehat{\mathfrak{C}}(P)\leq\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\mathfrak{C}(P)\leq\frac{(1+\delta)(1+\epsilon)}{(1-\epsilon)}\cdot\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\widehat{\mathfrak{C}}(P).$$

Notice that $1-\epsilon>1-\epsilon-2\epsilon^2=(1-2\epsilon)(1+\epsilon)$, and hence $(1-\epsilon)/(1+\epsilon)>1-2\epsilon$; similarly, $1+\epsilon>1+\epsilon-2\epsilon^2=(1+2\epsilon)(1-\epsilon)$, and hence $(1+\epsilon)/(1-\epsilon)>1+2\epsilon$. Moreover, because $0<\epsilon,\delta<1$, $(1-\delta)(1-2\epsilon)=1-(2\epsilon+\delta-2\epsilon\delta)>1-(4\epsilon+\delta)$, and $(1+\delta)(1+2\epsilon)=1+(2\epsilon+\delta+2\epsilon\delta)<1+(4\epsilon+\delta)$. Thus,

$$(1-(4\epsilon+\delta))\cdot\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\mathfrak{C}(P)\leq\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\widehat{\mathfrak{C}}(P)\leq(1+(4\epsilon+\delta))\cdot\sum_{P\in\mathcal{P}:P\cap Q=\emptyset}\mathfrak{C}(P).$$

Since the choice of $Q\in\mathcal{Q}$ was arbitrary, the proof is complete. □

Our computation of representative counters will be done in a sampling procedure defined as follows. (Some explanation of the intuition behind it is given ahead.)

**Definition 3.6** (**$(\mathfrak{C},\mathcal{F})$-Counter Sampling).** *Let $U$ be a universe, and let $\mathcal{F}\subseteq 2^U$ with $U\in\mathcal{F}$. Let $p,L\in\mathbb{N}_0$ and $\mathcal{P}\subseteq\binom{U}{p}$. Let $\mathfrak{C}:\mathcal{P}\rightarrow\mathbb{N}_0$ be a counter. Then,* $(\mathfrak{C},\mathcal{F},L)$-counter sampling *is the randomized procedure that constructs a counter $\widehat{\mathfrak{C}}:\mathcal{P}\rightarrow\mathbb{N}_0$ as follows. For any set $P\in\mathcal{P}$, define*

$$\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)\triangleq\min_{F\in\mathcal{F}:P\subseteq F}\mathfrak{C}_{\text{ext}}(F),$$

$$\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)\triangleq\min(1,L\cdot\frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)}),\text{ and}$$

$$\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P)\triangleq\frac{\mathfrak{C}(P)}{\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)}.$$

*Then, for any set $P\in\mathcal{P}$, set $\widehat{\mathfrak{C}}(P)$ to $\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P)$ with probability $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)$ and to $0$ with probability $1-\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)$.*[3]

Firstly, observe that the support of any counter that can be potentially output is contained in the support of the input counter. Essentially, with this sampling procedure we aim to discard as many sets as possible from the support of the input counter while modifying the values of those that are kept so that we obtain a representative counter of small support. Intuitively, each set $P\in\mathcal{P}$ is associated, among the sets in $\mathcal{F}$ that contain it and hence whose value is effected by the value of $P$ (as assigned by the counter), with a set $F$ having minimum value. Thus,

[3] Since $U\in\mathcal{F}$, there exists $F\in\mathcal{F}$ such that $P\subseteq F$, hence $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)$ is well defined.

$P$ is associated with a set $F$ for which $P$ is most significant among all sets in $\mathcal{F}$—that is, in which the fraction of the value of $P$ from the entire value of $F$ is largest. In a sense, this means that the value of $F$ is most "vulnerable" in case $P$ will be dropped from the support of the counter. Next, the probability of keeping $P$ in the support is chosen to be proportional to its fraction of value within $F$—the larger $\mathfrak{C}(P)$ is, the larger is the probability to choose it, but at the same time, the larger $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)$ is (which means that the set $F$ associated with $P$, and hence all other sets in $\mathcal{F}$ as well, are less vulnerable to $P$ being dropped out), the smaller is the probability to choose $P$. The factor $L$ (whose exact value will be determined later) is meant to boost up the probability to be larger than just the fraction of the value of $P$ within $F$ (else we may drop "too many" sets from the support, and hence the output counter will not represent the input counter). Due to this boosting factor, we also need to trim down the boosted fraction to be 1 so that it will indeed represent a probability. Lastly, the new value of $P$ when decided to be kept in the support, is chosen in a way as to ensure that its expected value (being the probability to keep it times its new value when it is kept) will be equal to its original value.

We first show the the size of the support of the output counter is expected to be "small" (in case the size of the family $\mathcal{F}$ and the boosting factor $L$ are both "small").

**Lemma 3.2.** *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$. Let $p, L \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. Then, the expected size of the support of the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling is upper bounded as follows.*

$$E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] \leq |\mathcal{F}| \cdot L.$$

*Moreover, for any $\widehat{c} \geq 0$, we have that $Pr(|\mathsf{supp}(\widehat{\mathfrak{C}})| > (\widehat{c} + 1) \cdot |\mathcal{F}| \cdot L) \leq 2e^{-\frac{\widehat{c}^2}{2}}$.*

*Proof.* Observe that

$$\begin{aligned}
E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] &= \sum_{P \in \mathcal{P}} \mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) && (1)\\
&= \sum_{P \in \mathcal{P}} \min(1, L \cdot \frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)}) && (2)\\
&\leq L \cdot \sum_{P \in \mathcal{P}} \frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)} && (3)\\
&\leq L \cdot \sum_{F \in \mathcal{F}} \sum_{P \in \mathcal{P} : \mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P) = \mathfrak{C}_{\text{ext}}(F)} \frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)} && (4)\\
&= L \cdot \sum_{F \in \mathcal{F}} \sum_{P \in \mathcal{P} : \mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P) = \mathfrak{C}_{\text{ext}}(F)} \frac{\mathfrak{C}(P)}{\mathfrak{C}_{\text{ext}}(F)} && (5)\\
&\leq L \cdot \sum_{F \in \mathcal{F}} \left( \frac{1}{\mathfrak{C}_{\text{ext}}(F)} \cdot \sum_{P \in \mathcal{P} : \mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P) = \mathfrak{C}_{\text{ext}}(F)} \mathfrak{C}(P) \right) && (6)\\
&\leq L \cdot \sum_{F \in \mathcal{F}} \left( \frac{1}{\mathfrak{C}_{\text{ext}}(F)} \cdot \sum_{P \in \mathcal{P} : P \subseteq F} \mathfrak{C}(P) \right) && (7)\\
&\leq L \cdot \sum_{F \in \mathcal{F}} \frac{1}{\mathfrak{C}_{\text{ext}}(F)} \cdot \mathfrak{C}_{\text{ext}}(F) = L \cdot |\mathcal{F}|. && (8)
\end{aligned}$$

Here, (1), (3), (5), (6) and the equality at (8) are immediate. The equality (2) follows from the definition of $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)$. The inequality (4) follows from the observation that for each set $P \in \mathcal{P}$, there exists a (not necessarily unique) set $F \in \mathcal{F}$ such that $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P) = \mathfrak{C}(F)$. The inequality (7) follows from the definition of $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}$, and the inequality at (8) follows from the definition of $\mathfrak{C}_{\text{ext}}$.

For the second claim in the proof, let $\widehat{c} \geq 1$. Because $E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] \leq |\mathcal{F}| \cdot L$, we have that $Pr(|\mathsf{supp}(\widehat{\mathfrak{C}})| > (\widehat{c}+1) \cdot |\mathcal{F}| \cdot L) \leq Pr(||\mathsf{supp}(\widehat{\mathfrak{C}})| - E[|\mathsf{supp}(\widehat{\mathfrak{C}})|]| > \widehat{c} \cdot E[|\mathsf{supp}(\widehat{\mathfrak{C}})|])$. By Chernoff bound (Proposition 2.2), the aforementioned term is upper bounded by $2e^{-\frac{\widehat{c}^2 E[|\mathsf{supp}(\widehat{\mathfrak{C}})|]}{2}}$. In case $E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] \geq 1$, then the aforementioned term is upper bounded by $2e^{-\frac{\widehat{c}^2}{2}}$, which completes the proof. Else, in case $E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] < 1$, we have that

$$\begin{aligned} E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] &= \sum_{P \in \mathcal{P}} \mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) \\ &= \sum_{P \in \mathcal{P}} \min(1, L \cdot \frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)}) < 1. \end{aligned}$$

Thus, for every $P \in \mathcal{P}$, we have that $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) = L \cdot \frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)}$. Moreover, for every $P \in \mathcal{P}$, we have that $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P) \leq \mathfrak{C}_{\mathrm{ext}}(U)$, and therefore $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) \geq L \cdot \frac{\mathfrak{C}(P)}{\mathfrak{C}_{\mathrm{ext}}(U)}$. However, we thus derive that $E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] = \sum_{P \in \mathcal{P}} \mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) \geq L \cdot \frac{\sum_{P \in \mathcal{P}} \mathfrak{C}(P)}{\mathfrak{C}_{\mathrm{ext}}(U)} = L \geq 1$, which is a contradiction. □

Now, we present a statement regarding the new values and probabilities assigned by the sampling procedure. This statement will be used soon together with the observation ahead, towards the proof that the output counter is likely to represent the input one.

**Lemma 3.3.** *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$. Let $p, L \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. Then, for all $P \in \mathcal{P}$ and $F \in \mathcal{F}$ such that $P \subseteq F$, at least one of the following two conditions is satisfied.*

- $\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P) \leq \dfrac{\mathfrak{C}_{\mathrm{ext}}(F)}{L}$.
- $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) = 1$.

*Proof.* Consider some $P \in \mathcal{P}$ and $F \in \mathcal{F}$ such that $P \subseteq F$. We need to prove that at least one of the two conditions in the lemma is satisfied. In case $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) = 1$, we are done. Thus, we next suppose that $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) \neq 1$. Then, by the definition of $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)$, we have that $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) = L \cdot \frac{\mathfrak{C}(P)}{\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)}$. By the definition of $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)$, we have that $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P) \leq \mathfrak{C}_{\mathrm{ext}}(F)$. Thus, $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) \geq L \cdot \frac{\mathfrak{C}(P)}{\mathfrak{C}_{\mathrm{ext}}(F)}$. From this inequality and the definition of $\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P)$, we derive that

$$\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P) = \frac{\mathfrak{C}(P)}{\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)} \leq \frac{\mathfrak{C}(P)}{L \cdot \frac{\mathfrak{C}(P)}{\mathfrak{C}_{\mathrm{ext}}(F)}} = \frac{\mathfrak{C}_{\mathrm{ext}}(F)}{L}.$$

This completes the proof. □

We will also need the following two simple observations where the first asserts representation in expectation and the second, which is an immediate consequence of the first, concerns the expected output value of each set in $\mathcal{F}$.

**Observation 3.1.** *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$. Let $p, L \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. The output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling represents in expectation $\mathfrak{C}$.*

*Proof.* Consider some set $P \in \mathcal{P}$. Then, the definition of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling yields that

$$E[\widehat{\mathfrak{C}}(P)] = \mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) \cdot \mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P) = \mathfrak{C}(P).$$

Since the choice of $P$ was arbitrary, we derive that $\widehat{\mathfrak{C}}$ represents in expectation $\mathfrak{C}$. □

**Observation 3.2.** *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$. Let $p, L \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. For any set $F \subseteq U$, for the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling, we have that $E[\widehat{\mathfrak{C}}_{\text{ext}}(F)] = \mathfrak{C}_{\text{ext}}(F)$.*

*Proof.* Consider some set $F \subseteq U$. Then,

$$\begin{aligned} E[\widehat{\mathfrak{C}}_{\text{ext}}(F)] &= E[\sum_{P \in \binom{F}{p} \cap \mathcal{P}} \widehat{\mathfrak{C}}(P)] & (1)\\ &= \sum_{P \in \binom{F}{p} \cap \mathcal{P}} E[\widehat{\mathfrak{C}}(P)] & (2)\\ &= \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \mathfrak{C}(P) = \mathfrak{C}_{\text{ext}}(F). & (3) \end{aligned}$$

Here, equality (1) and the second equality at (3) follow from the definition of domain extension, equality (2) follows from the linearity of expectation, and the first equality at (3) follows from Observation 3.1. □

From Lemma 3.3 and Observation 3.2, we derive the following corollary.

**Corollary 3.1.** *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$. Let $p, L \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. For any $F \in \mathcal{F}$, for the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling, we have that $E[\frac{\widehat{\mathfrak{C}}_{\text{ext}}(F)}{W}] \geq L$ where $W = \max\{\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P) : P \in \mathcal{P}, \mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) < 1\}$.*

*Proof.* Consider some $F \in \mathcal{F}$. By Lemma 3.3, for all $P \in \binom{F}{p} \cap \mathcal{P}$ such that $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) < 1$, it must hold that $\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P) \leq \frac{\mathfrak{C}_{\text{ext}}(F)}{L}$, implying that necessarily $W \leq \frac{\mathfrak{C}_{\text{ext}}(F)}{L}$. Therefore,

$$E[\frac{\widehat{\mathfrak{C}}_{\text{ext}}(F)}{W}] = \frac{1}{W} \cdot E[\widehat{\mathfrak{C}}_{\text{ext}}(F)] \geq \frac{L}{\mathfrak{C}_{\text{ext}}(F)} \cdot E[\widehat{\mathfrak{C}}_{\text{ext}}(F)] = L.$$

Here, the last equality follows from Observation 3.2. □

We are now ready to prove that the output counter is likely to represent the input one.

**Lemma 3.4.** *Let $U$ be a universe, and let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$. Let $p, c, L \in \mathbb{N}_0$ such that $L \geq 2\frac{1}{\epsilon^2}\ln(2c|\mathcal{F}|)$. Let $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. Then, the probability that $\mathfrak{C}$ and the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling are $(\epsilon, \mathcal{F})$-similar is at least $1 - \frac{1}{c}$.*

*Proof.* Consider some $F \in \mathcal{F}$, and denote $W = \max\{\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P) : P \in \mathcal{P}, \mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) < 1\}$. For any $P \in \binom{F}{p} \cap \mathcal{P}$ such that $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) < 1$, define $\ell_P = 1$ and the random variable $X_{P,1} = \frac{\widehat{\mathfrak{C}}(P)}{W}$. For any $P \in \binom{F}{p} \cap \mathcal{P}$ such that $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P) = 1$, define $\ell_P = \lceil \frac{\widehat{\mathfrak{C}}(P)}{W} \rceil$ and the deterministic variable $X_{P,\ell_P} = \frac{\widehat{\mathfrak{C}}(P)}{W} - (\ell_P - 1)$, as well as for any $i \in \{1, \ldots, \ell_P - 1\}$, the deterministic variable $X_{P,i} = 1$, and notice that $\sum_{i=1}^{\ell_P} X_{P,i} = \frac{\widehat{\mathfrak{C}}(P)}{W}$. Then, $\{X_{P,i} : P \in \binom{F}{p} \cap \mathcal{P}, i \in \{1, \ldots, \ell_P\}\}$ is a collection of independent random variables bounded by the interval $[0, 1]$. Here, independence among variables $X_{P,i}$ corresponding to the same set $P \in \binom{F}{p} \cap \mathcal{P}$ follows because these variables are deterministic. Let $\overline{X} = \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \sum_{i=1}^{\ell_P} X_{P,i}$. By Chernoff bound (Proposition 2.2),

$$Pr(|\overline{X} - E[\overline{X}]| > \epsilon E[\overline{X}]) \leq 2e^{-\frac{\epsilon^2 E[\overline{X}]}{2}}.$$

Observe that $\overline{X} = \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \sum_{i=1}^{\ell_P} X_{P,i} = \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \frac{\widehat{\mathfrak{C}}(P)}{W} = \frac{\widehat{\mathfrak{C}}_{\text{ext}}(F)}{W}$. Thus, $E[\overline{X}] = E[\frac{\widehat{\mathfrak{C}}_{\text{ext}}(F)}{W}]$, hence by Observation 3.2, $E[\overline{X}] = \frac{\mathfrak{C}_{\text{ext}}(F)}{W}$. This means that $|\overline{X} - E[\overline{X}]| > \epsilon E[\overline{X}]$ is true if and only if $|\widehat{\mathfrak{C}}_{\text{ext}}(F) - \mathfrak{C}_{\text{ext}}(F)| > \epsilon \cdot \mathfrak{C}_{\text{ext}}(F)$ is true. Thus, by this equivalence between events,

$$Pr(|\widehat{\mathfrak{C}}_{\text{ext}}(F) - \mathfrak{C}_{\text{ext}}(F)| > \epsilon \cdot \mathfrak{C}_{\text{ext}}(F)) \leq 2e^{-\frac{\epsilon^2 E[\overline{X}]}{2}}.$$

Recall that $E[\overline{X}] = E[\frac{\widehat{\mathfrak{C}}_{\text{ext}}(F)}{W}]$, hence by Corollary 3.1 and the given lower bound on $L$, $E[\overline{X}] \geq L \geq 2\frac{1}{\epsilon^2}\ln(2c|\mathcal{F}|)$. Thus,

$$\begin{aligned} Pr(|\widehat{\mathfrak{C}}_{\text{ext}}(F) - \mathfrak{C}_{\text{ext}}(F)| > \epsilon \cdot \mathfrak{C}_{\text{ext}}(F)) &\leq 2e^{-\frac{\epsilon^2 \cdot (2\frac{1}{\epsilon^2}\ln(2c|\mathcal{F}|))}{2}} \\ &= 2e^{-\ln(2c|\mathcal{F}|)} = \frac{2}{2c|\mathcal{F}|} = \frac{1}{c|\mathcal{F}|}. \end{aligned}$$

As the choice of $F \in \mathcal{F}$ was arbitrary, union bound implies that the probability that there exists $F \in \mathcal{F}$ such that $(1-\epsilon) \cdot \widehat{\mathfrak{C}}_{\text{ext}}(F) > \mathfrak{C}_{\text{ext}}(F)$ or $\mathfrak{C}_{\text{ext}}(F) > (1+\epsilon) \cdot \widehat{\mathfrak{C}}_{\text{ext}}(F)$ is upper bounded by $|\mathcal{F}| \cdot \frac{1}{c|\mathcal{F}|} = \frac{1}{c}$. Thus, the probability that $\mathfrak{C}$ and $\widehat{\mathfrak{C}}$ are $(\epsilon, \mathcal{F})$-similar is at least $1 - \frac{1}{c}$. □

We now turn to analyze the time complexity of the sampling procedure.

**Lemma 3.5.** *Let $U$ be a universe. Let $p, L \in \mathbb{N}_0$ and $\mathcal{P} \subseteq \binom{U}{p}$. Let $\mathcal{F} \subseteq 2^U$ with $U \in \mathcal{F}$ be an $\epsilon$-parsimonious $(n,p,q)$-universal family $\mathcal{F}$, equipped with a $T$-membership query procedure. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. Then, the time complexity of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling is bounded by $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot T)$.*

*Proof.* First, we initialize the value $\mathfrak{C}_{\text{ext}}(F)$ of each set $F \in \mathcal{F}$ to be 0. Then, for every set $P \in \mathsf{supp}(\mathfrak{C})$, we compute $\mathcal{F}' = \{F \in \mathcal{F} : P \subseteq F\}$ in time $\mathcal{O}(T)$ using the membership query procedure (which implies that $|\mathcal{F}'| = \mathcal{O}(T)$), and then for each set $F \in \mathcal{F}'$ we update $\mathfrak{C}_{\text{ext}}(F)$ by adding $\mathfrak{C}(P)$ to it. Thus, in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot T)$ we correctly compute $\mathfrak{C}_{\text{ext}}(F)$ for all $F \in \mathcal{F}$. Now, for each set $P \in \mathsf{supp}(\mathfrak{C})$, we can compute $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)$ in time $\mathcal{O}(T)$, then $\mathsf{prob}_{\mathfrak{C},\mathcal{F},L}(P)$ in time $\mathcal{O}(1)$, and lastly $\mathsf{count}_{\mathfrak{C},\mathcal{F},L}(P)$ in time $\mathcal{O}(1)$. Overall, we have so far spent time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot T)$. Finally, picking up sets using their probabilities and new values is done in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})|)$. □

We conclude this subsection with the following theorem.

**Theorem 3.1.** *Let $U$ be a universe. Let $0 < \epsilon < 1$, $p, q, c \in \mathbb{N}_0$, $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathcal{F} \subseteq 2^U$ be an $\frac{1}{5}\epsilon$-parsimonious $(n,p,q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$ of size $S$, equipped with a $T$-membership query procedure. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. Then, a counter $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ such that*

1. *$\widehat{\mathfrak{C}}$ necessarily (with probability 1) represents in expectation $\mathfrak{C}$, and*
2. *with success probability at least $1 - \frac{1}{c}$, $\widehat{\mathfrak{C}}$ $(\epsilon, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$ and satisfies $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq \mathcal{O}((\frac{1}{\epsilon})^2 S \log c(\log c + \log S))$,*

*can be computed in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot T)$.*

*Proof.* Without loss of generality, we suppose that $U \in \mathcal{F}$, else we just add $U$ to $\mathcal{F}$. By Lemma 3.1, to prove the theorem, it suffices to compute in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot T)$ a counter $\widehat{\mathfrak{C}} : \binom{U}{p} \to \mathbb{N}_0$ that necessarily represents in expectation $\mathfrak{C}$, and that with probability at least $1 - \frac{1}{c}$ is $(\frac{\epsilon}{5}, \mathcal{F})$-similar to $\mathfrak{C}$ and satisfies $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq \mathcal{O}((\frac{1}{\epsilon})^2 S \log c \log(cS))$.

Fix $L = \lceil 2 \frac{1}{(\frac{\epsilon}{5})^2} \ln(4cS) \rceil = \mathcal{O}((\frac{1}{\epsilon})^2 \log(cS))$. By Lemma 3.2 with $\widehat{c} = \sqrt{2 \ln(4c)}$, with probability at most $2e^{-\frac{\widehat{c}^2}{2}} = 2e^{-\ln(4c)} = \frac{1}{2c}$, we have that the expected size of the support of the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}, \mathcal{F}, L)$-counter sampling is upper bounded as follows.

$$E[|\mathsf{supp}(\widehat{\mathfrak{C}})|] \leq (\widehat{c} + 1) SL.$$

Moreover, by Lemma 3.4, $\mathfrak{C}$ and $\widehat{\mathfrak{C}}$ are $(\frac{\epsilon}{5}, \mathcal{F})$-similar with probability at least $1 - \frac{1}{2c}$. By union bound, the probability that $|\mathsf{supp}(\widehat{\mathfrak{C}})| \geq (\widehat{c} + 1)SL$ or that $\mathfrak{C}$ and $\widehat{\mathfrak{C}}$ are *not* $(\frac{\epsilon}{5}, \mathcal{F})$-similar is at most $\frac{1}{2c} + \frac{1}{2c} = \frac{1}{c}$. Thus, with probability at least $1 - \frac{1}{c}$, both $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq (\widetilde{c} + 1)SL = \mathcal{O}(\log c \cdot S \cdot (\frac{1}{\epsilon})^2 \log(cS)) = \mathcal{O}((\frac{1}{\epsilon})^2 S \log c (\log c + \log S))$ and $\mathfrak{C}$ and $\widehat{\mathfrak{C}}$ are $(\frac{\epsilon}{5}, \mathcal{F})$-similar. Further, by Lemma 3.5, $\widehat{\mathfrak{C}}$ is computed in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C})| \cdot T)$. Lastly, by Observation 3.1, $\widehat{\mathfrak{C}}$ necessarily represents in expectation $\mathfrak{C}$. This completes the proof. □

We remark that as a corollary to this theorem (with $c = 2$ and where the membership query procedure is simply brute-force) and Proposition 2.1, we can already assert the *existence* of representative counters of small support.

## 3.2 Computation of Parsimonious Universal Families Equipped with Membership Query Procedures for Balancedly-Split Sets

We will be able to equip our parsimonious universal families with efficient membership query procedures only when we deal with $\mathcal{P}$ and $\mathcal{Q}$ that are "balancedly split". Towards the definition of this term, we first present the following definition.

**Definition 3.7 (Partitioned Universe, Splitting Function Pair).** *Let $t, k, p, b \in \mathbb{N}$. A tuple $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ where $U_1, U_2, \ldots, U_t$ are pairwise-disjoint universes is called a $t$-*partitioned universe. *Moreover, a function $f : \{1, 2, \ldots, t\} \to \{0, 1, \ldots, \lceil bk/t \rceil\}$ that satisfies $\sum_{i=1}^{t} f(i) = k$ is called a $(t, k, b)$-*splitting function. *Lastly, a pair $(f, g)$ of a $(t, k, b)$-splitting function $f : \{1, 2, \ldots, t\} \to \{0, 1, \ldots, \lceil bk/t \rceil\}$ and a function $g : \{1, 2, \ldots, t\} \to \{0, 1, \ldots, \lceil bk/t \rceil\}$ that satisfies $g \leq f$ and $\sum_{i=1}^{t} g(i) = p$, is called a $(t, k, p, b)$-*splitting function pair.

When $t$ or $(t, k, p, b)$ is clear from context, we do not mention it explicitly. Notice that when $p = k$, necessarily $g = f$. We now present a definition which will be useful only for Section 5; by considering it already here, we will be able to avoid repetition of arguments.

Now, we define the notion of balancedly split sets.

**Definition 3.8 (Balancedly Split Sets I).** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$, and let $(f, g)$ be a splitting function pair. Then, $P \in \binom{U}{p}$ is* $(\overline{\mathbf{U}}, f, g)$-balancedly split *if for every $i \in \{1, 2, \ldots, t\}$, it holds that $|P \cap U_i| = g(i)$; in case $k = p$, $P$ is* $(\overline{\mathbf{U}}, f)$-balancedly split . *Further, $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}}, f, g} \subseteq \binom{U}{p}$ denotes the collection of all $(\overline{\mathbf{U}}, f, g)$-balancedly split sets. Moreover, $Q \in \binom{U}{k-p}$ is* complementary $(\overline{\mathbf{U}}, f, g)$-balancedly split *if for every $i \in \{1, 2, \ldots, t\}$, it holds that $|Q \cap U_i| = f(i) - g(i)$. Further, $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}}, f, g} \subseteq \binom{U}{k-p}$ denotes the collection of all complementary $(\overline{\mathbf{U}}, f, g)$-balancedly split sets.*

When $\overline{\mathbf{U}}$, $f$ and $g$ are clear from context, we do not mention it explicitly.

Our computation of universal families will be done in a sampling procedure defined as follows.

**Definition 3.9 (Parsimonious Universal Family Sampling).** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. Then, $(\overline{\mathbf{U}}, f, g, \epsilon, c, d)$-*universal family sampling *is the randomized procedure that constructs a family $\mathcal{F} \subseteq 2^U$ as follows.*

- *For $i \in \{1, 2, \ldots, t\}$:*
  - *For $j \in \{1, 2, \ldots, s_i\}$ with*

$$s_i = \frac{(d \cdot f(i))^{f(i)}}{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}} \cdot \frac{1}{\widehat{\epsilon}^2} \cdot 10k \cdot \ln(nc),$$

    *where $\widehat{\epsilon} = \frac{\ln(1+\epsilon)}{t}$, construct a set $F_{i,j} \subseteq U_i$ as follows. Each element in $U_i$ is inserted independently with probability $\dfrac{g(i)}{d \cdot f(i)}$ into $F_{i,j}$.*
  - *Denote $\mathcal{F}_i = \{F_{i,j} : j \in \{1, 2, \ldots, s_i\}\}$.*
- *Then, construct $\mathcal{F} = \{F_{1,j_1} \cup F_{2,j_2} \cup \cdots \cup F_{t,j_t} : F_{1,j_1} \in \mathcal{F}_1, F_{i,j_2} \in \mathcal{F}_2, \ldots, F_{i,j_t} \in \mathcal{F}_t\}$.*

We remark that $d$ can depend on any argument of interest (e.g., $k$ and $p$). We begin the analysis of the sampling procedure by an observation concerning its time complexity and by giving an upper bound on the size of the family it produces.

**Observation 3.3.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. Then, the time complexity of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal is $\mathcal{O}(|\mathcal{F}|n)$, where $\mathcal{F} \subseteq 2^U$ is the output family.*

**Lemma 3.6.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. Then, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling necessarily satisfies $|\mathcal{F}| \leq \dfrac{(dk)^k}{p^p(dk-p)^{k-p}} \cdot (\dfrac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc))^t$.*

*Proof.* Because $t \leq k$ and $\widehat{\epsilon} = \frac{\ln(1+\epsilon)}{t}$, we have that $\frac{1}{\widehat{\epsilon}^2} \leq \frac{1}{\ln^2(1+\epsilon)} \cdot k^2$. Thus,

$$\begin{aligned}
|\mathcal{F}| &= \prod_{i=1}^{t} s_i \\
&= \prod_{i=1}^{t} \left( \frac{(d \cdot f(i))^{f(i)}}{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}} \cdot \frac{1}{\widehat{\epsilon}^2} \cdot 10k \cdot \ln(nc) \right) \\
&\leq \left( \prod_{i=1}^{t} \frac{(d \cdot f(i))^{f(i)}}{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}} \right) \cdot \left( \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) \right)^t.
\end{aligned}$$

Recall that $f : \{1, 2, \ldots, t\} \to \{1, 2, \ldots, \lceil bk/t \rceil\}$ and $g \leq f$ satisfy $\sum_{i=1}^{t} f(i) = k$ and $\sum_{i=1}^{t} g(i) = p$. Relaxing the supposition $f : \{1, 2, \ldots, t\} \to \{1, 2, \ldots, \lceil bk/t \rceil\}$ to $f : \{1, 2, \ldots, t\} \to \{1, 2, \ldots, k\}$, the maximum value of the term $\prod_{i=1}^{t} \dfrac{(d \cdot f(i))^{f(i)}}{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}}$ is attained when $f(i) = k$ and $g(i) = p$ for some $i \in \{1, 2, \ldots, t\}$, and $f(i') = g(i') = 0$ for all other $i' \in \{1, 2, \ldots, t\} \setminus \{i\}$. Then, the value is $\dfrac{(dk)^k}{p^p(dk-p)^{k-p}}$. This completes the proof. □

We proceed by giving a lower bound for the probability of failure of the procedure to produce a parsimonious universal family with respect to a balancedly split pair.

**Lemma 3.7.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. With probability at least $1 - \frac{1}{2c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling is an $\epsilon$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}_{k,f,g}, \mathcal{Q}^{\mathrm{CBAL}}_{k,f,g})$ with correction factor upper bounded by $\left(\frac{1}{(\frac{\ln(1+\epsilon)}{t})^2} \cdot 10k \cdot \ln(nc)\right)^t$.*

*Proof.* Towards the proof of the lemma, we first show that the following claim is correct.

**Claim 3.1.** *With probability at least $1 - \frac{1}{2c}$, for every $i \in \{1, 2, \ldots, t\}$, we have that $\mathcal{F}_i$ is an $\widehat{\epsilon}$-parsimonious $(|U_i|, g(i), f(i) - g(i))$-universal family with respect to $(\binom{U_i}{g(i)}, \binom{U_i}{f(i)-g(i)})$ with correction factor $T_i = \frac{1}{\widehat{\epsilon}^2} \cdot 10k \cdot \ln(nc)$.*

*Proof.* By union bound, it suffices to choose some $i \in \{1, 2, \ldots, t\}$, and prove that with failure probability at most $\frac{1}{2ct}$, we have that $\mathcal{F}_i$ is an $\widehat{\epsilon}$-parsimonious $(|U_i|, g(i), f(i) - g(i))$-universal family with respect to $(\binom{U_i}{g(i)}, \binom{U_i}{f(i)-g(i)})$ with correction factor $T_i = \frac{1}{\widehat{\epsilon}^2} \cdot 10k \cdot \ln(nc)$. Further, by union bound, because there are at most $|U_i|^{f(i)} \leq n^k$ pairs of disjoint sets $P \in \binom{U_i}{g(i)}$ and $Q \in \binom{U_i}{f(i)-g(i)}$, it suffices to choose some such pair of disjoint sets $P \in \binom{U_i}{g(i)}$ and $Q \in \binom{U_i}{f(i)-g(i)}$, and prove that with failure probability at most $\frac{1}{2ctn^k}$, it holds that $(1-\widehat{\epsilon})T_i \leq |\mathcal{F}_i[P, Q]| \leq (1+\widehat{\epsilon})T_i$.

Towards the proof of the above, observe that each set $F_{i,j} \in \mathcal{F}_i$ contains $P$ and is disjoint from $Q$ with probability $\frac{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}}{(d \cdot f(i))^{f(i)}}$. Thus, the expected number of sets in $\mathcal{F}_i$ that contain $P$ and are disjoint from $Q$ is $T_i$. Because the sets in $\mathcal{F}_i$ are sampled independently from one another, by Chernoff bound (Proposition 2.2), we have that

$$\begin{aligned} Pr(||\mathcal{F}_i[P, Q]| - T_i| > \widehat{\epsilon} T_i) &\leq 2e^{-\frac{\widehat{\epsilon}^2 T_i}{2}} \\ &= 2e^{-5k \cdot \ln(nc)} = \frac{2}{(nc)^{5k}} \leq \frac{2}{n^4 \cdot n^k \cdot c} \leq \frac{1}{2ct} \end{aligned}$$

Here, the last inequality follows since $n \geq \max(2, t)$. This completes the proof of the claim. □

We now return to the proof of the lemma. Let $T = \prod_{i=1}^{t} T_i$ where $T_i$ is the correction factor of $\mathcal{F}_i$. Then, $T = \left(\frac{1}{\widehat{\epsilon}^2} \cdot 10k \cdot \ln(nc)\right)^t$. Due to Claim 3.1, to prove the lemma it suffices to show that, under the assumption that for every $i \in \{1, 2, \ldots, t\}$, we have that $\mathcal{F}_i \subseteq 2^{U_i}$ is an $\widehat{\epsilon}$-parsimonious $(|U_i|, g(i), f(i) - g(i))$-universal family with respect to $(\binom{U_i}{g(i)}, \binom{U_i}{f(i)-g(i)})$, it holds that $\mathcal{F}$ is an $\epsilon$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}, \mathcal{Q}^{\mathrm{CBAL}})$ with correction factor $T$. Towards the proof of this, consider some pair of disjoint sets $P \in \mathcal{P}^{\mathrm{BAL}}$ and $Q \in \mathcal{Q}^{\mathrm{CBAL}}$). Then,

$$|\mathcal{F}[P, Q]| = \prod_{i=1}^{t} |\mathcal{F}_i[P \cap U_i, Q \cap U_i]|.$$

Because $P \in \mathcal{P}^{\mathrm{BAL}}$ and $Q \in \mathcal{Q}^{\mathrm{CBAL}}$, it holds that for every $i \in \{1, 2, \ldots, t\}$, $P \cap U_i \in \binom{U_i}{g(i)}$ and $Q \cap U_i \in \binom{U_i}{f(i)-g(i)}$. Thus, for every $i \in \{1, 2, \ldots, t\}$, because $\mathcal{F}_i$ is an $\widehat{\epsilon}$-parsimonious

$(|U_i|, g(i), f(i)-g(i))$-universal family with respect to $(\binom{U_i}{g(i)}, \binom{U_i}{f(i)-g(i)})$, it holds that

$$(1-\widehat{\epsilon})T_i \leq |\mathcal{F}_i[P \cap U_i, Q \cap U_i]| \leq (1+\widehat{\epsilon})T_i$$

Therefore, on the one hand,

$$|\mathcal{F}[P,Q]| \leq \prod_{i=1}^{t}(1+\widehat{\epsilon})T_i = (1+\widehat{\epsilon})^t \cdot T = (1+\frac{\ln(1+\epsilon)}{t})^t \cdot T \leq e^{\ln(1+\epsilon) \cdot T} = (1+\epsilon)\cdot T.$$

On the other hand,

$$|\mathcal{F}[P,Q]| \geq \prod_{i=1}^{t}(1-\widehat{\epsilon})T_i = (1-\widehat{\epsilon})^t \cdot T = (1-\frac{\ln(1+\epsilon)}{t})^t \cdot T \geq (1-\ln(1+\epsilon))\cdot T \geq (1-\epsilon)\cdot T.$$

Here, the inequality $(1-\frac{\ln(1+\epsilon)}{t})^t \geq (1-\ln(1+\epsilon))$ follows since the larger $t$ is (starting at 1), the larger the value of $(1-\frac{\ln(1+\epsilon)}{t})^t$ (approaching $e^{-\ln(1+\epsilon)}$), and the inequality $\ln(1+\epsilon) \leq \epsilon$ follows from Taylor series. Because the choice of the disjoint sets $P \in \mathcal{P}^{\mathrm{BAL}}$ and $Q \in \mathcal{Q}^{\mathrm{CBAL}}$ was arbitrary, the proof is complete. □

To devise an efficient membership query procedure, we also need to upper bound, for any set $P$, the number of sets in $\mathcal{F}$ that contain $P$. We consider any choice of $P$ of size $p' \leq p$ rather than just any choice of $P$ of size exactly $p$ to have general membership procedures as required for Section 5.

**Lemma 3.8.** *Let $t,k,p,b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f,g)$ be a $(t,k,p,b)$-splitting function pair. With probability at least $1-\frac{1}{2c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling has the following property: For every $g'$ be such that $(f,g')$ is a $(t,k,p',b)$-splitting function pair (for some $p' \leq p$) where $g' \leq g$ and set $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$, we have that $|\{F \in \mathcal{F} : P \subseteq F\}| \leq (\frac{d \cdot k}{d \cdot k - p})^{k-p} \cdot (\frac{d \cdot k}{p})^{p-p'} \cdot (\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc))^t$.*

*Proof.* Towards the proof of the lemma, we first show that the following claim is correct.

**Claim 3.2.** *With probability at least $1-\frac{1}{2c}$, for every $i \in \{1,2,\ldots,t\}$, $g'(i) \leq g(i)$ and $P \in \binom{U_i}{g'(i)}$, we have that $|\{F \in \mathcal{F}_i : P \subseteq F\}| \leq (\frac{d \cdot f(i)}{d \cdot f(i) - g(i)})^{f(i)-g(i)} \cdot (\frac{d \cdot f(i)}{g(i)})^{g(i)-g'(i)} \cdot \frac{1}{\ln^2(1+\epsilon)}$ $\cdot 20k^3 \cdot \ln(nc)$.*

*Proof.* Denote $E_i = (\frac{d \cdot f(i)}{d \cdot f(i) - g(i)})^{f(i)-g(i)} \cdot (\frac{d \cdot f(i)}{g(i)})^{g(i)-g'(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc)$. By union bound and because $|\binom{U_i}{\leq p}| \leq n^k$, it suffices to choose some $i \in \{1,2,\ldots,t\}, g'(i) \leq g(i)$ and $P \in \binom{U_i}{g'(i)}$, and prove that with failure probability at most $\frac{1}{2ctn^k}$, we have that $|\{F \in \mathcal{F}_i : P \subseteq F\}| \leq E_i$. To this end, observe that each set $F_{i,j} \in \mathcal{F}_i$ contains $P$ with probability $(\frac{g(i)}{d \cdot f(i)})^{g'(i)}$. Thus, the expected number of sets in $\mathcal{F}_i$ that contain $P$ is $E_i$. Because the sets in $\mathcal{F}_i$ are sampled independently from one another, by Chernoff bound (Proposition 2.2), we have that

$$\begin{aligned} Pr(||\mathcal{F}_i[P,Q]| - E_i| > E_i) &\leq 2e^{-\frac{E_i}{2}} \\ &\leq 2e^{-5k \cdot \ln(nc)} = \frac{2}{(nc)^{5k}} \leq \frac{2}{n^4 \cdot n^k \cdot c} \leq \frac{1}{2ct} \end{aligned}$$

Here, the last inequality follows since $n \geq \max(2,t)$. This completes the proof of the claim. □

We now return to the proof of the lemma. Due to Claim 3.1, to prove the lemma it suffices to show that, under the assumption that for every $i \in \{1,2,\dots,t\}$, $g'(i) \leq g(i)$ and $P \in \binom{U_i}{g'(i)}$, we have that $|\{F \in \mathcal{F}_i : P \subseteq F\}| \leq (\frac{d\cdot f(i)}{d\cdot f(i)-g(i)})^{f(i)-g(i)} \cdot (\frac{d\cdot f(i)}{g(i)})^{g(i)-g'(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)$, it holds that for every $g'$ be such that $(f,g')$ is a $(t,k,p',b)$-splitting function pair where $g' \leq g$ and set $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$, we have that $|\{F \in \mathcal{F} : P \subseteq F\}| \leq \frac{(d\cdot k)^{k-p}}{(d\cdot k-p)^{k-p}} \cdot (\frac{d\cdot k}{p})^{p-p'} \cdot (\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc))^t$. Towards the proof of this, consider some set $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$. Then,

$$|\{F \in \mathcal{F} : P \subseteq F\}| = \prod_{i=1}^{t} |\{F \in \mathcal{F}_i : P \cap U_i \subseteq F\}|.$$

Because $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$, it holds that for every $i \in \{1,\dots,t\}$, $P \cap U_i \in \binom{U_i}{g'(i)}$, and therefore $|\{F \in \mathcal{F}_i : P\cap U_i \subseteq F\}| \leq (\frac{d\cdot f(i)}{d\cdot f(i)-g(i)})^{f(i)-g(i)} \cdot (\frac{d\cdot f(i)}{g(i)})^{g(i)-g'(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)$. Thus,

$$\begin{aligned}|\{F \in \mathcal{F} :\ & P \subseteq F\}| \\ & \leq \prod_{i=1}^{t}\left( (\frac{d\cdot f(i)}{d\cdot f(i)-g(i)})^{f(i)-g(i)} \cdot (\frac{d\cdot f(i)}{g(i)})^{g(i)-g'(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)\right) \\ & \leq \left(\prod_{i=1}^{t}(\frac{d\cdot f(i)}{d\cdot f(i)-g(i)})^{f(i)-g(i)} \cdot (\frac{d\cdot f(i)}{g(i)})^{g(i)-g'(i)}\right) \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)\right)^t.\end{aligned}$$

Recall that $f : \{1,2,\dots,t\} \to \{1,2,\dots,\lceil bk/t\rceil\}$ and $g' \leq g \leq f$ satisfy $\sum_{i=1}^{t} f(i) = k$, $\sum_{i=1}^{t} g(i) = p$ and $\sum_{i=1}^{t} g'(i) = p'$. Relaxing the supposition $f : \{1,2,\dots,t\} \to \{1,2,\dots,\lceil bk/t\rceil\}$ to $f : \{1,2,\dots,t\} \to \{1,2,\dots,k\}$, the maximum of $\prod_{i=1}^{t}(\frac{d\cdot f(i)}{d\cdot f(i)-g(i)})^{f(i)-g(i)} \cdot (\frac{d\cdot f(i)}{g(i)})^{g(i)-g'(i)}$ is attained when $f(i) = k$, $g(i) = p$ and $g'(i) = p'$ for some $i \in \{1,2,\dots,t\}$, and $f(i') = g(i') = g'(i') = 0$ for all other $i' \in \{1,2,\dots,t\} \setminus \{i\}$. Then, the value is $\frac{(d\cdot k)^{k-p}}{(d\cdot k-p)^{k-p}} \cdot (\frac{d\cdot k}{p})^{p-p'}$. This completes the proof. □

The property in Lemma 3.7 together with the product-like manner in which we construct $\mathcal{F}$ yields an efficient membership query procedure as follows.

**Definition 3.10 (Membership Query Procedure for Parsimonious Universal Family Sampling).** *Let $t,k,p,b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c,d \geq 1$. Let $\overline{\mathbf{U}} = (U_1,U_2,\dots,U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$. Let $(f,g)$ be a $(t,k,p,b)$-splitting function pair. Let $\mathcal{F} \subseteq 2^U$ be the output family of $(\overline{\mathbf{U}},b,f,g,\epsilon,c,d)$-universal family sampling. Then, the procedure* MEMBERSHIP *is defined as follows. Let $\{\mathcal{F}_i\}_{i=1}^{t}$ be the collection of families sampled to construct $\mathcal{F}$ (see Definition 3.9). Given $g'$ such that $(f,g')$ is a $(t,k,p',b)$-splitting function pair (for some $p' \leq p$) where $g' \leq g$ and $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$,* MEMBERSHIP *naively computes $\mathcal{F}'_i = \{F_{i,j_i} \in \mathcal{F}_i : P \cap U_i \subseteq F_{i,j_i}\}$ by iterating over every set in $\mathcal{F}_i$; then, it outputs $\{F_{1,j_1} \cup F_{2,j_2} \cup \dots \cup F_{t,j_t} : F_{1,j_1} \in \mathcal{F}'_1, F_{2,j_2} \in \mathcal{F}'_2, \dots, F_{t,j_t} \in \mathcal{F}'_t\}$, computed using naive enumeration.*

We now assert that our procedure is indeed an efficient membership query procedure as a corollary of Lemma 3.8.

**Corollary 3.2.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$. Let $(f, g)$ be a $(f, g)$ be a $(t, k, p, b)$-splitting function pair. Let $\mathcal{F} \subseteq 2^U$ be the output family of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling. Then, with probability at least $1 - \frac{1}{2c}$, for every $g'$ be such that $(f, g')$ is a $(t, k, p', b)$-splitting function pair (for some $p' \leq p$) where $g' \leq g$, the procedure* MEMBERSHIP *is a $T$-membership query procedure with respect to $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$ for*

$$T = \left((d \cdot bk)^{bk/t} + (\frac{d \cdot k}{d \cdot k - p})^{k-p} \cdot (\frac{d \cdot k}{p})^{p-p'}\right) \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)\right)^t.$$

*Proof.* Let $X = (\frac{d \cdot k}{d \cdot k - p})^{k-p} \cdot (\frac{d \cdot k}{p})^{p-p'} \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)\right)^t$. The claim that MEMBERSHIP is a membership query procedure (i.e., that given $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$, the output is indeed $\{F \in \mathcal{F} : P \subseteq F\}$) is immediate from the definition of $\mathcal{F}$. Further, by Lemma 3.8, $\max_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}} |\{F \in \mathcal{F} : P \subseteq F\}| \leq X$ with probability at least $1 - \frac{1}{2c}$. Now, under the assumption that the aforementioned inequality holds, consider any set $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$. Then, the running time of MEMBERSHIP is bounded by

$$\begin{aligned}
&\mathcal{O}(\sum_{i=1}^{t} s_i + |\{F \in \mathcal{F} : P \subseteq F\}|) \\
&= \mathcal{O}(\sum_{i=1}^{t} \frac{(d \cdot f(i))^{f(i)}}{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) \\
&= \mathcal{O}(\sum_{i=1}^{t} (d \cdot f(i))^{f(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) \\
&= \mathcal{O}(\sum_{i=1}^{t} (d \cdot bk/t)^{bk/t} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) \\
&= \mathcal{O}(t \cdot (d \cdot bk)^{bk/t} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) = \mathcal{O}(T).
\end{aligned}$$

□

By putting together Observation 3.3, Lemma 3.6, Lemma 3.7 and Corollary 3.2, we derive our main statement regarding the produced family $\mathcal{F}$.

**Theorem 3.2.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. With probability at least $1 - \frac{1}{c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling, computed in time $\mathcal{O}(|\mathcal{F}|n)$, satisfies all of the following conditions.*

1. $|\mathcal{F}| \leq \frac{(dk)^k}{p^p(dk-p)^{k-p}} \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc)\right)^t$.
2. *$\mathcal{F}$ is an $\epsilon$-parsimonious $(n, p, k-p)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}, \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g})$, whose correction factor is upper bounded by* $\left(\frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc)\right)^t$.
3. *With respect to $\mathcal{F}$ and any $g'$ be such that $(f, g')$ is a $(t, k, p', b)$-splitting function pair (for some $p' \leq p$) where $g' \leq g$,* MEMBERSHIP *is a $T$-membership query procedure with respect to $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$ for*

$$T = \left((d \cdot bk)^{bk/t} + (\frac{dk}{dk-p})^{k-p}(\frac{dk}{p})^{p-p'}\right) \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)\right)^t.$$

In Section 4, we will be interested only in the case where $b = 2$, $\epsilon = \frac{\ln\frac{3}{2}}{5k^2}$, $p' = p$ and $d = 1.447 = \mathcal{O}(1)$; later, we will run our entire process multiple times to enable having arbitrarily small error. Then, $(\frac{1}{\ln^2(1+\epsilon)})^t \leq (\epsilon - \epsilon^2/2)^t$ (by Taylor series), upper bounded by $2^{\mathcal{O}(\sqrt{k}\log k)}$. Further, we will choose $c \geq n$ and $t = \lceil\sqrt{k}\rceil$. By these substitutions, we obtain the following corollary of Theorem 3.2.

**Corollary 3.3.** *Let $k, p \in \mathbb{N}$ with $p \leq k$, and $c \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \dots, U_{\lceil\sqrt{k}\rceil})$ be a partitioned universe with $U = \bigcup_{i=1}^{\lceil\sqrt{k}\rceil} U_i$ of size $n \leq c$, and let $(f, g)$ be a splitting function pair. With probability at least $1 - \frac{1}{c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, 2, f, g, \frac{\ln\frac{3}{2}}{5k^2}, c, 1.447)$-universal family sampling, computed in time $\mathcal{O}(|\mathcal{F}|n)$, satisfies all of the following conditions.*

1. *$|\mathcal{F}| \leq \dfrac{(1.447k)^k}{p^p(1.447k-p)^{k-p}} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}} c$.*

2. *$\mathcal{F}$ is a $\frac{\ln\frac{3}{2}}{5k^2}$-parsimonious $(n, p, k-p)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}, \mathcal{Q}^{\mathrm{CBAL}})$.*

3. *With respect to $\mathcal{F}$,* MEMBERSHIP *is a $T$-membership query procedure for*

$$T = \left(\frac{1.447k}{1.447k-p}\right)^{k-p} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}} c.$$

## 3.3 Reducing a Problem to Its Split Version

Because we only deal with balancedly split sets, we now develop a simple procedure whose employment will allow us to reduce the general case to one focused only on balancedly split sets. To this end, we need the following definition.

**Definition 3.11 (Balancedly Split Sets II).** *Let $t, k, b \in \mathbb{N}$. Let $\overline{\mathbf{U}} = (U_1, U_2, \dots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$. Then, $P \in \binom{U}{k}$ is* $(\overline{\mathbf{U}}, k, b)$-balancedly split *if for every $i \in \{1, 2, \dots, t\}$, it holds that $|P \cap U_i| \leq \lceil bk/t \rceil$.*

We now present the procedure.

**Lemma 3.9.** *Given $t, k, b \in \mathbb{N}$ and $c \geq 1$, a universe $U$ of size $n$, and $0 < \delta < 1$ with $b^2\frac{k}{2t} \geq \ln(4t)$, a collection $\mathcal{U}$ of $\frac{4}{\delta^2}k\ln(2nc)$ $t$-partitioned universes over $U$ such that the following property holds with probability at least $1 - \frac{1}{c}$ (resp. 1) can be computed in time $\mathcal{O}(n\frac{1}{\delta^2}k\ln(nc))$: for every set $P \in \binom{U}{k}$, the (resp. expected) number of partitioned universes $\overline{\mathbf{U}} \in \mathcal{U}$ such that $P$ is $(\overline{\mathbf{U}}, k, b)$-balancedly split is between $(1-\delta)X$ and $(1+\delta)X$ (resp. exactly $X$) for some $X = X(n, k, t, b, \delta) > 0$. (We note that $X$ can be computed in time $\mathcal{O}(|\mathcal{U}| \cdot (\frac{2bk}{t})^t)$).*

*Proof.* Denote $r = \frac{4}{\delta^2}k\ln(2nc)$. Given the input $t, k, c, U, b, \delta$, the algorithm constructs $\mathcal{U} = \{\overline{\mathbf{U}}_1, \overline{\mathbf{U}}_2, \dots, \overline{\mathbf{U}}_r\}$ as follows. For $i = 1, 2, \dots, r$, the partitioned universe $\overline{\mathbf{U}}_i = (U_{i,1}, U_{i,2}, \dots, U_{i,t})$ is constructed as follows. Each element $u \in U$ is inserted into exactly one part $U_{i,j}$ where the choices of $j \in \{1, 2, \dots, t\}$ are made independently and uniformly at random. Clearly, the time complexity of the algorithm is $\mathcal{O}(nr)$.

Let $X$ denote the expected number of partitioned universes $\overline{\mathbf{U}} \in \mathcal{U}$ such that any set $P \in \binom{U}{k}$ is $(\overline{\mathbf{U}}, k, b)$-balancedly split. Note that $X$ is the same for all sets $P \in \binom{U}{k}$, thus it is well defined. The exact value of $X$ will be calculated later.

Now, arbitrarily choose some set $P \in \binom{U}{k}$. Additionally, consider some $i \in \{1, 2, \dots, r\}$. Notice that for any $j \in \{1, 2, \dots, t\}$, the expected number of elements in $P$ contained in $U_{i,j}$ is $k/t$, therefore Chernoff bound (Proposition 2.2) implies that the probability that the number of elements in $P$ contained in $U_{i,j}$ is not upper bounded by $\lceil bk/t \rceil$ is at most $2e^{\frac{-b^2(k/t)}{2}} \leq 2e^{-\ln(4t)} =$

$\frac{1}{2t}$ where the inequality follows from the supposition $b^2\frac{k}{2t} \geq \ln(4t)$ in the lemma. Then, by union bound, the probability that $P$ is not $(\overline{\mathbf{U}}, k, b)$-balancedly split is at most $t \cdot \frac{1}{2t} = \frac{1}{2}$, hence the probability that it is $(\overline{\mathbf{U}}, k, b)$-balancedly split is at least $\frac{1}{2}$. Therefore, $X \geq \frac{r}{2}$. In turn, by Chernoff bound (Proposition 2.2) and this lower bound on $X$, the probability that the number of partitioned universes $\overline{\mathbf{U}} \in \mathcal{U}$ such that $P$ is $(\overline{\mathbf{U}}, k, b)$-balancedly split is not between $(1-\delta)X$ and $(1+\delta)X$ is at most $2e^{-\frac{\delta^2 X}{2}} \leq 2e^{-\frac{\delta^2 r}{4}} = 2e^{-k\ln(2nc)} = \frac{2}{(2nc)^k} \leq \frac{1}{n^k c}$.

Since the choice of $P \in \binom{U}{k}$ was arbitrary and by union bound, the probability that there exists $P \in \binom{U}{k}$ such that the number of partitioned universes $\overline{\mathbf{U}} \in \mathcal{U}$ such that $P$ is $(\overline{\mathbf{U}}, k, \epsilon)$-balancedly split is not between $(1-\delta)X$ and $(1+\delta)X$ is upper bounded by $\binom{n}{k} \cdot \frac{1}{n^k c} \leq \frac{1}{c}$. Thus, with probability at least $1 - \frac{1}{c}$, for every set $P \in \binom{U}{k}$ the number of partitioned universes $\overline{\mathbf{U}} \in \mathcal{U}$ such that $P$ is $(\overline{\mathbf{U}}, k, b)$-balancedly split is between $(1-\delta)X$ and $(1+\delta)X$.

It remains to calculate $X$. To this end, arbitrarily choose some set $P \in \binom{U}{k}$ and $i \in \{1, 2, \ldots, r\}$. Clearly, $X = r \cdot Y$, where $Y$ is the probability that $P$ is $(\overline{\mathbf{U}}_i, k, b)$-balancedly split. Now, observe that

$$Y = \sum_{\substack{\ell_1, \ell_2, \ldots, \ell_t \in \{1,2,\ldots,\lceil bk/t \rceil\} \\ \text{s.t. } \sum_{j=1}^{t} \ell_j = k}} \binom{k}{\ell_1} \cdot \binom{k-\ell_1}{\ell_2} \cdots \binom{k - \sum_{j=1}^{t-1} \ell_j}{\ell_t} \cdot (1/t)^k.$$

This completes the proof. □

We now present the our main utility of this procedure, which is a reduction of a problem to a "split" version of itself. To this end, we first define the notion of a split version of a problem.

**Definition 3.12 (Splittable Problem).** *Let $\Pi$ be a problem whose input consists, among possibly other components, of a universe $U$ of size $n$ and $k \in \mathbb{N}$, and whose solutions are subsets (resp. ordered subsets) of $U$ of size $k$. Such a problem $\Pi$ is said to be* splittable. *Then, the* general split version of $\Pi$ *is defined as follows. Its input consists of the same components as the input of $\Pi$, and in addition, of a $t$-partitioned universe $\overline{\mathbf{U}}$ for some $t \in \mathbb{N}$, $b \in \mathbb{N}$ and a $(t, k, b)$-splitting function $f$, and whose solutions are all the subsets (resp. ordered subsets) of $U$ that are both solutions of $\Pi$ and are $(\overline{\mathbf{U}}, f)$-balancedly split. When $t = \sqrt{k}$ and $b = 2$, the general split version is called the* split version *in short.*

Next, we present the reduction.

**Lemma 3.10.** *Let $\Pi$ be a splittable problem such that the number of solutions of the general split version of $\Pi$ can be approximately counted with multiplicative error $(1 \pm \alpha)$ (resp. and the expectation equals the exact number of solutions) in time $T = T(\alpha, t, b)$ (where $t, b$ are input to the split version) and with success probability at least $1 - \frac{1}{c'}$. Then, for any $c \in \mathbb{N}$ such that $(2bk/t)^t \cdot \frac{1}{\beta^2} k \ln(nc) \cdot \frac{1}{c'} \leq \frac{1}{2c}$ and $0 < \beta < 1$, the number of solutions of $\Pi$ can be approximately counted with multiplicative error $(1 \pm \alpha)(1 \pm \beta)$ (resp. and the expectation equals the exact number of solutions) in time $\mathcal{O}(((2bk/t)^t \cdot T + n) \cdot \frac{1}{\beta^2} k \ln(nc))$ where $b^2\frac{k}{2t} \geq \ln(4t)$ and with success probability at least $1 - \frac{1}{c}$.*

*Proof.* Let ALG1 be the algorithm supposed to approximately count solutions of the general split version of $\Pi$ with multiplicative error $(1 \pm \alpha)$ where the expectation equals the exact number of solutions in time $T$ and with success probability at least $1 - \frac{1}{c'}$. We remark that if the condition regarding the expectation is not assumed to hold, then disregard the arguments below concerning its satisfaction for the output. Then, we design an algorithm ALG2 as follows. Given an instance $I$ of $\Pi$, $c \in \mathbb{N}$ and $0 < \beta < 1$, ALG2 executes the following operations.

1. Use the algorithm in Lemma 3.9 to compute a collection $\mathcal{U}$ of $\frac{4}{\beta^2}k\ln(4nc)$ $t$-partitioned universes over $U$ such that the following property holds with probability at least $1-\frac{1}{2c}$ (resp. 1): for every set $P \in \binom{U}{k}$, the (resp. expected) number of partitioned universes $\overline{\mathbf{U}} \in \mathcal{U}$ such that $P$ is $(\overline{\mathbf{U}}, k, b)$-balancedly split is between $(1-\beta)X$ and $(1+\beta)X$ (resp. exactly $X$) for some $X = X(n,k,t,b,\beta) > 0$.
2. Let $\mathcal{F}$ be the family of all $(t,k,b)$-splitting functions.
3. For every partitioned universe $\overline{\mathbf{U}} \in \mathcal{U}$:
   (a) For every $f \in \mathcal{F}$:
      i. Run ALG1 on $(I, \overline{\mathbf{U}}, b, f)$ as input, and denote its output by $O_{\overline{\mathbf{U}},f}$.
   (b) Let $O_{\overline{\mathbf{U}}} = \displaystyle\sum_{f\in\mathcal{F}} O_{\overline{\mathbf{U}},f}$.
4. Output $O = \dfrac{1}{X} \cdot \displaystyle\sum_{\overline{\mathbf{U}}\in\mathcal{U}} O_{\overline{\mathbf{U}}}$.

By Lemma 3.9, Step 1 is performed in time $\mathcal{O}(n\frac{1}{\beta^2}k\ln(nc))$. Now, observe that $|\mathcal{F}| \leq (\lceil bk/t\rceil + 1)^t = \mathcal{O}((2bk/t)^t)$. Thus, we perform Step 3(a)i $|\mathcal{U}|\cdot|\mathcal{F}| = \mathcal{O}(\frac{1}{\beta^2}k\ln(nc)\cdot(2bk/t)^t)$ times, where each single performance is done in time $\mathcal{O}(T)$. Thus, the total running time is indeed $\mathcal{O}(((2bk/t)^t \cdot T + n)\cdot\frac{1}{\beta^2}k\ln(nc))$.

By union bound, with probability at least $1-|\mathcal{U}||\mathcal{F}|\cdot\frac{1}{c'}-\frac{1}{2c}$, which is lower bounded by $1-(2bk/t)^t\cdot\frac{1}{\beta^2}k\ln(nc)\cdot\frac{1}{c'}-\frac{1}{2c} \geq 1-\frac{1}{c}$, the call to the algorithm in Lemma 3.9 as well as all calls to ALG2 are successful. Thus, to prove the lemma, it suffices to prove that $E[O]$ is the exact number of solutions, and that under the aforementioned condition (of all calls being successful), the number of solutions of $\Pi$ is necessarily approximated by $O$ with multiplicative error $(1\pm\alpha)(1\pm\beta)$.

First, observe that for any $\overline{\mathbf{U}} \in \mathcal{U}$, the number of $(\overline{\mathbf{U}}, k, b)$-balancedly split solutions is exactly the sum over all $f \in \mathcal{F}$ of the number of $(\overline{\mathbf{U}}, f)$-balancedly split solutions. Thus, because the approximation factor of ALG2 is $(1\pm\alpha)$ and the expectation is exact, we have that for any $\overline{\mathbf{U}} \in \mathcal{U}$, the number of $(\overline{\mathbf{U}}, k, b)$-balancedly split solutions is exactly $E[O_{\overline{\mathbf{U}}}]$, and (under the aforementioned condition) it is approximated by $O_{\overline{\mathbf{U}}}$ with multiplicative error $(1\pm\alpha)$. Now, recall that for every set $P \in \binom{U}{k}$ (and, in particular, for every solution of $\Pi$), the number of partitioned universes $\overline{\mathbf{U}} \in \mathcal{U}$ such that $P$ is $(\overline{\mathbf{U}}, k, b)$-balancedly split is in expectation $X$, and (under the aforementioned condition) it is between $(1-\beta)X$ and $(1+\beta)X$. Since $O = \dfrac{1}{X}\cdot\displaystyle\sum_{\overline{\mathbf{U}}\in\mathcal{U}} O_{\overline{\mathbf{U}}}$, we conclude that indeed the number of solutions of $\Pi$ is $E[O]$, and that (under the aforementioned condition) it is necessarily approximated by $O$ with multiplicative error $(1\pm\alpha)(1\pm\beta)$. □

For the (non-general) split version and $\alpha = \beta = \frac{1}{2}$, in which we will be specifically interested, we obtain the following corollary.

**Corollary 3.4.** *Let $\Pi$ be a splittable problem such that the number of solutions of the split version of $\Pi$ can be approximately counted with multiplicative error $(1\pm\frac{1}{2})$ where the expectation equals the exact number of solutions in time $T$ and with success probability at least $1-\frac{1}{c'}$. Then, for any $c \in \mathbb{N}$ such that $4k(4\sqrt{k})^{\sqrt{k}}\ln(nc)\cdot\frac{1}{c'} \leq \frac{1}{c}$, the number of solutions of $\Pi$ can be approximately counted with multiplicative error between $\frac{1}{4}$ and $2\frac{1}{4}$ where the expectation equals the exact number of solutions in time $\mathcal{O}((2^{\mathcal{O}(\sqrt{k}\log k)}\cdot T + n)\cdot k\ln(nc))$ and with success probability at least $1-\frac{1}{c}$.*

Lastly, we give a lemma that can be considered folklore (but whose proof is given for completeness), whose utility is to enable us to focus on achieving some small constant multiplicative error for a counting problem, as this can be boosted to an arbitrarily small error as follows.

**Lemma 3.11.** *Let $\Pi$ be a problem that admits a randomized algorithm that, given an instance of $\Pi$ whose number of solutions is $X$, returns a number $Y$ such that $E[Y]=X$ and $\alpha X \leq Y \leq \beta X$ for some $0<\alpha\leq 1$ and $\beta \geq 1$ in time $T$ with success probability $1-\frac{1}{c'}$. Then, for any $0<\epsilon<1$ and $c\geq 1$ such that $\frac{t}{c'}\leq\frac{1}{2c}$ where $t=\frac{2\beta}{\epsilon^2}\lceil\ln(4c)\rceil$, $\Pi$ also admits an algorithm that, given an instance of $\Pi$ whose number of solutions is $X$, returns a number $Z$ such that $(1-\epsilon)X\leq Z\leq(1+\epsilon)X$ in time $\mathcal{O}(\frac{\beta}{\epsilon^2}\log c\cdot T)$ with success probability at least $1-\frac{1}{c}$.*

*Proof.* Let ALG1 denote the algorithm given in the supposition of the lemma. Let $0<\epsilon<1$. Then, we design an algorithm ALG2 as follows. Given an instance $I$ of $\Pi$, ALG2 executes the following operations.

1. For $i=1,2,\ldots t$: Call ALG1 with $I$ as input and let $Y_i$ denote the result.
2. Output $Z=\frac{1}{t}\cdot\sum_{i=1}^{t}Y_i$.

First, notice that the time complexity of ALG2 is $\mathcal{O}(t\cdot T)=\mathcal{O}(\frac{\beta}{\epsilon^2}\log c\cdot T)$. Second, by union bound, with success probability at least $1-\frac{t}{c'}\geq 1-\frac{1}{2c}$, all the calls it makes to ALG1 are successful. Thus, by union bound, to prove the lemma, it suffices to prove that under the assumption that all the calls made to ALG1 are successful, with probability at least $1-\frac{1}{2c}$, it holds that $(1-\epsilon)X\leq Z\leq(1+\epsilon)X$.

For all $i\in\{1,2,\ldots,t\}$, denote $Y'_i=\frac{Y_i}{\beta X}$. Moreover, denote $Z'=\sum_{i=1}^{t}Y'_i$. Notice that $(1-\epsilon)X\leq Z\leq(1+\epsilon)X$ if and only if $(1-\epsilon)\frac{t}{\beta}\leq Z'\leq(1+\epsilon)\frac{t}{\beta}$, and thus it suffices to consider the probability that the latter event occurs. Since all calls are assumed to be successful, we have that $0\leq Y'_i\leq 1$. Moreover, by linearity of expectation, $E[Z']=\sum_{i=1}^{t}E[Y'_i]=\sum_{i=1}^{t}\frac{E[Y_i]}{\beta X}=t/\beta$. Therefore, $(1-\epsilon)\frac{t}{\beta}\leq Z'\leq(1+\epsilon)\frac{t}{\beta}$ if and only if $|Z'-E[Z']|\leq\epsilon E[Z']$, and thus it further suffices to consider the probability that the latter event occurs. By Chernoff Bound (Proposition 2.2), we have that

$$\begin{aligned} Pr(|Z'-E[Z']|>\epsilon E[Z']) &\leq 2e^{-\frac{\epsilon^2 E[Z']}{2}}\\ &=2e^{-\frac{\epsilon^2 t}{2\beta}}\\ &=2e^{-\ln(4c)}=\frac{1}{2c}.\end{aligned}$$

Thus, $|Z'-E[Z']|\leq\epsilon E[Z']$ with probability at least $1-\frac{1}{2c}$. As claimed above, this completes the proof. □

Combining Corollary 3.4 and Lemma 3.11, we have the following read-to-use corollary. We did not make any attempt to optimize the lower bound on $c'$, but just give a short expression. Clearly, the success probability can be boosted to any constant close to 1. To simplify notation, we will work with $\frac{9}{10}$.

**Corollary 3.5.** *Let $\Pi$ be a splittable problem such that the number of solutions of the split version of $\Pi$ can be approximated with multiplicative error $(1\pm\frac{1}{2})$ in time $T\geq n$ where the expectation equals the exact number of solutions, and with success probability at least $1-\frac{1}{c'}$. Then, for any $0<\epsilon<1$ such that $c'\geq\frac{1}{\epsilon^2}\cdot(1000\sqrt{k})^{\sqrt{k}}\cdot\ln(n\frac{1}{\epsilon})$, the number of solutions of $\Pi$ can be approximated with multiplicative error $(1\pm\epsilon)$ in time $2^{\mathcal{O}(\sqrt{k}\log k)}\cdot T\cdot\frac{1}{\epsilon^2}(\log n+\log\frac{1}{\epsilon}))$ and with success probability at least $\frac{9}{10}$.*

*Proof.* Denote $c'' = 2ct$ where $c = 10$, $\alpha = \frac{1}{4}$, $\beta = 2\frac{1}{4}$ and $t = \frac{2\beta}{\epsilon^2}\ln(4c)$. Then, $4k(4\sqrt{k})^{\sqrt{k}}\ln(nc'')\cdot \frac{1}{c'} \leq \frac{1}{c''}$. Thus, by Corollary 3.4, the number of solutions of $\Pi$ can be approximately counted with multiplicative error between $\frac{1}{4}$ and $2\frac{1}{4}$ in time $T' = \mathcal{O}((2^{\mathcal{O}(\sqrt{k}\log k)} \cdot T + n) \cdot k\ln(nc''))$ and with success probability at least $1 - \frac{1}{c''}$. Therefore, by Lemma 3.11, the number of solutions of $\Pi$ can be approximated with multiplicative error $(1 \pm \epsilon)$ in time $\mathcal{O}(\frac{\beta}{\epsilon^2}\log c \cdot T') = 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot T \cdot \frac{1}{\epsilon^2}(\log n + \log \frac{1}{\epsilon}))$ and with success probability at least $\frac{9}{10}$. □

# 4 Main Applications

We will first consider the problem of approximately counting the number of multilinear monomials of a polynomial encoded by a "skewed" arithmetic circuit. Because many problems are known to be reducible to the aforementioned problem where the output arithmetic circuit is "skewed" and the reduction is parsimonious, we will immediately derive a large number of applications.

## 4.1 An Algorithm for #Multilinear Monomial Detection on Skewed Circuits

In the Multilinear Monomial Detection problem, the input consists of an arithmetic circuit $C$ over $\mathbb{Z}^+$ with variable set $X = \{x_1, \ldots, x_n\}$ representing a polynomial $P_C(X)$ over $\mathbb{Z}$, and the objective is to decide whether $P_C(X)$ construed as a sum of monomials contains a multilinear monomial of degree $k$. In the approximate counting version called #Multilinear Monomial Detection, we aim to approximate $\sum_{m \in \texttt{MulLin}_k(P_C(X))} \texttt{coeff}(P_C(X), m)$ where $\texttt{MulLin}_k(P_C(X))$ is the set of multilinear monomials of $P_C(X)$ of degree $k$ and $\texttt{coeff}(P_C(X), m)$ is the coefficient of $m$.

Given a node $v$ of an arithmetic circuit $C$, we denote by $C_v$ the sub-arithmetic circuit of $C$ defined by the subdigraph of $C$ induced by the set of nodes reachable from $v$ in $C$; note that $v$ is the root of $C_v$. For any integer $d \in \mathbb{N}$, we say that a node $v$ of an arithmetic circuit $C$ is *$d$-skewed* if the number of distinct (including both multilinear and non-multilinear) monomials (with non-zero coefficient) of the polynomial $P_{C_v}(X)$ represented by $C_v$ is at most $d$. Notice that if a node if $d$-skewed, then all of its children are $d$-skewed as well. Moreover, we say that an arithmetic circuit is *$d$-multiplication skewed* (or, in short, *$d$-skewed*) if for every multiplication node $v$ of $X$ (which, by our assumption, has two children), at most one of the children of $v$ is not $d$-skewed. We also denote by $\texttt{deg}_{\text{add}}(C)$ the maximum number of multiplication nodes that can be reached from an addition node in $C$ while traversing internally only addition nodes. When all addition nodes can only have outgoing arcs to multiplication nodes, $\texttt{deg}_{\text{add}}(C)$ is upper bounded by the maximum outgoing degree of an addition node in $C$. Clearly, in general, $\texttt{deg}_{\text{add}}(C) \leq s(C)$.

We will be using the following folklore proposition. It can be obtained by straightforward dynamic programming, where at each node of the circuit, we explicitly store the polynomial corresponding to the sub-circuit rooted at it, where when we reach a polynomial having more than $d$ distinct monomials, we do not continue the computation for that node (notice that the monomials of the polynomial can be computed one by one, so if we reach monomial $d+1$, we stop and assign null).

**Proposition 4.1** (Folklore)**.** *On $d$-skewed circuits for any $d \in \mathbb{N}$, the following can be computed in time $\mathcal{O}(s(C)d)$: For every node $v$ in $C$, if $v$ is $d$-skewed, then compute $W_v := P_v$, and otherwise compute $W_v := null$.*

We now present our main application. We remark that the success probability can be boosted to any constant below 1. Further, we can deal with $d$-skewed circuits also when $d$ is not bounded by $2^{o(k)}\cdot s(C)^{\mathcal{O}(1)}$, though this will worsen the running time (depending on how much larger $d$ is).[4] In our applications, $\ell=0$, and thus we emphasize the time obtained in that case.

**Theorem 4.1.** *For any $0<\epsilon<1$ and $\ell\geq 0$, the #Multilinear Monomial Detection problem on $2^{o(k)}s(C)^{\ell}$-skewed circuits can be approximated with factor $(1\pm\epsilon)$ and success probability at least $\frac{9}{10}$ in time $\mathcal{O}((2.619^k+s(C)^{o(1)})\cdot\frac{1}{\epsilon^2}\cdot s(C)^{1+\ell}\mathtt{deg}_{\text{add}}(C))$. In particular, when $\ell=0$, the time is $\mathcal{O}((2.619^k+s(C)^{o(1)})\cdot\frac{1}{\epsilon^2}\cdot s(C)\mathtt{deg}_{\text{add}}(C))$.*

The correctness will follow from Lemma 4.1 below as we will show immediately. Notice that in the splittable version of #Multilinear Monomial Detection, the value to approximate is $\sum_{m\in\widetilde{\mathtt{MulLin}}_k(P_C(X))}\mathtt{coeff}(P_C(X),m)$ where $\widetilde{\mathtt{MulLin}}_k(P_C(X))$ is the set of multilinear monomials of $P_C(X)$ of degree $k$ where for every $i\in\{1,2,\ldots,\sqrt{k}\}$, the number of variables that are part of the monomial and belong to $U_i$ is $f(i)$, for $\overline{\mathbf{U}}=(U_1,U_2,\ldots,U_{\sqrt{k}})$ being the input a $\sqrt{k}$-partitioned universe (where $\bigcup_{i=1}^{\sqrt{k}}U_i=X$) and $f$ being the input $(\sqrt{k},k,2)$-splitting function.

**Lemma 4.1.** *For any $0<\epsilon<1$ and $\ell\geq 0$, the splittable version of #Multilinear Monomial Detection problem on $2^{o(k)}s(C)^{\ell}$-skewed circuits can be computed exactly in expectation and approximated with factor $(1\pm\frac{1}{2})$ and success probability at least $1-\frac{\epsilon^2}{(1000\sqrt{k})^{\sqrt{k}}\ln(s(C)\frac{1}{\epsilon})}$ in time $2.61804^k\cdot 2^{o(k)}\cdot s(C)^{\ell+1}\mathtt{deg}_{\text{add}}(C)\cdot\log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}\cdot s(C))$.*

Before we prove Lemma 4.1, we first assert that together with Corollary 3.5, it implies Theorem 4.1.

*Proof of Theorem 4.1.* By Corollary 3.5, Lemma 4.1 implies that following statement. For any $0<\epsilon<1$ and $\ell\geq 0$, the #Multilinear Monomial Detection problem on $2^{o(k)}s(C)^{\ell}$-skewed circuits can be approximated with factor $(1\pm\epsilon)$ and success probability at least $\frac{9}{10}$ in time $2.61804^k\cdot 2^{o(k)}\cdot\frac{1}{\epsilon^2}\cdot s(C)^{\ell+1}\mathtt{deg}_{\text{add}}(C)\cdot\log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}\cdot s(C))$. First, notice that in case $\frac{1}{\epsilon}\geq(n+1)^k$, we can solve the problem exactly in time $\mathcal{O}(\frac{1}{\epsilon^2}\cdot s(C))$ since we can solve it in time $\mathcal{O}((n+1)^{2k}\cdot s(C))$ by simple dynamic programming that stores, for each node, what is the coefficient of each multilinear monomial of the polynomial corresponding to that node (because there are at most $(n+1)^k$ distinct multilinear monomials). Thus, we can assume that $\frac{1}{\epsilon}<(n+1)^k$ and attain the time complexity $2.61804^k\cdot 2^{o(k)}\cdot\frac{1}{\epsilon^2}\cdot s(C)^{\ell+1}\mathtt{deg}_{\text{add}}(C)\cdot\log^{\mathcal{O}(\sqrt{k})}s(C)$. Second, observe that if $s(C)\leq 2^{2^{o(\sqrt{k})}}$, then $\log^{\mathcal{O}(\sqrt{k})}s(C)=2^{o(k)}$, so the problem is solvable within time $2.61804^k\cdot 2^{o(k)}\cdot\frac{1}{\epsilon^2}\cdot s(C)^{\ell+1}\mathtt{deg}_{\text{add}}(C)$. Otherwise, when $s(C)>2^{2^{o(\sqrt{k})}}$, we have that $k<o(\log^2\log s(C))$ and thus $2.61804^k\cdot 2^{o(k)}\cdot\log^{\mathcal{O}(\sqrt{k})}s(C)=s(C)^{o(1)}$, so the problem is solvable in time $\frac{1}{\epsilon^2}\cdot s(C)^{\ell+1+o(1)}\mathtt{deg}_{\text{add}}(C)$. From this, and since $2.61804^k\cdot 2^{o(k)}=\mathcal{O}(2.619^k)$, we derive the theorem. □

We now turn to prove Lemma 4.1.

*Proof of Lemma 4.1.* It is well known that we can replace any arithmetic circuit $C$ with an equivalent circuit with fan-in two for all the internal nodes with linear blow up in the size. For example, replacing each node of in-degree greater than 2 with at most $s(C)$ many nodes of the same label and in-degree 2, we can convert a circuit $C$ to a circuit $C'$ of size $s(C')=\mathcal{O}(s(C))$ (specifically, $s_{\mathsf{V}}(C')=\mathcal{O}(s_{\mathsf{V}}(C)+s_{\mathsf{A}}(C))$ and $s_{\mathsf{A}}(C')=\mathcal{O}(s_{\mathsf{A}}(C))$). Moreover, $\mathtt{deg}_{\text{add}}(C')=\mathtt{deg}_{\text{add}}(C)$. So, from now onwards, we always assume that we are given a circuit of this form.

[4] Specifically, if $d=2^{\delta k}\cdot s(C)^{\mathcal{O}(1)}$, then the time complexity can be bounded by $c^k\cdot\frac{1}{\epsilon^2}\cdot s(C)^{\mathcal{O}(1)}$ for a constant $c$ that is clearly upper bounded but can also be can be substantially smaller than $2.619\cdot 2^{\delta}$ by optimizing choices of constants tailored to the $\delta$ at hand.

We first describe our algorithm. Afterwards, we assert its correctness and analyze its running time. The algorithm, denoted by $\mathsf{ALG}$, is based on dynamic programming. We treat the input set of variables as our universe (e.g., when computing representative families or parsimonious universal families). Consider an input $(C, k, \overline{\mathbf{U}}, f)$ (where $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_{\sqrt{k}})$ is a $\sqrt{k}$-partitioned universe and $f$ is a $(\sqrt{k}, k, 2)$-splitting function), and let $X$ denote the variable set corresponding to $C$.

We will use the following notation (we do *not* compute all the counters defined here, but only some of them in a table $N$ defined immediately; the rest will be approximately represented as will be argued in the proof of correctness later). For every node $v$ of $C$, $p \in \{1, 2, \ldots, k\}$, and $g$ such that $(f, g)$ is a $(\sqrt{k}, k, p, 2)$-splitting pair, let $\mathfrak{B}_{v,p,k} : \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} \to \mathbb{N}_0$ be the counter defined as follows. For every set $P \in \binom{X}{p}$, $\mathfrak{B}_{v,p,k}(P)$ is defined to be $\sum_{m \in \mathtt{MulLin}_{p,g}(P_{C_v}(X))} \mathtt{coeff}(P_{C_v}(X), m)$ where $\mathtt{MulLin}_{p,g}(P_{C_v}(X))$ is the set of multilinear monomials of $P_{C_v}(X)$ whose degree is $p$ and such that for every $i \in \{1, 2, \ldots, \sqrt{k}\}$, the number of variables that are part of the monomial and belong to $U_i$ is $g(i)$ (we remark that the second condition, regarding $g$, implies the first condition, regarding $p$).

We now start the description of our algorithm, $\mathsf{ALG}$. It first allocates a table $M$ that has an entry $M[v, p, g]$ for every node $v$ of $C$, $p \in \{1, 2, \ldots, k\}$, and $g$ such that $(f, g)$ is a $(\sqrt{k}, k, p, 2)$-splitting pair. Notice that, given $v$ and $p$, there are at most $(2k)^{\sqrt{k}}$ choices for $g$, and thus the table has at most $s(C)k(2k)^{\sqrt{k}}$ entries. The entry will store a counter $\mathfrak{C} : \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} \to \mathbb{N}_0$ (recall that we only explicitly store the assignment of values to the support.) We also use the algorithm in Proposition 4.1 to compute $W_v$ for each node $v$ of $C$. From this, we can clearly compute a table $N[v, p, g]$ indexed like $M$ and where $N[v, p, g]$ stores the counter that assigned to each

First, for all $p \in \{1, 2, \ldots, k\}$ and $g$ such that $(f, g)$ is a $(\sqrt{k}, k, p, 2)$-splitting pair, $\mathsf{ALG}$ invokes Corollary 3.3 to compute, with success probability at least $1 - \frac{1}{c}$, a family $\mathcal{F}_{p,g} \subseteq 2^X$ of $(\overline{\mathbf{U}}, 2, f, g, \frac{\ln \frac{3}{2}}{5k^2}, c, 1.447)$-universal family sampling, where $c = \frac{(1000\sqrt{k})^{\sqrt{k}} \ln(s(C)\frac{1}{\epsilon}}{\epsilon^2} \cdot s(C)(2k)^{2(\sqrt{k}+1)}$, that satisfies all of the following conditions.

1. $|\mathcal{F}_{p,g}| \leq \dfrac{(1.447k)^k}{p^p(1.447k - p)^{k-p}} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}}(\dfrac{1}{\epsilon}s(C))$.
2. $\mathcal{F}_{p,g}$ is a $\frac{\ln \frac{3}{2}}{5k^2}$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}, \mathcal{Q}^{\mathrm{CBAL}})$.
3. With respect to $\mathcal{F}_{p,g}$, $\mathsf{MEMBERSHIP}$ is a $T$-membership query procedure for
$$T = (\frac{1.447k}{1.447k - p})^{k-p} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}}(\frac{1}{\epsilon}s(C)).$$

Next, the counters at the entries of $M$ are computed by using a topological order with respect to the first argument $v$ (thus, when we compute an entry $M[v, p, g]$ where $v$ is not a leaf, the two entries $M[v_1, p_1, g_1]$ and $M[v_2, p_2, g_2]$ such that $v_1$ and $v_2$ are outgoing neighbors of $v$ have already been computed). The computation of an entry $M[v, p, g]$ is done as follows. (We could have included all skewed nodes at the basis, but prefer to write in this way so it will be easily extendible later to the case of general circuits where we do not have the table $N$.)

**Basis (Leaf).** If $p \neq 1$, then $M[v, p, g]$ stores the counter having empty support. Else, if $v \notin U_i$ where $i$ is the unique index in $\{1, 2, \ldots, \sqrt{k}\}$ such that $g(i) = 1$, then also $M[v, p, g]$ stores the counter having empty support. Otherwise, $M[v, p, g]$ stores the counter whose support contains only the set $\{x_i\}$ where $x_i$ is the label of $v$, and this set is assigned 1.

**Addition Node.** Let $v_1$ and $v_2$ be the two outgoing neighbors of $v$. Then, the counter $\mathfrak{C} : \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} :\to \mathbb{N}_0$ stored at the entry $M[v, p, g]$ is defined as follows. For every set $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$, define

$\mathfrak{C}(P) = \mathfrak{C}_1(P) + \mathfrak{C}_2(P)$ where $\mathfrak{C}_1$ and $\mathfrak{C}_2$ are the counters stored at $M[v_1, p, g]$ and $M[v_2, p, g]$, respectively.

**Multiplication Node.** Let $v_1$ and $v_2$ be the two outgoing neighbors of $v$. Let $\mathcal{I}$ be the set consisting of all quadruples $(p_1, p_2, g_1, g_2)$ such that $p_1, p_2 \in \mathbb{N}, p_1 + p_2 = p, (f, g_1)$ and $(f, g_2)$ are $(\sqrt{k}, k, p_1, 2)$ and $(\sqrt{k}, k, p_2, 2)$-splitting function pairs, respectively, and for every $i \in \{1, 2, \dots, \sqrt{k}\}$, it holds that $g_1(i)+g_2(i) = g(i)$. Then, for every quadruple $(p_1, p_2, g_1, g_2) \in \mathcal{I}$, let $\mathfrak{C}_{(p_1,p_2,g_1,g_2)} : \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} \to \mathbb{N}_0$ be the counter defined as follows. For every set $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$, define $\mathfrak{C}_{(p_1,p_2,g_1,g_2)}(P) = \sum_{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1}, P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : P_1 \cap P_2 = \emptyset} \mathfrak{C}_{v_1,p_1,g_1}(P) \cdot \mathfrak{C}_{v_2,p_2,g_2}(P)$ where $\mathfrak{C}_{v_1,p_1,g_1}$ and $\mathfrak{C}_{v_2,p_2,g_2}$ are the counters stored at $M[v_1, p_1, g_1]$ and $M[v_2, p_2, g_2]$, respectively, where if $N[v_1, p_1, g_1]$ is not null, then use it instead to define $\mathfrak{C}_{v_1,p_1,g_1}$, and otherwise ($N[v_2, p_2, g_2]$ must not be null), use $N[v_2, p_2, g_2]$ instead to define $\mathfrak{C}_{v_2,p_2,g_2}$. Then, the counter $\mathfrak{C} : \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} :\to \mathbb{N}_0$ is defined as follows. For every set $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$, define $\mathfrak{C}(P) = \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \mathfrak{C}_{(p_1,p_2,g_1,g_2)}(P)$. The entry $M[v, p, g]$ stores the counter $\widehat{\mathfrak{C}} : \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} \to \mathbb{N}_0$ obtained by applying Theorem 3.1 with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$, $\mathcal{F}_{p,g}$ and the aforementioned counter $\mathfrak{C}$, where $n$ is clear (being $|X|$), $p$ is clear, $q = k-p$, $\epsilon'$ (denoted by $\epsilon$ in the statement but here denoted by $\epsilon'$ to avoid overloading notation) is $\frac{\ln \frac{3}{2}}{k^2}$, and $c = \frac{(1000\sqrt{k})^{\sqrt{k}}}{\epsilon^2} \ln(s(C)\frac{1}{\epsilon}) \cdot s(C)(2k)^{2(\sqrt{k}+1)}$ (the same as defined earlier in this proof).

**Output.** The final answer returned by $\mathsf{ALG}$ is $\sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,f}} \mathfrak{C}(P)$ where $\mathfrak{C}$ is the counter stored at $M[\mathrm{root}(C), k, f]$. This completes the description of the algorithm.

**Time Complexity.** First, notice that the time complexity of the computation of each family $\mathcal{F}_{p,g}$ is upper bounded by $\mathcal{O}(|\mathcal{F}_{p,g}|s(C)) = \dfrac{(1.447k)^k}{p^p(1.447k-p)^{k-p}} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot s(C) \cdot \log^{\sqrt{k}}(\dfrac{1}{\epsilon}s(C))$. Notice that for any non-negative integer $p \leq k$, $\dfrac{(1.447k)^k}{p^p(1.447k-p)^{k-p}} = \mathcal{O}(2.61804^k)$. (Later, when considering the computation corresponding to a multiplication node, we will even bound a larger expression by $\mathcal{O}(2.61804^k)$.) Moreover, there are at most $2^{o(k)}$ pairs $(p, g)$ for which we compute a family $\mathcal{F}_{p,g}$. Thus, the time taken to compute all families $\mathcal{F}_{p,g}$ is altogether bounded by $2.61804^k \cdot s(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon} \cdot s(C))$.

We are left with the time complexity of the computation of $M$ and $N$. To this end, notice that the number of entries of $M$ is upper bounded by $2^{o(k)}s(C)$. This are also the number of entries of $N$, and thus, by Propostion 4.1, we can already conclude that $N$ can be computed within the desired time. Thus, to conclude the time complexity in the lemma, it remains to show that each entry of $M$ can be computed in time $2.61804^k \cdot 2^{o(k)} \cdot s(C)^\ell \mathtt{deg}_{\mathrm{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon} \cdot s(C))$. For this purpose, consider some entry $M[v, p, g]$. If $v$ is a leaf node, then the time complexity is constant. Else, $v$ has two outgoing neighbors $v_1$ and $v_2$. We have two cases.

- In case $v$ is an addition node, $|\mathsf{supp}(\mathfrak{C})| \leq |\mathsf{supp}(\mathfrak{C}_1)| + |\mathsf{supp}(\mathfrak{C}_2)|$. Moreover, $\mathfrak{C}$ can clearly be computed in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C}_1)| + |\mathsf{supp}(\mathfrak{C}_2)|)$.
- In case $v$ is a multiplication node, $|\mathsf{supp}(\mathfrak{C})| \leq \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} |\mathsf{supp}(\mathfrak{C}_{v_1,p_1,g_1})| \cdot |\mathsf{supp}(\mathfrak{C}_{v_2,p_2,g_2})|$. Moreover, $\mathfrak{C}$ is clearly computable in time $\mathcal{O}(\sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} |\mathsf{supp}(\mathfrak{C}_{v_1,p_1,g_1})| \cdot |\mathsf{supp}(\mathfrak{C}_{v_2,p_2,g_2})|)$.

By Theorem 3.1 (because each counter stored at an entry corresponding to a multiplication node above has been obtained by the application of this theorem), for any counter $\mathfrak{C}_{\text{mul}}$ stored at some entry $M[v', p', g']$ where $v'$ is a multiplication node, we have that

$$\begin{aligned}|\mathtt{supp}(\mathfrak{C}_{\text{mul}})| \quad &\leq \quad \mathcal{O}(k^2 \cdot |\mathcal{F}_{p',g'}| \cdot \log(\tfrac{(1000\sqrt{k})^{\sqrt{k}}\ln(s(C)\frac{1}{\epsilon})}{\epsilon^2} s(C)(2k)^{2(\sqrt{k}+1)}) \\ &\qquad \cdot (\log(\tfrac{(1000\sqrt{k})^{\sqrt{k}}\ln(s(C)\frac{1}{\epsilon})}{\epsilon^2} s(C)(2k)^{2(\sqrt{k}+1)}) + \log|\mathcal{F}_{p',g'}|)) \\ &\leq \quad \frac{(1.447k)^k}{p'^{p'}(1.447k - p')^{k-p'}} \cdot 2^{o(k)} \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C)).\end{aligned}$$

Second, notice that for any counter $\mathfrak{C}_{\text{add}}$ stored at some entry $M[v', p', g']$ where $v'$ is a addition node, we have that the support of $\mathfrak{C}_{\text{add}}$ is the union of the support of counters stored at entries $M[v'', p'', g'']$ where $v''$ is a multiplication node reachable from $v'$ while internally traversing only addition nodes. Therefore,

$$|\mathtt{supp}(\mathfrak{C}_{\text{add}})| \leq \frac{(1.447k)^k}{p'^{p'}(1.447k - p')^{k-p'}} \cdot 2^{o(k)} \cdot \mathtt{deg}_{\text{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C)).$$

Thus, in case $v$ is an addition node, we can already conclude that the computation time of the entry $M[v, p, g]$ is upper bounded by $\frac{(1.447k)^k}{p^p(1.447k - p)^{k-p}} \cdot 2^{o(k)} \cdot \mathtt{deg}_{\text{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C))$, where $\frac{(1.447k)^k}{p^p(1.447k - p)^{k-p}} = \mathcal{O}(2.61804^k)$ (similarly to our consideration of the computation of parsimonious families).

We thus now focus only on the case where $v$ is a multiplication node. Recall that both $|\mathtt{supp}(\mathfrak{C})|$ and the time to compute $\mathfrak{C}$ are upper bounded by $\mathcal{O}(\sum_{(p_1,p_2,g_1,g_2)\in\mathcal{I}} |\mathtt{supp}(\mathfrak{C}_{v_1,p_1,g_1})| \cdot |\mathtt{supp}(\mathfrak{C}_{v_2,p_2,g_2})|)$. Therefore, because $|\mathcal{I}| = 2^{o(k)}$, we further obtain an upper bound of $2^{o(k)} \cdot \max_{(p_1,p_2,g_1,g_2)\in\mathcal{I}} |\mathtt{supp}(\mathfrak{C}_{v_1,p_1,g_1})| \cdot |\mathtt{supp}(\mathfrak{C}_{v_2,p_2,g_2})|$. From the above discussion, we know that for any $(p_1, p_2, g_1, g_2) \in \mathcal{I}$ and $i \in \{1, 2\}$ such that $\mathfrak{C}_{v_i,p_i,g_i} = M[v_i, p_i, g_i]$, we have that

$$|\mathtt{supp}(\mathfrak{C}_{v_i,p_i,g_i})| \leq \frac{(1.447k)^k}{p_i^{p_i}(1.447k - p_i)^{k-p_i}} \cdot 2^{o(k)} \cdot \mathtt{deg}_{\text{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C)).$$

Thus, because the input arithmetic circuit $C$ is $2^{o(k)}s(C)^{\ell}$-skewed (we remark in the proof for general circuits we will not be able to rely on this, but just use the analogous equation to the one above for both $i = 1$ and $i = 2$), we further derive that both $|\mathtt{supp}(\mathfrak{C})|$ and the time to compute $\mathfrak{C}$ are bounded from above by

$$\max_{1\leq\widehat{p}\leq k} \frac{(1.447k)^k}{\widehat{p}^{\widehat{p}}(1.447k - \widehat{p})^{k-\widehat{p}}} \cdot 2^{o(k)} \cdot s(C)^{\ell}\mathtt{deg}_{\text{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C)).$$

By Theorem 3.1, the time required to compute $\widehat{\mathfrak{C}}$ is bounded by $\mathcal{O}(|\mathtt{supp}(\mathfrak{C})| \cdot T)$ where

$$T = (\frac{1.447k}{1.447k - p})^{k-p} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}}(\frac{1}{\epsilon}s(C)).$$

Combining this with the aforementioned upper bound on $|\mathtt{supp}(\mathfrak{C})|$, we derive that, overall, the time complexity is bounded by

$$\max_{1\leq\widehat{p}\leq p} \frac{(1.447k)^k}{\widehat{p}^{\widehat{p}}(1.447k - \widehat{p})^{k-\widehat{p}}}(\frac{1.447k}{1.447k - p})^{k-p} \cdot 2^{o(k)} \cdot s(C)^{\ell}\mathtt{deg}_{\text{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C)).$$

Because any value $p \in \{1, \ldots, k\}$ is to be taken into consideration, we derive that the time required to compute a single entry of $M$ is bounded by

$$\max_{1\leq p\leq k, 1\leq\widehat{p}\leq p} \frac{(1.447k)^k}{\widehat{p}^{\widehat{p}}(1.447k - \widehat{p})^{k-\widehat{p}}}(\frac{1.447k}{1.447k - p})^{k-p} \cdot 2^{o(k)} \cdot s(C)^{\ell}\mathtt{deg}_{\text{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C)).$$

First, notice that the maximum is achieved when $\widehat{p} = p$—indeed, the larger $p$ is, the smaller $(\frac{1.447k}{1.447k-p})^{k-p}$ is, thus one is to choose a larger $p$ only in order to choose the largest possible $\widehat{p}$. Therefore, we further derive an upper bound of

$$\max_{1\leq p\leq k} \frac{(1.447k)^{2k-p}}{p^p(1.447k-p)^{2k-2p}} \cdot 2^{o(k)} \cdot s(C)^{\ell}\mathtt{deg}_{\mathrm{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon}s(C)).$$

The maximum is achieved when $p = \alpha k$ for $\alpha \approx 0.55277$ (the exact same computation of maximum of this expression is done in both [FLPS16, SZ16] where the reader can find further details). Then, the expression above is upper bounded by $2.61804^k \cdot 2^{o(k)} \cdot \mathtt{deg}_{\mathrm{add}}(C) \cdot \log^{\mathcal{O}(\sqrt{k})}(\frac{1}{\epsilon} \cdot s(C))$. This completes the analysis of the time complexity of $\mathsf{ALG}$.

**Correctness.** Notice that $\mathsf{ALG}$ performs at most $k \cdot (2k)^{\sqrt{k}}$ calls to the algorithm in Corollary 3.3 (one for every choice of $p$ and $g$), and at most $s(C) \cdot k \cdot (2k)^{\sqrt{k}}$ calls to the algorithm in Theorem 3.1 (at most one for every choice of $v, p$ and $g$). As the failure probability for each one of them is upper bounded by $\frac{1}{c} = 1/\left(\frac{(1000\sqrt{k})^{\sqrt{k}}\ln(s(C)\frac{1}{\epsilon})}{\epsilon^2} \cdot s(C)(2k)^{2(\sqrt{k}+1)}\right)$, we have by union bound that the probability that at least one call will fail is upper bounded by $\frac{\epsilon^2}{(1000\sqrt{k})^{\sqrt{k}}\ln(s(C)\frac{1}{\epsilon})}$. Thus, with probability at least $1 - \frac{\epsilon^2}{(1000\sqrt{k})^{\sqrt{k}}\ln(s(C)\frac{1}{\epsilon})}$, all calls to the algorithms in Corollary 3.3 and Theorem 3.1 were successful. Thus, to prove the correctness of $\mathsf{ALG}$, we prove the following claim. (We remark that one can use $p$ instead of $p^2$ and still the proof will go through by relying on the skeweness of the circuit, but we refrain from doing that so that the proof for general circuits later will be similar.)

**Claim 4.1.** *For every node $v$ of $C$, $p \in \{1, 2, \ldots, k\}$, and $g$ such that $(f, g)$ is a $(\sqrt{k}, k, p, 2)$-splitting pair, the following holds: The counter $\widehat{\mathfrak{C}}$ stored at $M[v, p, g]$ equals $\mathfrak{B}_{v,p,g}$ in expectation (i.e., for each $P \in \mathcal{P}^{\mathrm{BAL}}_{\mathbf{U},f,g}$, the expected value assigned by $\widehat{\mathfrak{C}}$ equals $\mathfrak{B}_{v,p,g}(P)$), and, under the assumption that all calls to the algorithms in Corollary 3.3 and Theorem 3.1 are successful, it $((1 - \frac{\ln\frac{3}{2}}{k^2})^{p^2}, (1 + \frac{\ln\frac{3}{2}}{k^2})^{p^2}, k - p)$-represents $\mathfrak{B}_{v,p,g}$ with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\mathbf{U},f,g}$.*

*Proof.* For arguments that do not concern expectation, we will implicitly assume that all calls to the algorithms in Corollary 3.3 and Theorem 3.1 are successful. The proof is done by induction on the order of the computation of the entries of $M$ (specifically topological order with respect to the first argument). In the basis, where $v$ is a leaf, the counter $\widehat{\mathfrak{C}}$ stored at $M[v, p, g]$ clearly equals $\mathfrak{B}_{v,p,g}$, thus the claim trivially holds. Now, consider some entry $M[v, p, g]$ where $v$ is not a leaf, and suppose that the claim holds for all entries that are computed before $M[v, p, g]$. We need to consider two cases depending on whether $v$ is an addition node or a multiplication node.

**Case I: Addition Node.** Note that $P_{C_v}(X) = P_{C_{v_1}}(X) + P_{C_{v_2}}(X)$. Therefore, for any $P \in \mathcal{P}^{\mathrm{BAL}}_{\mathbf{U},f,g}$, we have that $\mathfrak{B}_{v,p,g}(P) = \mathfrak{B}_{v_1,p,g}(P) + \mathfrak{B}_{v_2,p,g}(P)$. By the inductive hypothesis, $\mathfrak{C}_1$ and $\mathfrak{C}_2$ represent in expectation and $((1 - \frac{\ln\frac{3}{2}}{k^2})^{p^2}, (1 + \frac{\ln\frac{3}{2}}{k^2})^{p^2}, k - p)$-represent $\mathfrak{B}_{v_1,p,g}$ and $\mathfrak{B}_{v_2,p,g}$, respectively, with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\mathbf{U},f,g}$. Thus, for every $P \in \mathcal{P}^{\mathrm{BAL}}_{\mathbf{U},f,g}$, we have that $E[\mathfrak{C}_2(P)] = \mathfrak{B}_{v_1,p,g}(P)$ and $E[\mathfrak{C}_2(P)] = \mathfrak{B}_{v_2,p,g}(P)$; therefore, by linearity of expectation, $E[\mathfrak{C}(P)] = E[\mathfrak{C}_1(P)] + E[\mathfrak{C}_2(P)] = \mathfrak{B}_{v_1,p,g}(P) + \mathfrak{B}_{v_2,p,g}(P)$, which means that $\mathfrak{C}$ represent in expectation $\mathfrak{B}_{v,p,g}$.

Moreover, by the aforementioned inductive assumption regarding approximate representation, for every $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\mathbf{U},f,g}$ and $i \in \{1, 2\}$, we have that

$$(1-\frac{\ln\frac{3}{2}}{k^2})^{p^2} \cdot \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\mathbf{U},f,g} : P \cap Q = \emptyset} \mathfrak{B}_{v_i,p,g}(P) \leq \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\mathbf{U},f,g} : P \cap Q = \emptyset} \mathfrak{C}_i(P) \leq (1+\frac{\ln\frac{3}{2}}{k^2})^{p^2} \cdot \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\mathbf{U},f,g} : P \cap Q = \emptyset} \mathfrak{B}_{v_i,p,g}(P).$$

Thus, for every $Q \in \mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,g}$, we have that

$$\begin{aligned}
&\sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{C}(P) \\
&= \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} (\mathfrak{C}_1(P) + \mathfrak{C}_2(P)) \\
&= \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{C}_1(P) + \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{C}_2(P) \\
&\leq (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2} \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{B}_{v_1,p,g}(P) + (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2} \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{B}_{v_2,p,g}(P) \\
&= (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2} \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} (\mathfrak{B}_{v_1,p,g}(P) + \mathfrak{B}_{v_2,p,g}(P)) \\
&= (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2} \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{B}_{v,p,g}(P).
\end{aligned}$$

Symmetrically, for every $Q \in \mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,g}$, we conclude that

$$\sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{C}(P) \geq (1 - \frac{\ln \frac{3}{2}}{k^2})^{p^2} \sum_{P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{B}_{v,p,g}(P).$$

Thus, $\mathfrak{C}$ $((1 - \frac{\ln \frac{3}{2}}{k^2})^{p^2}, (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2}, k - p)$-represents $\mathfrak{B}_{v,p,g}$ with respect to $\mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,g}$.

**Case II: Multiplication Node.** Note that $P_{C_v}(X) = P_{C_{v_1}}(X) \cdot P_{C_{v_2}}(X)$. Therefore, for any $P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g}$, we have that

$$\mathfrak{B}_{v,p,g}(P) = \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{\substack{P_1 \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g_1}, P_2 \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g_2} : \\ P_1 \cap P_2 = \emptyset, P_1 \cup P_2 = P}} \mathfrak{B}_{v_1,p_1,g_1}(P_1) \cdot \mathfrak{B}_{v_2,p_2,g_2}(P_2).$$

Further, by the computation at a multiplication node, for every $P \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g_i}$,

$$\mathfrak{C}(P) = \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{\substack{P_1 \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g_1}, P_2 \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g_2} : \\ P_1 \cap P_2 = \emptyset, P_1 \cup P_2 = P}} \mathfrak{C}_{v_1,p_1,g_1}(P_1) \cdot \mathfrak{C}_{v_2,p_2,g_2}(P_2)$$

We first consider representation in expectation. By the inductive hypothesis, for every $(p_1, p_2, g_1, g_2) \in \mathcal{I}$ and $i \in \{1, 2\}$, $\mathfrak{C}_{v_i,p_i,g_i}$ represents in expectation $\mathfrak{B}_{v_i,p_i,g_i}$, and therefore for every $P_i \in \mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g_i}$, we have that $E[\mathfrak{C}_{v_i,p_i,g_i}(P_i)] = \mathfrak{B}_{v_i,p_i,g_i}(P_i)$. Unfortunately, this does not suffice to obtain representation in expectation, and here we have to rely on the skewness of the circuit. This is also the reason why for general circuits, we do not claim representatiob in expectation, which therefore requires us to make certain modification that overall lead to another factor of $\frac{1}{\epsilon^2}$ in the running. Towards the correct proof that relies on skeweness, we will also briefly explain why otherwise the proof does not work. By the two aforementioned equalities and linearity of expectation, we get that

$$\begin{aligned}E[\mathfrak{C}(P)] &= E[\sum_{(p_1,p_2,g_1,g_2)\in\mathcal{I}} \sum_{\substack{P_1\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1},P_2\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2}:\\ P_1\cap P_2=\emptyset,P_1\cup P_2=P}} \mathfrak{C}_{v_1,p_1,g_1}(P_1)\cdot\mathfrak{C}_{v_2,p_2,g_2}(P_2)]\\ &= \sum_{(p_1,p_2,g_1,g_2)\in\mathcal{I}} \sum_{\substack{P_1\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1},P_2\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2}:\\ P_1\cap P_2=\emptyset,P_1\cup P_2=P}} E[\mathfrak{C}_{v_1,p_1,g_1}(P_1)\cdot\mathfrak{C}_{v_2,p_2,g_2}(P_2)]\end{aligned}$$

Then, we would have like to say that the above is equal to

$$\begin{aligned}&= \sum_{(p_1,p_2,g_1,g_2)\in\mathcal{I}} \sum_{\substack{P_1\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1},P_2\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2}:\\ P_1\cap P_2=\emptyset,P_1\cup P_2=P}} E[\mathfrak{C}_{v_1,p_1,g_1}(P_1)]\cdot E[\mathfrak{C}_{v_2,p_2,g_2}(P_2)]\\ &= \sum_{(p_1,p_2,g_1,g_2)\in\mathcal{I}} \sum_{\substack{P_1\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1},P_2\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2}:\\ P_1\cap P_2=\emptyset,P_1\cup P_2=P}} \mathfrak{B}_{v_1,p_1,g_1}(P_1)\cdot\mathfrak{B}_{v_2,p_2,g_2}(P_2) = \mathfrak{B}_{v,p,g}(P).\end{aligned}$$

Unfortunately, if both $\mathfrak{C}_{v_1,p_1,g_1}(P_1)$ and $\mathfrak{C}_{v_2,p_2,g_2}(P_2)$ were to be taken from $M$, then they might not be independent and thus the above may not be true. However, by relying on skewness, we computed one of them, say, $\mathfrak{C}_{v_1,p_1,g_1}(P_1)$, by taking the entry $N[v_1,p_1,g_1]$, which makes $\mathfrak{C}_{v_1,p_1,g_1}(P_1)$ deterministic, being necessarily equal to $\mathfrak{B}_{v_1,p_1,g_1}$. Thus, the above transitions hold, and $\mathfrak{C}$ indeed represents in expectation $\mathfrak{B}_{v,p,g}$.

We now consider approximate representation. By relying on skeweness, the proof can be slightly simplified, but as said earlier, we refrain from doing this so that this proof can also be used for general circuits. In case $p=1$, $\mathcal{I}$ is empty (since $p_1$ and $p_2$ should each be at least 1 and together sum to $p$), and then for any $P\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$, we have that $\mathfrak{C}(P)=\mathfrak{B}_{v,p,g}(P)$. Thus, in this case, clearly $\mathfrak{C}$ $((1-\frac{\ln\frac{3}{2}}{k^2})^{p^2-1},(1+\frac{\ln\frac{3}{2}}{k^2})^{p^2-1},k-p)$-represents $\mathfrak{B}_{v,p,g}$ with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$.

Next, suppose that $p\geq 2$. Arbitrarily choose some $Q\in\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$. By the inductive hypothesis, for every $(p_1,p_2,g_1,g_2)\in\mathcal{I}$ and $i\in\{1,2\}$, $\mathfrak{C}_{v_i,p_i,g_i}$ $((1-\frac{\ln\frac{3}{2}}{k^2})^{p_i^2},(1+\frac{\ln\frac{3}{2}}{k^2})^{p_i^2},k-p_i)$-represents $\mathfrak{B}_{v_i,p_i,g_i}$ with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g_i}$. Thus, for every $i\in\{1,2\}$ and $\widetilde{Q}\in\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g_i}$, we have that

$$(1-\frac{\ln\frac{3}{2}}{k^2})^{p_i^2}\cdot\sum_{P_i\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_i}:P_i\cap\widetilde{Q}=\emptyset}\mathfrak{B}_{v_i,p_i,g_i}(P_i)\leq\sum_{P_i\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_i}:P_i\cap\widetilde{Q}=\emptyset}\mathfrak{C}_i(P_i)\leq(1+\frac{\ln\frac{3}{2}}{k^2})^{p_i^2}\cdot\sum_{P_i\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_i}:P_i\cap\widetilde{Q}=\emptyset}\mathfrak{B}_{v_i,p_i,g_i}(P_i).$$

We now proceed to combine the above expressions to show that $\sum_{P\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{C}(P)\leq(1+\frac{\ln\frac{3}{2}}{k^2})^{p^2-1}\sum_{P\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{B}_{v,p,g}(P)$. The proof that also $\sum_{P\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{C}(P)\geq(1-\frac{\ln\frac{3}{2}}{k^2})^{p^2-1}\sum_{P\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{B}_{v,p,g}(P)$ is symmetric. To this end, observe that for every $(p_1,p_2,g_1,g_2)\in\mathcal{I}$ and $P_1\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1}$ such that $P_1\cap Q=\emptyset$, it holds that $P_1\cup Q\in\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g_2}$; similarly, for every $(p_1,p_2,g_1,g_2)\in\mathcal{I}$ and $P_2\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2}$ such that $P_2\cap Q=\emptyset$, it holds that $P_2\cup Q\in\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g_1}$. Thus, we overall have that

$$
\begin{aligned}
&\sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{C}(P) \\
&= \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{\substack{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1}, P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : \\ P_1 \cap P_2 = \emptyset, P_1 \cup P_2 = P}} \mathfrak{C}_{v_1,p_1,g_1}(P_1) \cdot \mathfrak{C}_{v_2,p_2,g_2}(P_2) \\
&= \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1} : P_1 \cap Q = \emptyset} \mathfrak{C}_{v_1,p_1,g_1}(P_1) \cdot \left( \sum_{\substack{P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : \\ P_2 \cap (P_1 \cup Q) = \emptyset}} \mathfrak{C}_{v_2,p_2,g_2}(P_2) \right) \\
&\leq \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1} : P_1 \cap Q = \emptyset} \mathfrak{C}_{v_1,p_1,g_1}(P_1) \cdot \left( (1 + \frac{\ln \frac{3}{2}}{k^2})^{p_2^2} \sum_{\substack{P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : \\ P_2 \cap (P_1 \cup Q) = \emptyset}} \mathfrak{B}_{v_2,p_2,g_2}(P_2) \right) \\
&= (1 + \frac{\ln \frac{3}{2}}{k^2})^{p_2^2} \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : P_2 \cap Q = \emptyset} \mathfrak{B}_{v_2,p_2,g_2}(P_2) \cdot \left( \sum_{\substack{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1} : \\ P_1 \cap (P_2 \cup Q) = \emptyset}} \mathfrak{C}_{v_1,p_1,g_1}(P_1) \right) \\
&\leq (1 + \frac{\ln \frac{3}{2}}{k^2})^{p_2^2} \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : P_2 \cap Q = \emptyset} \mathfrak{B}_{v_2,p_2,g_2}(P_2) \cdot \left( (1 + \frac{\ln \frac{3}{2}}{k^2})^{p_1^2} \sum_{\substack{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1} : \\ P_1 \cap (P_2 \cup Q) = \emptyset}} \mathfrak{B}_{v_1,p_1,g_1}(P_1) \right) \\
&= (1 + \frac{\ln \frac{3}{2}}{k^2})^{p_1^2 + p_2^2} \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{\substack{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1}, P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : \\ P_1 \cap P_2 = \emptyset, P_1 \cup P_2 = P}} \mathfrak{B}_{v_1,p_1,g_1}(P_1) \cdot \mathfrak{B}_{v_2,p_2,g_2}(P_2) \\
&\leq (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2 - 1} \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \sum_{(p_1,p_2,g_1,g_2) \in \mathcal{I}} \sum_{\substack{P_1 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_1}, P_2 \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g_2} : \\ P_1 \cap P_2 = \emptyset, P_1 \cup P_2 = P}} \mathfrak{B}_{v_1,p_1,g_1}(P_1) \cdot \mathfrak{B}_{v_2,p_2,g_2}(P_2) \\
&\leq (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2 - 1} \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{B}_{v,p,g}(P)
\end{aligned}
$$

Here, the last inequality followed because $p_1^2 + p_2^2 \leq (p-1)^2 + 1^2 = p^2 - 2p + 2 \leq p^2 - 1$, where the last inequality is satisfied because $p \geq 2$ in the current case.

Thus, because the choice of $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$ was arbitrary, also when $p \geq 2$, we have that $\mathfrak{C}$ $((1 - \frac{\ln \frac{3}{2}}{k^2})^{p^2-1}, (1 + \frac{\ln \frac{3}{2}}{k^2})^{p^2-1}, k - p)$-represents $\mathfrak{B}_{v,p,g}$ with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$.

Note that $\widehat{\mathfrak{C}}$ represents in expectation and $(\frac{\ln \frac{3}{2}}{k^2}, k - p)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$. First, for every $P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$, we thus have that both $E[\widehat{\mathfrak{C}}(P)] = \mathfrak{C}(P)$ and $E[\mathfrak{C}(P)] = \mathfrak{B}_{v,p,g}(P)$, therefore $E[\widehat{\mathfrak{C}}(P)] = \mathfrak{B}_{v,p,g}(P)$. Thus, $\widehat{\mathfrak{C}}$ represents in expectation $\mathfrak{B}_{v,p,g}$. Second, for every $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$, we have that

$$
(1 - \frac{\ln \frac{3}{2}}{k^2}) \cdot \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{C}(P) \leq \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P) \leq (1 + \frac{\ln \frac{3}{2}}{k^2}) \cdot \sum_{P \in \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} : P \cap Q = \emptyset} \mathfrak{C}(P).
$$

Recall that, because $\mathfrak{C}$ $((1-\frac{\ln\frac{3}{2}}{k^2})^{p^2-1},(1+\frac{\ln\frac{3}{2}}{k^2})^{p^2-1},k-p)$-represents $\mathfrak{B}_{v,p,g}$ with respect to $\mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,g}$, we also have that

$$(1-\frac{\ln\frac{3}{2}}{k^2})^{p^2-1}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{C}(P)\leq\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\widehat{\mathfrak{C}}(P)\leq(1+\frac{\ln\frac{3}{2}}{k^2})^{p^2-1}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{C}(P).$$

Combining both expression, we derive that

$$(1-\frac{\ln\frac{3}{2}}{k^2})^{p^2}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{B}_{v,p,g}(P)\leq\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\widehat{\mathfrak{C}}(P)\leq(1+\frac{\ln\frac{3}{2}}{k^2})^{p^2}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,g}:P\cap Q=\emptyset}\mathfrak{B}_{v,p,g}(P).$$

Since the choice of $Q\in\mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,g}$ was arbitrary, we conclude that $\mathfrak{C}$ $((1-\frac{\ln\frac{3}{2}}{k^2})^{p^2},(1+\frac{\ln\frac{3}{2}}{k^2})^{p^2},k-p)$-represents $\mathfrak{B}_{v,p,g}$ with respect to $\mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,g}$. □

Having proved the claim, we turn to complete the proof of the theorem. To this end, notice that the exact solution to splittable version, being $\sum_{m\in\widetilde{\texttt{MulLin}}_k(P_C(X))}\texttt{coeff}(P_C(X),m)$, equals $\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{B}_{r,k,f}(P)$ where $r=\text{root}(C)$. Recall that the final answer returned by $\mathsf{ALG}$ is $\sum\limits_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{C}(P)$ where $\mathfrak{C}$ is the counter stored at $M[r,k,f]$. By Claim 4.1, for every $P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}$, the expected value of $\mathfrak{C}(P)$ equals $\mathfrak{B}_{r,k,f}(P)$, and therefore the expected value of the final answer returned by $\mathsf{ALG}$ is $\sum_{m\in\texttt{MulLin}_k(P_C(X))}\texttt{coeff}(P_C(X),m)$. Moreover, by By Claim 4.1, $\mathfrak{C}$ $((1-\frac{\ln\frac{3}{2}}{k^2})^{k^2},(1+\frac{\ln\frac{3}{2}}{k^2})^{k^2},0)$-represents $\mathfrak{B}_{r,k,f}$ with respect to $\mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,f}$. Thus, for every $Q\in\mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,f}$,

$$(1-\frac{\ln\frac{3}{2}}{k^2})^{k^2}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}:P\cap Q=\emptyset}\mathfrak{B}_{r,k,f}(P)\leq\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}:P\cap Q=\emptyset}\mathfrak{C}(P)\leq(1+\frac{\ln\frac{3}{2}}{k^2})^{k^2}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}:P\cap Q=\emptyset}\mathfrak{B}_{r,k,f}(P).$$

Observe that when $f=g$, the family $\mathcal{Q}^{\text{CBAL}}_{\overline{\mathbf{U}},f,g}$ consists only of the empty set. Thus, the expression above simplifies to

$$(1-\frac{\ln\frac{3}{2}}{k^2})^{k^2}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{B}_{r,k,f}(P)\leq\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{C}(P)\leq(1+\frac{\ln\frac{3}{2}}{k^2})^{k^2}\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{B}_{r,k,f}(P).$$

Lastly, note that $(1-\frac{\ln\frac{3}{2}}{k^2})^{k^2}\geq(1-\ln\frac{3}{2})>(1-\frac{1}{2})$, as well as that $(1+\frac{\ln\frac{3}{2}}{k^2})^{k^2}\leq e^{\ln\frac{3}{2}}=(1+\frac{1}{2})$. Therefore, the last expression yields that

$$(1-\frac{1}{2})\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{B}_{r,k,f}(P)\leq\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{C}(P)\leq(1+\frac{1}{2})\cdot\sum_{P\in\mathcal{P}^{\text{BAL}}_{\overline{\mathbf{U}},f,f}}\mathfrak{B}_{r,k,f}(P).$$

This completes the proof. □

For all our problem-specific applications, the natural circuit constructions in the reductions produce circuits that are not only skewed, but also have the following property: Every addition node has out-degree at most 1 and all of its out-going neighbors are multiplication nodes. We refer to such circuits as *additively simple*. It is straightforward to see that for such circuits, we do not obtain the term $\texttt{deg}_{\text{add}}(C)$ in the time complexity in Theorem 4.1. Indeed, in the computation of the time complexity in Lemma 4.1, when we consider a multiplication node, the size of the support of the counters stored in its addition child are dependent only on their outgoing-degrees (because every addition node has all of its out-going neighbors are multiplication nodes), and the out-degree of each addition node will only be accounted for once (because every addition node has in-degree 1). Thus, we also state the following theorem.

**Theorem 4.2.** *For any $0 < \epsilon < 1$ and $\ell \geq 0$, the* #MULTILINEAR MONOMIAL DETECTION *problem on $2^{o(k)} s(C)^{\ell}$-skewed additively simple circuits can be approximated with factor $(1 \pm \epsilon)$ and success probability at least $\frac{9}{10}$ in time $\mathcal{O}((2.619^k + s(C)^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot s(C)^{1+\ell})$. In particular, when $\ell = 0$, the time is $\mathcal{O}((2.619^k + s(C)^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot s(C))$.*

## 4.2 Reductions to #MULTILINEAR MONOMIAL DETECTION on Skewed Circuits

A wide variety of problems are well-known to be (in fact, even often very easily) reducible to the MULTILINEAR MONOMIAL DETECTION problem on $2^{o(k)} s(C)^{\ell}$-skewed additively simple circuits where the size of the output circuit is linear in $k^{\mathcal{O}(1)}$ multiplied by the size of the input instance (see, e.g., [KW16b, KW16a]), and it can be easily verified that the reductions are parsimonious. This includes, for example, $k$-PATH, $k$-TREE (or, more generally, SUBGRAPH ISOMORPHISM where the treewidth of input graph is bounded by a fixed constant), $r$-SET $k$-PACKING, $r$-DIMENSIONAL $k$-MATCHING, GRAPH MOTIF, and more. For all of the aforementioned problems except $k$-TREE, the produced circuit is, in fact, $\mathcal{O}(1)$-skewed, where for $k$-TREE, $s(C)^{\mathcal{O}(1)}$ circuits are produced and each of them is $2^{o(k)}$-skewed (see the direct application of the method of representative families to solve $k$-TREE in [FLPS16]). First, we state the aforementioned applications as a corollary of Theorem 4.2.

**Theorem 4.3.** *For any $0 < \epsilon < 1$, the* #$k$-PATH, #$q$-SET $p$-PACKING *with $k = qp$,* #$q$-DIMENSIONAL $p$-MATCHING *with $k = (q-1)p$ and* # GRAPH MOTIF *problems can be approximated with factor $(1 \pm \epsilon)$ and success probability at least $\frac{9}{10}$ in time $\mathcal{O}((2.619^k + |I|^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot |I|)$, where $k$ is the parameter and $|I|$ is the input size. Moreover, for any $0 < \epsilon < 1$, the* #$k$-TREE *(or, more generally,* #SUBGRAPH ISOMORPHISM *where the treewidth of input graph is bounded by a fixed constant) can be approximated with factor $(1 \pm \epsilon)$ and success probability at least $\frac{9}{10}$ in time $2.619^k \cdot \frac{1}{\epsilon^2} \cdot |I|^{\mathcal{O}(1)}$.*

For the sake of illustration, we will explicitly give here the circuit produced for #$k$-PATH, and verify that it is 1-skewed and additively simple. For the other problems, we refer the reader to the aforementioned references in this subsection (which also encompass #$k$-PATH).

*Proof of Theorem 4.3 for* #$k$-PATH. In the #$k$-PATH problem, we are given a digraph $D$ and a parameter $k \in \mathbb{N}$, and the objective is to count the number of solutions being $k$-paths—simple directed paths in $D$ that consist of exactly $k$ vertices.[5] We will only explicitly give a proof for the case of #$k$-PATH. To this end, we present a parsimonious reduction from #$k$-PATH to #MULTILINEAR MONOMIAL DETECTION on 1-skewed additively simple circuits where the output circuit is of size linear in the size of the input graph. By Theorem 4.2, this will yield that for any $0 < \epsilon < 1$, #$k$-PATH can be approximated with factor $(1 \pm \epsilon)$ and success probability at least $\frac{9}{10}$ in time $\mathcal{O}((2.619^k + |G|^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot |G|)$ (where $|G| = |V(G)| + |E(G)|$).

Given an instance $(G, k)$ of #$k$-PATH, we construct an arithmetic circuit $C$ as follows. The set of variables is $X = \{x_v : v \in V(G)\}$. For every pair $(v, p)$ of a vertex $v \in V(G)$ and integer $p \in \{1, \ldots, k\}$, we have two nodes $N_{v,p,+}$ and $N_{v,p,\times}$ if $p \geq 2$, and only one node $N_{v,p,\times}$ if $p = 1$. Every node $N_{v,p,\times}$ where $p = 1$ is a leaf labeled $x_v$. Every node $N_{v,p,+}$ is an addition node with outgoing arcs to all nodes $N_{u,p-1,\times}$ such that $(u, v) \in E(G)$. Every node $N_{v,p,\times}$ is a multiplication node with outgoing arcs to $N_{v,p,+}$ and $N_{v,1,\times}$. Lastly, we have a root node $N_+$

[5] The case where the input graph is undirected can be reduced to the case where it is directed by transforming each undirected edge to two arcs of opposing directions—this blows up the number of solutions by precisely 2.

with outgoing arcs to $N_{v,k,\times}$ for all $v \in V(G)$. This completes the description of the reduction. In what follows, we will use the abbreviation $C_{v,p}$ to denote $C_{N_{v,p,\times}}$.

First, it is clear that the construction correspond to an arithmetic circuit (in particular, the constructed digraph is acyclic) and that it is 1-skewed and additively simple. Moreover, the number of nodes is $\mathcal{O}(k|V(G)|)$ and the number of arcs is $\mathcal{O}(k\sum_{v\in V(G)} \mathsf{degree}_G(v)) = \mathcal{O}(k|E(G)|)$. Thus, is remains to prove that the reduction is correct—specifically, we will show that the number of $k$-paths in $G$ is equal to $\sum_{m\in \mathtt{MulLin}_k(P_C(X))} \mathtt{coeff}(P_C(X), m)$. To this end, we will prove the following claim.

**Claim 4.2.** *For every node $N_{v,p,\times}$ where $v \in V(G)$ and $p \in \{1, 2, \ldots, k\}$, we have that $\bigcup_{i=1}^{k} \mathtt{MulLin}_i(P_{C_{v,p}}(X)) = \mathtt{MulLin}_p(P_{C_{v,p}}(X)) = \{x_{v_{j_1}} x_{v_{j_2}} \cdots x_{v_{j_p}} : \{v_{j_1}, v_{j_2}, \ldots, v_{j_p}\}$ is the vertex set of a $p$-path in $G$ that ends at $v\}$, and the coefficient of every multilinear monomial $x_{v_{j_1}} x_{v_{j_2}} \cdots x_{v_{j_p}} \in \mathtt{MulLin}_p(P_{C_{v,p}}(X))$ is the number of $p$-paths in $G$ with vertex set $\{v_{j_1}, v_{j_2}, \ldots, v_{j_p}\}$ that end at $v$.*

*Proof.* We prove the claim by induction on $v$ corresponding to a topological order on $C$ (as an undirected acyclic graph). For leaf nodes $N_{v,p,\times}$, where $p = 1$, the claim is trivially true. Now, consider some multiplication node $N_{v,p,\times}$, and suppose that the claim is true for every node $N_{v',p',\times}$ that is reachable from $v$. On the one hand, from the construction of $C$, we have that $P_{C_{v,p}}(X) = x_v \cdot \sum_{u:(u,v)\in E(G)} P_{C_{u,p-1}}(X)$. On the other hand, each $p$-path in $G$ that ends at $v$ consists of a distinct $(p-1)$-path in $G$ that excludes $v$ to which we append $v$ at the end. By the inductive hypothesis, for every outgoing neighbor $u$ of $v$ and $p - 1$, we already know that the claim holds. These three statements together straightforwardly yield that the claim holds for $v$ and $p$ as well. □

Having proved this claim, we know that for every vertex $v \in V(G)$ and $p \in \{1, 2, \ldots, k\}$, the number of $p$-paths in $G$ that end at $v$ is equal to the sum $\sum_{m\in \mathtt{MulLin}_p(P_{C_{v,p}}(X))} \mathtt{coeff}(P_{C_{v,p}}(X), m)$. Therefore, the sum $\sum_{m\in \mathtt{MulLin}_k(P_C(X))} \mathtt{coeff}(P_C(X), m)$, which is equal to the sum $\sum_{v\in V(G)} \sum_{m\in \mathtt{MulLin}_k(P_{C_{v,k}}(X))} \mathtt{coeff}(P_{C_{v,k}}(X), m)$, is precisely the number of $k$-paths in $G$. □

# 5 Extension to Representation of Product Counters and #Multilinear Monomial Detection on General Circuits

In this section we will compute representative counter for counters implicitly described as the product of two counters, and then use it to derive an algorithm for #Multilinear Monomial Detection on general circuits.

**Definition 5.1 ((Implicit) Product Counter).** *Let $U$ be a universe. Let $p_1, p_2 \in \mathbb{N}_0$, and let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ and $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Then, the* product of $\mathfrak{C}_1$ and $\mathfrak{C}_2$, *denoted by $\mathfrak{C}_1 \times \mathfrak{C}_2$, is the counter $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ defined as follows: For each $P \in \mathcal{P} \to \mathbb{N}_0$, $\mathfrak{C}(P) = \displaystyle\sum_{\substack{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:\\ P_1\cap P_2=\emptyset,P_1\cup P_2=P}} \mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)$ (which equals*

*$\displaystyle\sum_{\substack{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:\\ P_1\cup P_2=P}} \mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)$).*

We remark that we will never be storing product counters explicitly but only representatives of them (so, computationally, the product counters will be stored and referred to only implicitly via their components $\mathcal{P}_1$ and $\mathcal{P}_2$); indeed, just storing product counters consumes too much time when $\mathcal{P}_1$ and $\mathcal{P}_2$ are large—even when defined as the product of two counters already reduced by Theorem 3.1 , the dependency on $k$ of the size of the support of the product counter can reach $4^k$.

In what follows, we first show (in Section 5.1) how to compute representative counters for product counters assuming that we have parsimonious families equipped with general membership and disjointness procedures. Additionally, we also show how to compute such parsimonious families, which just requires a minor adaptation of Section 3.2. Afterwards, we show (in Section 5.3) that the algorithm in Section 4.1, with very slight modification that in particular involves using the computations of representative counters and parsimonious families given in this section rather than just those in Section 3, already solves #Multilinear Monomial Detection on general circuits in the desired time. Specifically, the new ideas required to handle the product case are present in the first subsection ahead, where the other two only contain straightforward modifications of Section 3.

## 5.1 Extension to Computation of Representative Product Counters of Small Support

We first remind that, by making use of Lemma 3.1, we can focus on the alternative definition of representation that depends on a given family $\mathcal{F}$. Now, we directly proceed to the definition of our sampling procedure. Here, four important modifications are made. First, we do not iterate over every set in the support of the counter that we want to represent (and decide whether to keep it or not) since just enumerating the support can be too time consuming for us—remember that we represent the counter that we want to represent only implicitly; instead, we decide in advance how many sets to choose. Second, linked to the first, we still attempt to somewhat simulate $(\mathfrak{C},\mathcal{F},\mathcal{H})$-counter sampling, yet the probabilities considered there cannot be computed efficiently enough—specifically, the problematic issue is the determination of the values $\mathsf{assoc}_{\mathfrak{C},\mathcal{F},L}(P)$. Instead, we will slightly alter the probabilities, so they may be larger than before, but still small enough so that the support of the output counter will be small. This leads to the third and fourth modifications, with the third one being the necessity of having not just one parsimonious family, but two, as well as general membership and disjointness procedures (whose usage will be explicitly addressed in the time complexity analysis of the process), and the fourth one referring to some calculations done in a "preprocessing step", that is, before the actual sampling begins.

**Definition 5.2** (**$(\mathfrak{C}_1,\mathfrak{C}_2,\mathcal{F},\mathcal{H},L_1,L_2)$-Counter Sampling**)**.** *Let $U$ be a universe. Let $p_1,p_2,L_1,L_2\in\mathbb{N}_0$, and let $\mathcal{P}_1\subseteq\binom{U}{p_1},\mathcal{P}_2\subseteq\binom{U}{p_2}$ and $\mathcal{P}\subseteq\binom{U}{p}$ where $p=p_1+p_2$. Let $\mathfrak{C}_1:\mathcal{P}_1\to\mathbb{N}_0$ and $\mathfrak{C}_2:\mathcal{P}_2\to\mathbb{N}_0$ be two counters. Let $\mathcal{F},\mathcal{H}\subseteq 2^U$. Then, $(\mathfrak{C}_1,\mathfrak{C}_2,\mathcal{F},\mathcal{H},L_1,L_2)$-*counter sampling *is the randomized procedure that constructs a counter $\widehat{\mathfrak{C}}:\mathcal{P}\to\mathbb{N}_0$ as follows.*

1. **Preprocessing I: Neighbourhood Computations.** *For every $P_1\in\mathsf{supp}(\mathfrak{C}_1)$, compute $N_F(P_1)=\{F\in\mathcal{F}:P_1\subseteq F\}$ and $N_H(P_1)=\{H\in\mathcal{H}:P_1\subseteq H\}$. For every $P_2\in\mathsf{supp}(\mathfrak{C}_2)$, compute $N_F(P_2)=\{F\in\mathcal{F}:P_2\subseteq F\}$ and $N_H(P_2)=\{H\in\mathcal{H}:P_1\cap H=\emptyset\}$. For every $F\in\mathcal{F}$, compute $N_1(F)=\{P_1\in\mathsf{supp}(\mathfrak{C}_1):P_1\subseteq F\}$ and $N_2(F)=\{P_2\in\mathsf{supp}(\mathfrak{C}_2):P_2\subseteq F\}$. For every $H\in\mathcal{H}$, compute $N_1(H)=\{P_1\in\mathsf{supp}(\mathfrak{C}_1):P_1\subseteq H\}$ and $N_2(H)=\{P_2\in\mathsf{supp}(\mathfrak{C}_2):P_2\cap H=\emptyset\}$.*

2. **Preprocessing II: Approximate Domain Extension.** *For every $F\in\mathcal{F}$ and $H\in\mathcal{H}$, compute $N_1(F,H)=N_1(F)\cap N_1(H)$, $N_2(F,H)=N_2(F)\cap N_2(H)$ and $W(F)=\dfrac{1}{L_1}\cdot\sum_{H\in\mathcal{H}}\sum_{(P_1,P_2)\in N_1(F,H)\times N_2(F,H)}\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)$.*

3. **Preprocessing III: Probabilities to Select Sets From $\mathsf{supp}(\mathfrak{C}_1)$.** *For every* $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$, *compute* $\widetilde{\mathsf{prob}}(P_1) = \sum_{(F,H)\in N_F(P_1)\times N_H(P_1)} \sum_{P_2\in N_2(F,H)} \frac{\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)}{W(F)}$. *Afterwards, for every* $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$, *compute* $\mathsf{prob}(P_1) = \widetilde{\mathsf{prob}}(P_1)/W^\star$ *where* $W^\star = \sum_{P_1\in\mathsf{supp}(\mathfrak{C}_1)} \widetilde{\mathsf{prob}}(P_1)$.

4. **Preprocessing IV: Probabilities to Select Family Pairs.** *For every* $F \in \mathcal{F}$ *and* $H \in \mathcal{H}$, *compute* $W_2(F,H) = \sum_{P_2\in N_2(F,H)} \mathfrak{C}_2(P_2)$. *Further, for every* $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$, $F \in N_F(P_1)$ *and* $H \in N_H(P_1)$, *compute* $\mathsf{prob}_{P_1}(F,H) = W_2(F,H)/W_2(P_1)$ *where* $W_2(P_1) = \sum_{(F,H)\in N_F(P_1)\times N_H(P_1)} W_2(F,H)$.

5. **Sampling.** *For* $i = 1, 2, \ldots, t$ *where* $t = L_2 \cdot W^\star$:

   (a) *Randomly select one set from* $\mathsf{supp}(\mathfrak{C}_1)$, *where the probability to select* $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$ *is* $\mathsf{prob}(P_1)$. *Denote the selected set by* $P_1^i$.

   (b) *Randomly select one pair from* $N_F(P_1^i) \times N_H(P_1^i)$ *where the probability to select* $(F,H) \in N_F(P_1^i) \times N_H(P_1^i)$ *is* $\mathsf{prob}_{P_1}(F,H)$. *Denote the selected pair by* $(F^i, H^i)$.

   (c) *Randomly select one set from* $N_2(F^i, H^i)$ *where the probability to select* $P_2 \in N_2(F^i, H^i)$ *is* $\mathfrak{C}_2(P_2)/W_2(F^i, H^i)$. *Denote the selected set by* $P_2^i$.

   (d) *Compute* $\widetilde{\mathsf{prob}}(P_1^i, P_2^i) = \widetilde{\mathsf{prob}}(P_1) \cdot \frac{\mathfrak{C}_2(P_2)}{W_2(P_1)}$. *(We remark that this term is equal to* $\widetilde{\mathsf{prob}}(P_1) \cdot \sum_{\substack{F\in N_F(P_1)\cap N_F(P_2),\\ H\in N_H(P_1)\cap N_H(P_2)}} \mathsf{prob}_{P_1}(F,H) \cdot \frac{\mathfrak{C}_2(P_2)}{W_2(F,H)}$.*)*

6. **Output.** *Lastly,* $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ *is constructed as follows. For every* $P \in \mathcal{P}$, *define* $\mathfrak{C}(P) = \sum_{i\in\{1,\ldots,t\}: P_1^i\cup P_2^i = P} \frac{\mathfrak{C}_1(P_1^i)\cdot\mathfrak{C}_2(P_2^i)}{L_2\cdot\widetilde{\mathsf{prob}}(P_1^i,P_2^i)}$.

We begin with a trivial observation regarding the size of the support of the output counter.

**Observation 5.1.** *Let* $U$ *be a universe. Let* $p_1, p_2, L_1, L_2 \in \mathbb{N}_0$, *and let* $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ *and* $\mathcal{P} \subseteq \binom{U}{p}$ *where* $p = p_1 + p_2$. *Let* $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ *and* $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ *be two counters. Let* $\mathcal{F}, \mathcal{H} \subseteq 2^U$. *Then, the size of the support of the output counter* $\widehat{\mathfrak{C}}$ *of* $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$*-counter sampling is* $L_2 \cdot W^\star$.

Thus, to upper bound the size of the support, we proceed to upper bound $W^\star$. Towards this, we first assert that in Step 2 of the sampling procedure, we compute domain extensions almost correctly.

**Lemma 5.1.** *Let* $U$ *be a universe. Let* $p_1, p_2, L_1, L_2 \in \mathbb{N}_0$, *and let* $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ *and* $\mathcal{P} \subseteq \binom{U}{p}$ *where* $p = p_1 + p_2$. *Let* $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ *and* $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ *be two counters. Let* $\mathcal{F}, \mathcal{H} \subseteq 2^U$ *where* $\mathcal{H}$ *is an* $\epsilon$*-parsimonious* $(n, p_1, p_2)$*-universal family with respect to* $(\mathcal{P}_1, \mathcal{P}_2)$ *with correction factor* $L_1$. *Then, for every* $F \in \mathcal{F}$, *we have that* $(1-\epsilon)\mathfrak{C}_{\mathrm{ext}}(F) \leq W(F) \leq (1+\epsilon)\mathfrak{C}_{\mathrm{ext}}(F)$ *where* $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$ *and* $W$ *is as defined in* $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$*-counter sampling.*

*Proof.* Consider some set $F \in \mathcal{F}$. Observe that, because $\mathcal{H}$ is an $\epsilon$-parsimonious $(n, p_1, p_2)$-universal family with respect to $(\mathcal{P}_1, \mathcal{P}_2)$ with correction factor $L_1$, we have the following bounds: for every pair of disjoint sets $P_1 \in \mathcal{P}_1$ and $P_2 \in \mathcal{P}_2$, it holds that $(1-\epsilon)\cdot L_1 \leq |\mathcal{H}[P_1, P_2]| \leq (1+\epsilon)\cdot$

$L_1$. Moreover, note that for every pair of sets $P_1 \in \mathcal{P}_1$ and $P_2 \in \mathcal{P}_2$, if $(P_1, P_2) \in N_1(H) \times N_2(H)$ for some $H \in \mathcal{H}$, then $P_1 \cap P_2 = \emptyset$. Thus, we have that

$$\begin{aligned}
W(F) &= \frac{1}{L_1} \cdot \sum_{H \in \mathcal{H}} \sum_{(P_1,P_2) \in N_1(F,H) \times N_2(F,H)} \mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2) \\
&= \frac{1}{L_1} \cdot \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2: \\ P_1, P_2 \subseteq F, P_1 \cap P_2 = \emptyset}} \sum_{\substack{H \in \mathcal{H}: \\ P_1 \subseteq H, P_2 \cap H = \emptyset}} \mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2) \\
&= \frac{1}{L_1} \cdot \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2: \\ P_1, P_2 \subseteq F, P_1 \cap P_2 = \emptyset}} |\mathcal{H}[P_1, P_2]| \cdot \mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2) \\
&\leq (1+\epsilon) \cdot \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2: \\ P_1, P_2 \subseteq F, P_1 \cap P_2 = \emptyset}} \mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2) \\
&= (1+\epsilon) \cdot \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2: \\ P_1 \cup P_2 = P}} \mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2) \\
&= (1+\epsilon) \cdot \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \mathfrak{C}(P) = (1+\epsilon)\mathfrak{C}_{\text{ext}}(F).
\end{aligned}$$

Symmetrically, we derive that $(1-\epsilon)\mathfrak{C}_{\text{ext}}(F) \leq W(F)$. □

We are now ready to upper bound $W^\star$.

**Lemma 5.2.** *Let $U$ be a universe. Let $p_1, p_2, L_1, L_2 \in \mathbb{N}_0$, and let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ and $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F}, \mathcal{H} \subseteq 2^U$ where $\mathcal{H}$ is an $\epsilon$-parsimonious $(n, p_1, p_2)$-universal family with respect to $(\mathcal{P}_1, \mathcal{P}_2)$ with correction factor $L_1$. Then, $W^\star \leq \frac{1+\epsilon}{1-\epsilon} \cdot L_1 \cdot |\mathcal{F}|$ where $W^\star$ is as defined in $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling.*

*Proof.* Denote $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$. Thus, we have that

$$\begin{aligned}
W^\star &= \sum_{P_1 \in \mathsf{supp}(\mathfrak{C}_1)} \widetilde{\mathsf{prob}}(P_1)\\
&= \sum_{P_1 \in \mathsf{supp}(\mathfrak{C}_1)} \sum_{(F,H) \in N_F(P_1) \times N_H(P_1)} \sum_{P_2 \in N_2(F,H)} \frac{\mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2)}{W(F)}\\
&\leq \frac{1}{1-\epsilon} \cdot \sum_{P_1 \in \mathsf{supp}(\mathfrak{C}_1)} \sum_{(F,H) \in N_F(P_1) \times N_H(P_1)} \sum_{P_2 \in N_2(F,H)} \frac{\mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2)}{\mathfrak{C}_{\mathrm{ext}}(F)}\\
&= \frac{1}{1-\epsilon} \cdot \sum_{F \in \mathcal{F}} \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2:\\ P_1, P_2 \subseteq F, P_1 \cap P_2 = \emptyset}} \sum_{\substack{H \in \mathcal{H}:\\ P_1 \subseteq H, P_2 \cap H = \emptyset}} \frac{\mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2)}{\mathfrak{C}_{\mathrm{ext}}(F)}\\
&= \frac{1}{1-\epsilon} \cdot \sum_{F \in \mathcal{F}} \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2:\\ P_1, P_2 \subseteq F, P_1 \cap P_2 = \emptyset}} |\mathcal{H}[P_1, P_2]| \cdot \frac{\mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2)}{\mathfrak{C}_{\mathrm{ext}}(F)}\\
&\leq \frac{1+\epsilon}{1-\epsilon} \cdot L_1 \cdot \sum_{F \in \mathcal{F}} \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2:\\ P_1, P_2 \subseteq F, P_1 \cap P_2 = \emptyset}} \frac{\mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2)}{\mathfrak{C}_{\mathrm{ext}}(F)}\\
&= \frac{1+\epsilon}{1-\epsilon} \cdot L_1 \cdot \sum_{F \in \mathcal{F}} \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \sum_{\substack{P_1 \in \mathcal{P}_1, P_2 \in \mathcal{P}_2:\\ P_1 \cup P_2 = P}} \frac{\mathfrak{C}_1(P_1) \cdot \mathfrak{C}_2(P_2)}{\mathfrak{C}_{\mathrm{ext}}(F)}\\
&= \frac{1+\epsilon}{1-\epsilon} \cdot L_1 \cdot \sum_{F \in \mathcal{F}} \sum_{P \in \binom{F}{p} \cap \mathcal{P}} \frac{\mathfrak{C}(P)}{\mathfrak{C}_{\mathrm{ext}}(F)}\\
&= \frac{1+\epsilon}{1-\epsilon} \cdot L_1 \cdot \sum_{F \in \mathcal{F}} 1 = \frac{1+\epsilon}{1-\epsilon} \cdot L_1 \cdot |\mathcal{F}|.
\end{aligned}$$

Here, the first inequality followed from Lemma 5.1, and the second inequality followed because $\mathcal{H}$ is an $\epsilon$-parsimonious $(n, p_1, p_2)$-universal family with respect to $(\mathcal{P}_1, \mathcal{P}_2)$ with correction factor $L_1$. □

From Observation 5.1 and Lemma 5.2, we derive the following corollary.

**Corollary 5.1.** *Let $U$ be a universe. Let $p_1, p_2, L_1, L_2 \in \mathbb{N}_0$, and let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ and $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F}, \mathcal{H} \subseteq 2^U$ where $\mathcal{H}$ is an $\epsilon$-parsimonious $(n, p_1, p_2)$-universal family with respect to $(\mathcal{P}_1, \mathcal{P}_2)$ with correction factor $L_1$. Then, the size of the support of the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling is upper bounded by $\frac{1+\epsilon}{1-\epsilon} \cdot L_1 \cdot L_2 \cdot |\mathcal{F}|$.*

We now show that the output counter represents in expectation the (implicit) input product counter.

**Lemma 5.3.** *Let $U$ be a universe. Let $p_1, p_2, L_1, L_2 \in \mathbb{N}_0$, and let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ and $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F}, \mathcal{H} \subseteq 2^U$. The output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling represents in expectation $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$.*

*Proof.* Consider some set $P \in \mathcal{P}$. Then, by the definition of $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling and linearity of expectation, we have that

$$
\begin{aligned}
&E[\widehat{\mathfrak{C}}(P)]\\
&= E[\sum_{i\in\{1,\dots,t\}:P_1^i\cup P_2^i=P} \frac{\mathfrak{C}_1(P_1^i)\cdot\mathfrak{C}_2(P_2^i)}{L_2\cdot\widetilde{\mathsf{prob}}(P_1^i,P_2^i)}]\\
&= \sum_{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:P_1\cup P_2=P}\sum_{i=1}^{t} Pr(P_1=P_1^i\wedge P_2=P_2^i)\cdot\frac{\mathfrak{C}_1(P_1^i)\cdot\mathfrak{C}_2(P_2^i)}{L_2\cdot\widetilde{\mathsf{prob}}(P_1^i,P_2^i)}\\
&= \sum_{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:P_1\cup P_2=P}\sum_{i=1}^{t}\left(\mathsf{prob}(P_1)\cdot\sum_{\substack{F\in N_F(P_1)\cap N_F(P_2),\\ H\in N_H(P_1)\cap N_H(P_2)}}\mathsf{prob}_{P_1}(F,H)\cdot\frac{\mathfrak{C}_2(P_2)}{W_2(F,H)}\right)\cdot\frac{\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)}{L_2\cdot\widetilde{\mathsf{prob}}(P_1,P_2)}\\
&= \sum_{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:P_1\cup P_2=P} t\cdot\frac{\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)}{L_2\cdot W^\star}\\
&= \sum_{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:P_1\cup P_2=P} \mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)=\mathfrak{C}(P)
\end{aligned}
$$

Since the choice of $P$ was arbitrary, we derive that $\widehat{\mathfrak{C}}$ represents in expectation $\mathfrak{C}$. □

Having proved Lemma 5.3, we derive the following observation (just like Observation 3.2 followed from Observation 3.1).

**Observation 5.2.** *Let $U$ be a universe. Let $p_1, p_2, L_1, L_2 \in \mathbb{N}_0$, and let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ and $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F}, \mathcal{H} \subseteq 2^U$. For any set $F \subseteq U$, for the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling, we have that $E[\widehat{\mathfrak{C}}_{\text{ext}}(F)] = \mathfrak{C}_{\text{ext}}(F)$ where $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$.*

We now turn to prove that the output counter is likely to represent the (implicit) input product counter.

**Lemma 5.4.** *Let $U$ be a universe of size $n$. Let $0 < \epsilon < 1$ and $0 < \delta < 1$. Let $p_1, p_2, L_1, L_2 \in \mathbb{N}_0$ and $c \geq 1$ with $L_2 \geq \frac{2(1+\delta)}{\epsilon^2}\ln(2c|\mathcal{F}|)$, and let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$ and $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F}, \mathcal{H} \subseteq 2^U$ where $\mathcal{H}$ is an $\delta$-parsimonious $(n, p_1, p_2)$-universal family with respect to $(\mathcal{P}_1, \mathcal{P}_2)$ with respect to $(\mathcal{P}_1, \mathcal{P}_2)$ with correction factor $L_1$. Then, the probability that $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$ and the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling are $(\epsilon, \mathcal{F})$-similar is at least $1 - \frac{1}{c}$.*

*Proof.* Consider some $F \in \mathcal{F}$. For every $P_1 \in \mathcal{P}_1$, $P_2 \in \mathcal{P}_2$ and $i \in \{1, 2, \dots, t\}$, let $X_{P_1,P_2,i}$ be a random variable that is equal to $\dfrac{\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)}{L_2\cdot\widetilde{\mathsf{prob}}(P_1,P_2)}$ if the sampling procedure selects $P^i = P$ (which occurs with probability $\mathsf{prob}(P_1)\cdot\sum_{\substack{F\in N_F(P_1)\cap N_F(P_2),\\ H\in N_H(P_1)\cap N_H(P_2)}}\mathsf{prob}_{P_1}(F,H)\cdot\dfrac{\mathfrak{C}_2(P_2)}{W_2(F,H)}$), and 0 otherwise. Notice that

$$\begin{aligned}\frac{\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)}{L_2\cdot\widetilde{\mathsf{prob}}(P_1,P_2)} &= \frac{\mathfrak{C}_1(P_1)\cdot W_2(P_1)}{L_2\cdot\widetilde{\mathsf{prob}}(P_1)}\\ &= \frac{\mathfrak{C}_1(P_1)\cdot W_2(P_1)\cdot W(F)}{L_2\cdot\sum_{(F,H)\in N_F(P_1)\times N_H(P_1)}\sum_{P_2\in N_2(F,H)}\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)}\\ &= \frac{\mathfrak{C}_1(P_1)\cdot\sum_{(F,H)\in N_F(P_1)\times N_H(P_1)}\sum_{P_2\in N_2(F,H)}\mathfrak{C}_2(P_2)\cdot W(F)}{L_2\cdot\sum_{(F,H)\in N_F(P_1)\times N_H(P_1)}\sum_{P_2\in N_2(F,H)}\mathfrak{C}_1(P_1)\cdot\mathfrak{C}_2(P_2)}\\ &= \frac{W(F)}{LL_2}\\ &\leq \frac{(1+\delta)\mathfrak{C}_{\text{ext}}(F)}{L_2}.\end{aligned}$$

Here, the inequality followed from Lemma 5.1. Denote $M = \dfrac{(1+\delta)\mathfrak{C}_{\text{ext}}(F)}{L_2}$. Moreover, for every $P\in\mathcal{P}$ and $i\in\{1,2,\dots,t\}$, denote $X'_{P_1,P_2,i} = X_{P_1,P_2,i}/M$, and observe that $X'_{P_1,P_2,i}$ is upper bounded by 1.

Denote $\overline{X} = \displaystyle\sum_{i=1}^{t}\sum_{\substack{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:\\ P_1,P_2\subseteq F,P_1\cap P_2=\emptyset}} X_{P_1,P_2,i}$ and $\overline{X}' = \displaystyle\sum_{i=1}^{t}\sum_{\substack{P_1\in\mathcal{P}_1,P_2\in\mathcal{P}_2:\\ P_1,P_2\subseteq F,P_1\cap P_2=\emptyset}} X'_{P_1,P_2,i}$. Observe that $\overline{X} = \widehat{\mathfrak{C}}_{\text{ext}}(F)$, and hence $\overline{X}' = \dfrac{\widehat{\mathfrak{C}}_{\text{ext}}(F)}{M} = \dfrac{L_2\cdot\widehat{\mathfrak{C}}_{\text{ext}}(F)}{(1+\delta)\cdot\mathfrak{C}_{\text{ext}}(F)}$. Further, by Observation 5.2, $E[\widehat{\mathfrak{C}}_{\text{ext}}(F)] = \mathfrak{C}_{\text{ext}}(F)$, and therefore $E[\overline{X}] = \mathfrak{C}_{\text{ext}}(F)$ and $E[\overline{X}'] = \dfrac{L_2}{1+\delta}$. Additionally, we derive that $(1-\epsilon)\cdot\widehat{\mathfrak{C}}_{\text{ext}}(F) > \mathfrak{C}_{\text{ext}}(F)$ or $\mathfrak{C}_{\text{ext}}(F) > (1+\epsilon)\cdot\widehat{\mathfrak{C}}_{\text{ext}}(F)$ if and only if $(1-\epsilon)\cdot\overline{X}' > E[\overline{X}']$ or $E[\overline{X}'] > (1+\epsilon)\cdot\overline{X}'$. By Chernoff Bound (Proposition 2.2), we derive that $Pr(|\overline{X}' - E[\overline{X}']| > \epsilon E[\overline{X}']) \leq 2e^{-\frac{\epsilon^2 E[\overline{X}']}{2}} = 2e^{-\frac{\epsilon^2 L_2}{2(1+\delta)}} \leq 2e^{-\ln(2c|\mathcal{F}|)} = \dfrac{1}{c|\mathcal{F}|}$.

As the choice of $F\in\mathcal{F}$ was arbitrary, union bound implies that the probability that there exists $F\in\mathcal{F}$ such that $(1-\epsilon)\cdot\widehat{\mathfrak{C}}_{\text{ext}}(F) > \mathfrak{C}_{\text{ext}}(F)$ or $\mathfrak{C}_{\text{ext}}(F) > (1+\epsilon)\cdot\widehat{\mathfrak{C}}_{\text{ext}}(F)$ is upper bounded by $|\mathcal{F}|\cdot\dfrac{1}{c|\mathcal{F}|} = \dfrac{1}{c}$. Thus, the probability that $\mathfrak{C}$ and $\widehat{\mathfrak{C}}$ are $(\epsilon,\mathcal{F})$-similar is at least $1-\frac{1}{c}$. □

We proceed to consider the time complexity of the sampling procedure.

**Lemma 5.5.** *Let $U$ be a universe of size $n$. Let $0 < \delta_{\mathcal{F}} < 1$ and $0 < \delta_{\mathcal{H}} < 1$. Let $p_1, p_2, q, L_1, L_2 \in \mathbb{N}_0$ and $c \geq 1$, and let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$, $\mathcal{P}\subseteq\binom{U}{p}$ where $p = p_1 + p_2$, and $\mathcal{Q}\subseteq\binom{U}{q}$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F},\mathcal{H}\subseteq 2^U$ where:*

- *$\mathcal{F}$ is an $\delta_{\mathcal{F}}$-parsimonious $(n,p,q)$-universal family with respect to $(\mathcal{P},\mathcal{Q})$ with a $T_1^{\mathcal{F}}$-membership query procedure with respect to $\mathcal{P}_1$ and a $T_2^{\mathcal{F}}$-membership query procedure with respect to $\mathcal{P}_2$.*
- *$\mathcal{H}$ is an $\delta_{\mathcal{H}}$-parsimonious $(n,p_1,p_2)$-universal family with respect to $(\mathcal{P}_1,\mathcal{P}_2)$ with a $T_{\text{con}}^{\mathcal{H}}$-membership query procedure and a $T_{\text{dis}}^{\mathcal{H}}$-membership query procedure.*

*Then, the running time of $(\mathfrak{C}_1,\mathfrak{C}_2,\mathcal{F},\mathcal{H},L_1,L_2)$-counter sampling is bounded by $\mathcal{O}(\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}}\cdot T_{\text{con}}^{\mathcal{H}}) + |\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}}\cdot T_{\text{dis}}^{\mathcal{H}}) + \frac{1+\delta_{\mathcal{H}}}{1-\delta_{\mathcal{H}}}\cdot L_1\cdot L_2\cdot|\mathcal{F}|)$.*

*Proof.* First, we note that for every $P_1\in\mathcal{P}_1$, $|N_F(P_1)|\leq T_1^{\mathcal{F}}$ and $|N_H(P_1)|\leq T_{\text{con}}^{\mathcal{H}}$, and for every $P_2\in\mathcal{P}_2$, $|N_F(P_2)|\leq T_2^{\mathcal{F}}$ and $|N_H(P_2)|\leq T_{\text{dis}}^{\mathcal{H}}$. This also implies that Step 1 can be performed in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}} + T_{\text{con}}^{\mathcal{H}}) + |\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}} + T_{\text{dis}}^{\mathcal{H}}))$.

For Step 2, first note that $N_1(F,H)$ and $N_2(F,H)$ simultaneously for all $F \in \mathcal{F}$ and $H \in \mathcal{H}$ can be computed in time $\mathcal{O}(\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}) + |\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}} \cdot T_{\text{dis}}^{\mathcal{H}}))$ by iterating over every set $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$ and adding every pair of set $F \in N_F(P_1)$ and $H \in N_H(P_1)$ to $N_1(F,H)$ (there are at most $T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}$ such pairs), as well as iterating over every set $P_2 \in \mathsf{supp}(\mathfrak{C}_2)$ and adding every pair of set $F \in N_F(P_2)$ and $H \in N_H(P_2)$ to $N_2(F,H)$ (there are at most $T_2^{\mathcal{F}} \cdot T_{\text{dis}}^{\mathcal{H}}$ such pairs). Second, note that $W(F)$ simultaneously for all $F \in \mathcal{F}$ can be computed in time $\mathcal{O}(\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}) + |\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}} \cdot T_{\text{dis}}^{\mathcal{H}}))$ by first computing $SUM_1(F,H) := \sum_{P_1 \in N_2(F,H)} \mathfrak{C}_1(P_1)$ for all $F \in \mathcal{F}$ and $H \in \mathcal{H}$ altogether in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}))$, as well as computing $SUM_2(F,H) := \sum_{P_2 \in N_2(F,H)} \mathfrak{C}_2(P_2)$ for all $F \in \mathcal{F}$ and $H \in \mathcal{H}$ altogether in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}} \cdot T_{\text{dis}}^{\mathcal{H}}))$, and secondly computing for all $F \in \mathcal{F}$ the expression $\frac{1}{L_1} \sum_{H \in \mathcal{H}} SUM_1(F,H) \cdot SUM_2(F,H)$ (which equals $W(F)$) altogether in time $\mathcal{O}(|\mathcal{F}||\mathcal{H}|)$, which is upper bounded by $\mathcal{O}(\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}) + |\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}} \cdot T_{\text{dis}}^{\mathcal{H}}))$.

For Step 3, for each set $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$, we can now compute $\widetilde{\mathsf{prob}}(P_1)$ in time $\mathcal{O}(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}})$ because it simply equals $\sum_{(F,H) \in N_F(P_1) \times N_H(P_1)} \frac{SUM_2(F,H)}{W(F)}$. Thus, we compute $\widetilde{\mathsf{prob}}(P_1)$ for all $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$ altogether in time $\mathcal{O}(|\mathsf{supp}(\mathcal{C}_1)| \cdot T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}})$. Afterwards, $W^{\star}$ can clearly be computed in time $\mathcal{O}(|\mathsf{supp}(\mathcal{C}_1)|)$, and then, $\mathsf{prob}(P_1)$ for all $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$ altogether can also clearly be computed in time $\mathcal{O}(|\mathsf{supp}(\mathcal{C}_1)|)$.

For Step 4, for each $F \in \mathcal{F}$ and $H \in \mathcal{H}$, observe that $W_2(F,H)$ equals $SUM_2(F,H)$ and hence has already been computed. For each $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$, we compute $W_2(P_1)$ in time $\mathcal{O}(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}})$. Then, for each $P_1 \in \mathsf{supp}(\mathfrak{C}_1)$, $F \in N_F(P_1)$ and $H \in N_H(P_1)$ (there are at most $|\mathsf{supp}(\mathfrak{C}_1)| \cdot T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}$ such triples to consider), we compute $\mathsf{prob}_{P_1}(F,H)$ in time $\mathcal{O}(1)$. Thus, altogether this step is performed in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C}_1)| \cdot T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}})$.

Due to all preprocessing steps, note that Step 5 is performed in time $\mathcal{O}(L_2 \cdot W^{\star})$. By Lemma 5.2, this is upper bounded by $\mathcal{O}(\frac{1+\delta_{\mathcal{H}}}{1-\delta_{\mathcal{H}}} \cdot L_1 \cdot L_2 \cdot |\mathcal{F}|)$.

Lastly, for Step 6, we can compute $\mathfrak{P}$ for all $P \in \mathsf{supp}(\mathfrak{C})$ simultaneously by iterating over each $i \in \{1,\ldots,t\}$, inserting $P = P_1^i \cup P_2^i$ to the support of the constructed counter $\mathfrak{C}$ (if it has not already been inserted), and adding $\frac{\mathfrak{C}_1(P_1^i) \cdot \mathfrak{C}_2(P_2^i)}{L_2 \cdot \widetilde{\mathsf{prob}}(P_1^i, P_2^i)}$ to the value assigned to $P$. Overall, this takes time $\mathcal{O}(t)$ as did the previous step. □

We conclude with the main theorem of this section.

**Theorem 5.1.** *Let $U$ be a universe. Let $0 < \epsilon < 1$. Let $p_1, p_2, q, L_1, c \in \mathbb{N}_0$, $c \geq 1$. Let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$, $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$, and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F}, \mathcal{H} \subseteq 2^U$ where:*

- *$\mathcal{F}$ is a $\frac{\epsilon}{5}$-parsimonious $(n,p,q)$-universal family with respect to $(\mathcal{P},\mathcal{Q})$ with a $T_1^{\mathcal{F}}$-membership query procedure with respect to $\mathcal{P}_1$ and a $T_2^{\mathcal{F}}$-membership query procedure with respect to $\mathcal{P}_2$.*

- *$\mathcal{H}$ is a $\frac{1}{2}$-parsimonious $(n,p_1,p_2)$-universal family with respect to $(\mathcal{P}_1,\mathcal{P}_2)$ with correction factor $L_1$, a $T_{\text{con}}^{\mathcal{H}}$-membership query procedure and a $T_{\text{dis}}^{\mathcal{H}}$-membership query procedure.*

*Denote $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$. Then, a counter $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ such that*

1. *$\widehat{\mathfrak{C}}$ necessarily (with probability 1) represents in expectation $\mathfrak{C}$ as well as satisfies $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq \mathcal{O}(L_1 \cdot \frac{1}{\epsilon^2} \cdot |\mathcal{F}| \cdot \ln(c|\mathcal{F}|))$, and*

2. *with success probability at least $1 - \frac{1}{c}$, $\widehat{\mathfrak{C}}$ $(\epsilon,q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$,*

*can be computed in time $\mathcal{O}(|\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}) + |\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}} \cdot T_{\text{dis}}^{\mathcal{H}}) + L_1 \cdot \frac{1}{\epsilon^2} \cdot |\mathcal{F}| \cdot \ln(2c|\mathcal{F}|))$.*

*Proof.* Let $L_2 = \frac{3}{(\frac{\epsilon}{5})^2}\ln(2c|\mathcal{F}|)$. We use $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling. The claim that $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq \mathcal{O}(L_1 \cdot \frac{1}{\epsilon^2} \cdot |\mathcal{F}| \cdot \ln(2c|\mathcal{F}|))$ directly follows from Lemma 5.3 (where $\epsilon$ in that statement is equal to $\frac{1}{2}$). The claim that $\widehat{\mathfrak{C}}$ represents in expectation $\mathfrak{C}$ directly follows from Lemma 5.3. By Lemma 5.4, the probability that $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$ and the output counter $\widehat{\mathfrak{C}}$ of $(\mathfrak{C}_1, \mathfrak{C}_2, \mathcal{F}, \mathcal{H}, L_1, L_2)$-counter sampling are $(\frac{1}{5\epsilon}, \mathcal{F})$-similar is at least $1 - \frac{1}{c}$, in which case, Lemma 3.1 implies that $\widehat{\mathfrak{C}}$ $(\epsilon, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$. This completes the proof. □

As already explained in the proof of Claim 4.1, we are not be able to argue that the output of our algorithm for #Multilinear Monomial Detection is correct in expectation, which will prevent us from re-running that algorithm multiple times in order to improve its error. Instead, we will re-run each computation of representative families locally to improve its error. The proof of this lemma is similar to the proof of Lemma 3.11.

**Lemma 5.6.** *Let $0 < \epsilon \leq \delta < 1$. Suppose that we have an algorithm that give a counter $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ corresponding to some $\mathcal{P} \subseteq \binom{U}{p}$ for some given universe $U$ of size $n$ and $k, p, c \in \mathbb{N}$, computes a counter that $(\delta, k-p)$-represents $\mathfrak{C}$ with respect to some $\mathcal{Q} \subseteq \binom{U}{k-p}$ with success probability $1 - \frac{1}{2c}$, and necessarily represents in expectation $\mathfrak{C}$ and has support size $S_{\delta,\mathfrak{C},n,k,p,c}$, in time $T_{\delta,\mathfrak{C},n,k,p,c}$. Then, we have an algorithm that give a counter $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ corresponding to some $\mathcal{P} \subseteq \binom{U}{p}$ for some given universe $U$ and $k, p, c \in \mathbb{N}$, computes a counter that $(\epsilon, k-p)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$ with success probability $1 - \frac{1}{c}$, and necessarily has support size at most $\mathcal{O}(\frac{1}{\epsilon^2}k\log(cn) \cdot S_{\delta,\mathfrak{C},n,k,p,c'})$, in time $\mathcal{O}(\frac{1}{\epsilon^2}k\log(cn) \cdot T_{\delta,\mathfrak{C},n,k,p,c'})$ where $c' = \frac{8}{\epsilon^2}k\ln(4cn) \cdot c$.*

*Proof.* Let $\mathsf{ALG1}$ denote the algorithm given in the supposition of the lemma. Let $0 < \epsilon, \delta < 1$. Then, we design an algorithm $\mathsf{ALG2}$ as follows. Given an input $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ corresponding to some $\mathcal{P} \subseteq \binom{U}{p}$ for some given universe $U$ of size $n$ and $k, p, c \in \mathbb{N}$, $\mathsf{ALG2}$ executes the following operations, where $t = \frac{2(1+\delta)}{\epsilon^2}k\ln(4cn) \leq \frac{4}{\epsilon^2}k\ln(4cn)$.

1. For $i = 1, 2, \ldots t$: Call $\mathsf{ALG1}$ with the same input but success parameter $c' = t \cdot 2c$ and let $\widehat{\mathfrak{C}}_i$ denote the result.

2. Output $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}$ defined as follows. For every $P \in \mathcal{P}$, define $\widehat{\mathfrak{C}}(P) = \frac{1}{t} \cdot \sum_{i=1}^{t} \widehat{\mathfrak{C}}_i(P)$.

First, it is clear that the support size and the time complexity are as stated in the lemma. Second, by union bound, with success probability at least $1 - \frac{t}{c'} \geq 1 - \frac{1}{2c}$, all the calls $\mathsf{ALG2}$ makes to $\mathsf{ALG1}$ are successful. Thus, by union bound, to prove the lemma, it suffices to prove that under the assumption that all the calls made to $\mathsf{ALG1}$ are successful, with probability at least $1 - \frac{1}{2c}$, it holds that $\widehat{\mathfrak{C}}$ $(\epsilon, k-p)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$. Since there are at most $n^k$ choices for $Q \in \mathcal{Q}$, by union bound, it suffices to consider some arbitrary $Q \in \mathcal{Q}$, and show that with probability at least $1 - \frac{1}{2c \cdot n^k}$, the following condition is satisfied.

$$(1-\epsilon) \cdot \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \mathfrak{C}(P) \leq \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P) \leq (1+\epsilon) \cdot \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \mathfrak{C}(P).$$

Denote $X = \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \mathfrak{C}(P)$, $Y = \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \widehat{\mathfrak{C}}(P)$. For all $i \in \{1, 2, \ldots, t\}$, denote $Y_i = \sum_{P \in \mathcal{P} : P \cap Q = \emptyset} \widehat{\mathfrak{C}}_i(P)$ and $Y_i' = \frac{Y_i}{(1+\delta)X}$. Moreover, denote $Z' = \sum_{i=1}^{t} Y_i'$. Notice that $(1-\epsilon)X \leq Z \leq (1+\epsilon)X$ if and only if $(1-\epsilon)\frac{t}{(1+\delta)} \leq Z' \leq (1+\epsilon)\frac{t}{(1+\delta)}$, and thus it suffices to consider the probability that the latter event occurs. Since all calls are assumed to be successful, we have that $0 \leq Y_i' \leq 1$. Moreover, by linearity of expectation, $E[Z'] = \sum_{i=1}^{t} E[Y_i'] = \sum_{i=1}^{t} \frac{E[Y_i]}{(1+\delta)X} = t/(1+\delta)$. Therefore, $(1-\epsilon)\frac{t}{(1+\delta)} \leq Z' \leq (1+\epsilon)\frac{t}{(1+\delta)}$ if and only if $|Z' - E[Z']| \leq$

$\epsilon E[Z']$, and thus it further suffices to consider the probability that the latter event occurs. By Chernoff Bound (Proposition 2.2), we have that

$$\begin{aligned} Pr(|Z' - E[Z']| > \epsilon E[Z']) &\leq 2e^{-\frac{\epsilon^2 E[Z']}{2}} \\ &= 2e^{-\frac{\epsilon^2 t}{2(1+\delta)}} \\ &= 2e^{-k\ln(4cn)} = 2\frac{1}{(4nc)^k} \leq \frac{1}{2c \cdot n^k}. \end{aligned}$$

Thus, $|Z' - E[Z']| \leq \epsilon E[Z']$ with probability at least $1 - \frac{1}{2c \cdot n^k}$. As claimed above, this completes the proof. □

From Theorem 5.1 and Lemma 5.6, we derive the following corollary.

**Corollary 5.2.** *Let $U$ be a universe. Let $0 < \epsilon \leq \delta < 1$. Let $p_1, p_2, q, L_1, c \in \mathbb{N}_0$, $c \geq 1$. Let $\mathcal{P}_1 \subseteq \binom{U}{p_1}, \mathcal{P}_2 \subseteq \binom{U}{p_2}$, $\mathcal{P} \subseteq \binom{U}{p}$ where $p = p_1 + p_2$, and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathfrak{C}_1 : \mathcal{P}_1 \to \mathbb{N}_0$ and $\mathfrak{C}_2 : \mathcal{P}_2 \to \mathbb{N}_0$ be two counters. Let $\mathcal{F}, \mathcal{H} \subseteq 2^U$ where:*

- *$\mathcal{F}$ is a $\frac{\delta}{5}$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$ with a $T_1^{\mathcal{F}}$-membership query procedure with respect to $\mathcal{P}_1$ and a $T_2^{\mathcal{F}}$-membership query procedure with respect to $\mathcal{P}_2$.*
- *$\mathcal{H}$ is a $\frac{1}{2}$-parsimonious $(n, p_1, p_2)$-universal family with respect to $(\mathcal{P}_1, \mathcal{P}_2)$ with correction factor $L_1$, a $T_{\text{con}}^{\mathcal{H}}$-membership query procedure and a $T_{\text{dis}}^{\mathcal{H}}$-membership query procedure.*

*Denote $\mathfrak{C} = \mathfrak{C}_1 \times \mathfrak{C}_2$. Then, a counter $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ such that*

1. *$\widehat{\mathfrak{C}}$ necessarily (with probability 1) satisfies $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq \mathcal{O}(\frac{1}{\epsilon^2} \cdot L_1 \cdot \frac{1}{\delta^2} \cdot |\mathcal{F}| \cdot \ln(\frac{1}{\epsilon} k \ln(nc)|\mathcal{F}|) \cdot \ln(nc))$, and*
2. *with success probability at least $1 - \frac{1}{c}$, $\widehat{\mathfrak{C}}$ $(\epsilon, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$,*

*can be computed in time*

$$\mathcal{O}\left(\left(\frac{1}{\epsilon^2} k \log(nc) \cdot (|\mathsf{supp}(\mathfrak{C}_1)|(T_1^{\mathcal{F}} \cdot T_{\text{con}}^{\mathcal{H}}) + |\mathsf{supp}(\mathfrak{C}_2)|(T_2^{\mathcal{F}} \cdot T_{\text{dis}}^{\mathcal{H}}) + L_1 \cdot \frac{1}{\epsilon^2} \cdot |\mathcal{F}| \cdot \ln(\frac{1}{\epsilon} k \ln(nc)|\mathcal{F}|)\right)\right).$$

Similarly, from Theorem 3.1 and Lemma 5.6, we derive the following corollary.

**Corollary 5.3.** *Let $U$ be a universe. Let $0 < \epsilon, \delta < 1$, $p, q, c \in \mathbb{N}_0$, $\mathcal{P} \subseteq \binom{U}{p}$ and $\mathcal{Q} \subseteq \binom{U}{q}$. Let $\mathcal{F} \subseteq 2^U$ be an $\frac{1}{5}\delta$-parsimonious $(n, p, q)$-universal family with respect to $(\mathcal{P}, \mathcal{Q})$, equipped with a $T$-membership query procedure. Let $\mathfrak{C} : \mathcal{P} \to \mathbb{N}_0$ be a counter. Then, a counter $\widehat{\mathfrak{C}} : \mathcal{P} \to \mathbb{N}_0$ such that with success probability at least $1 - \frac{1}{c}$, $\widehat{\mathfrak{C}}$ $(\epsilon, q)$-represents $\mathfrak{C}$ with respect to $\mathcal{Q}$ and satisfies $|\mathsf{supp}(\widehat{\mathfrak{C}})| \leq \mathcal{O}(\frac{1}{\epsilon^2} k \log(cn) \cdot \frac{1}{\delta^2} |\mathcal{F}| \log(\frac{1}{\epsilon^2} k \log(cn)) \log(\frac{1}{\epsilon^2} k \log(cn)|\mathcal{F}|))$, can be computed in time $\mathcal{O}(\frac{1}{\epsilon^2} k \log(cn) \cdot |\mathsf{supp}(\mathfrak{C})| \cdot T)$.*

Clearly, Corollary 5.3 can be used directly after Corollary 5.2 so as to shrink the support of the output as best as possible, as well as before it so as to shrink the support of the input counter as best as possible.

## 5.2 Extension of the Computation of Parsimonious Universal Families to Equip General Membership and Disjointness Procedures

We first present a lemma analogous to Lemma 3.8, to bound the number of sets in the family resulting from the process of universal family sampling that are disjoint from a give complementary balancedly split set (of size $k-p$). Because we do not need to consider general disjointness, the description is in fact slightly simpler than that of Lemma 3.8. The proof is symmetric, but we present it here for the sake of completeness.

**Lemma 5.7.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a $(t, k, p, b)$-splitting function pair. With probability at least $1 - \frac{1}{2c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling has the following property: For every set $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$, we have that $|\{F \in \mathcal{F} : Q \cap F = \emptyset\}| \leq (\dfrac{d \cdot k}{p})^p \cdot (\dfrac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc))^t$.*

*Proof.* Towards the proof of the lemma, we first show that the following claim is correct.

**Claim 5.1.** *With probability at least $1 - \frac{1}{2c}$, for every $i \in \{1, 2, \ldots, t\}$ and $Q \in \binom{U_i}{f(i)-g(i)}$, we have that $|\{F \in \mathcal{F}_i : Q \cap F = \emptyset\}| \leq (\dfrac{d \cdot f(i)}{g(i)})^{g(i)} \cdot \dfrac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)$.*

*Proof.* Denote $E_i = (\dfrac{d \cdot f(i)}{g(i)})^{g(i)} \cdot \dfrac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc)$. By union bound and because $|\binom{U_i}{\leq k-p}| \leq n^k$, it suffices to choose some $i \in \{1, 2, \ldots, t\}$ and $Q \in \binom{U_i}{f(i)-g(i)}$, and prove that with failure probability at most $\frac{1}{2ctn^k}$, we have that $|\{F \in \mathcal{F}_i : Q \cap F = \emptyset\}| \leq E_i$. To this end, observe that each set $F_{i,j} \in \mathcal{F}_i$ is disjoint from $Q$ with probability $(\dfrac{f(i)-g(i)}{d \cdot f(i)})^{f(i)-g(i)}$. Thus, the expected number of sets in $\mathcal{F}_i$ that are disjoint from $Q$ is $E_i$. Because the sets in $\mathcal{F}_i$ are sampled independently from one another, by Chernoff bound (Proposition 2.2), we have that

$$\begin{aligned} Pr(||\mathcal{F}_i[P,Q]| - E_i| > E_i) &\leq 2e^{-\frac{E_i}{2}} \\ &\leq 2e^{-5k \cdot \ln(nc)} = \frac{2}{(nc)^{5k}} \leq \frac{2}{n^4 \cdot n^k \cdot c} \leq \frac{1}{2ct} \end{aligned}$$

Here, the last inequality follows since $n \geq \max(2, t)$. This completes the proof of the claim. □

We now return to the proof of the lemma. Due to Claim 3.1, to prove the lemma it suffices to show that, under the assumption that for every $i \in \{1, 2, \ldots, t\}$ and $Q \in \binom{U_i}{f(i)-g(i)}$, we have that $|\{F \in \mathcal{F}_i : P \subseteq F\}| \leq (\dfrac{d \cdot f(i)}{g(i)})^{g(i)} \cdot \dfrac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)$, it holds that for every set $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$, we have that $|\{F \in \mathcal{F} : Q \cap F = \emptyset\}| \leq (\dfrac{d \cdot k}{p})^p \cdot (\dfrac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc))^t$. Towards the proof of this, consider some set $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$. Then,

$$|\{F \in \mathcal{F} : Q \cap F\}| = \prod_{i=1}^{t} |\{F \in \mathcal{F}_i : Q \cap U_i \cap F = \emptyset\}|.$$

Because $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$, it holds that for every $i \in \{1, \ldots, t\}$, $Q \cap U_i \in \binom{U_i}{f(i)-g(i)}$, and therefore

$|\{F \in \mathcal{F}_i : Q \cap U_i \cap F = \emptyset\}| \leq (\frac{d \cdot f(i)}{g(i)})^{g(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc)$. Thus,

$$\begin{aligned} |\{F \in \mathcal{F} : Q \cap F = \emptyset\}| &\leq \prod_{i=1}^{t} \left( (\frac{d \cdot f(i)}{g(i)})^{g(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc) \right) \\ &\leq \left( \prod_{i=1}^{t} (\frac{d \cdot f(i)}{g(i)})^{g(i)} \right) \cdot \left( \frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc) \right)^t . \end{aligned}$$

Recall that $f : \{1,2,\ldots,t\} \to \{1,2,\ldots,\lceil bk/t \rceil\}$ and $g \leq f$ satisfy $\sum_{i=1}^{t} f(i) = k$ and $\sum_{i=1}^{t} g(i) = p$. Relaxing the supposition $f : \{1,2,\ldots,t\} \to \{1,2,\ldots,\lceil bk/t \rceil\}$ to $f : \{1,2,\ldots,t\} \to \{1,2,\ldots,k\}$, the maximum value of $\prod_{i=1}^{t} (\frac{d \cdot f(i)}{g(i)})^{g(i)}$ is attained when $f(i) = k$ and $g(i) = p$ some $i \in \{1,2,\ldots,t\}$, and $f(i') = g(i') = 0$ for all other $i' \in \{1,2,\ldots,t\} \setminus \{i\}$. Then, the value is $(\frac{d \cdot k}{p})^p$. This completes the proof. □

We now present a procedure symmetric to the procedure MEMBERSHIP from Section 3.2 to handle disjointness rather than membership. Again, because we do not need to consider general disjointness, the description is in fact simpler than that of MEMBERSHIP.

**Definition 5.3 (Disjointness Query Procedure for Parsimonious Universal Family Sampling).** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$. Let $(f,g)$ be a $(t,k,p,b)$-splitting function pair. Let $\mathcal{F} \subseteq 2^U$ be the output family of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling. Then, the procedure* DISJOINTNESS *is defined as follows. Let $\{\mathcal{F}_i\}_{i=1}^{t}$ be the collection of families sampled to construct $\mathcal{F}$ (see Definition 3.9). Given $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$,* MEMBERSHIP *naively computes $\mathcal{F}'_i = \{F_{i,j_i} \in \mathcal{F}_i : Q \cap U_i \cap F_{i,j_i}\}$ by iterating over every set in $\mathcal{F}_i$; then, it outputs $\{F_{1,j_1} \cup F_{2,j_2} \cup \cdots \cup F_{t,j_t} : F_{1,j_1} \in \mathcal{F}'_1, F_{2,j_2} \in \mathcal{F}'_2, \ldots, F_{t,j_t} \in \mathcal{F}'_t\}$, computed using naive enumeration.*

We now assert that our procedure is indeed an efficient membership query procedure as a corollary of Lemma 5.7. The proof is symmetric to the proof of the analogous Corollary 3.2, but we present it here for the sake of completeness.

**Corollary 5.4.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$. Let $(f,g)$ be a $(f,g)$ be a $(t,k,p,b)$-splitting function pair. Let $\mathcal{F} \subseteq 2^U$ be the output family of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling. Then, with probability at least $1 - \frac{1}{2c}$, the procedure* MEMBERSHIP *is a $T$-membership query procedure with respect to $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$ for*

$$T = \left( (d \cdot bk)^{bk/t} + (\frac{d \cdot k}{p})^{k-p} \right) \cdot \left( \frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc) \right)^t .$$

*Proof.* Let $X = (\frac{d \cdot k}{p})^p \cdot (\frac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc))^t$. The claim that DISJOINTNESS is a membership query procedure (i.e., that given $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$, the output is indeed $\{F \in \mathcal{F} : Q \cap F = \emptyset\}$) is immediate from the definition of $\mathcal{F}$. Further, by Lemma 5.7, $\max_{Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}} |\{F \in \mathcal{F} : Q \cap F = \emptyset\}| \leq X$ with probability at least $1 - \frac{1}{2c}$. Now, under the assumption that the aforementioned inequality holds, consider any set $Q \in \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$. Then, the running time of DISJOINTNESS is bounded by

$$
\begin{aligned}
&\mathcal{O}(\sum_{i=1}^{t} s_i + |\{F \in \mathcal{F} : Q \cap F = \emptyset\}|) \\
&= \mathcal{O}(\sum_{i=1}^{t} \frac{(d \cdot f(i))^{f(i)}}{g(i)^{g(i)}(d \cdot f(i) - g(i))^{f(i)-g(i)}} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) \\
&= \mathcal{O}(\sum_{i=1}^{t} (d \cdot f(i))^{f(i)} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) \\
&= \mathcal{O}(\sum_{i=1}^{t} (d \cdot bk/t)^{bk/t} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) \\
&= \mathcal{O}(t \cdot (d \cdot bk)^{bk/t} \cdot \frac{1}{\ln^2(1+\epsilon)} \cdot 10k^3 \cdot \ln(nc) + X) = \mathcal{O}(T).
\end{aligned}
$$

□

From Theorem 3.2 (where $c$ in that theorem is equal to twice the $c$ in the new theorem, so that the failure probability is divided by 2, and therefore the constants 10 and 20 are changed to 20 and 30, respectively, here) and Corollary 5.4, we derive the following theorem.

**Theorem 5.2.** *Let $t, k, p, b \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. With probability at least $1 - \frac{1}{c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, b, f, g, \epsilon, c, d)$-universal family sampling, computed in time $\mathcal{O}(|\mathcal{F}|n)$, satisfies all of the following conditions.*

1. $|\mathcal{F}| \leq \dfrac{(dk)^k}{p^p(dk-p)^{k-p}} \cdot (\dfrac{1}{\ln^2(1+\epsilon)} \cdot 20k^3 \cdot \ln(nc))^t$.

2. *$\mathcal{F}$ is an $\epsilon$-parsimonious $(n, p, k-p)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}, \mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g})$, whose correction factor is upper bounded by* $(\dfrac{1}{\ln(1+\epsilon)})^2 \cdot 20k^3 \cdot \ln(nc))^t$.

3. *With respect to $\mathcal{F}$ and any $g'$ be such that $(f, g')$ is a $(t, k, p', b)$-splitting function pair (for some $p' \leq p$) where $g' \leq g$,* MEMBERSHIP *is a $T^{p'}_{\mathrm{con}}$-membership query procedure with respect to $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g'}$ for*
$$T^{p'}_{\mathrm{con}} = \left((d \cdot bk)^{bk/t} + (\frac{dk}{dk-p})^{k-p}(\frac{dk}{p})^{p-p'}\right) \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 30k^3 \cdot \ln(nc)\right)^t.$$

4. *With respect to $\mathcal{F}$,* DISJOINTNESS *is a $T_{\mathrm{dis}}$-disjointness query procedure for*
$$T_{\mathrm{dis}} = \left((d \cdot bk)^{bk/t} + (\frac{dk}{p})^p\right) \cdot \left(\frac{1}{\ln^2(1+\epsilon)} \cdot 30k^3 \cdot \ln(nc)\right)^t.$$

We will be specifically interested in the case where $t = \lceil\sqrt{k}\rceil$, $b = 2$, $\epsilon = \frac{\ln\frac{3}{2}}{5(2k)^2}$, $c \geq n$, and $d = \mathcal{O}(1)$ (but not 1.447 as before), thus we explicitly give the following corollary.

**Corollary 5.5.** *Let $t, k, p \in \mathbb{N}$ with $p \leq k$, $0 < \epsilon < 1$ and $c, d \geq 1$. Let $\overline{\mathbf{U}} = (U_1, U_2, \ldots, U_t)$ be a partitioned universe with $U = \bigcup_{i=1}^{t} U_i$ of size $n$, and let $(f, g)$ be a splitting function pair. With probability at least $1 - \frac{1}{c}$, the output family $\mathcal{F} \subseteq 2^U$ of $(\overline{\mathbf{U}}, 2, f, g, \frac{\ln\frac{3}{2}}{5(2k)^2}, c, d)$-universal family sampling, computed in time $\mathcal{O}(|\mathcal{F}|n)$, satisfies all of the following conditions.*

1. $|\mathcal{F}| \leq \dfrac{(dk)^k}{p^p(dk-p)^{k-p}} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}} c$.

2. $\mathcal{F}$ *is an* $\frac{\ln\frac{3}{2}}{5(2k)^2}$*-parsimonious* $(n,p,k-p)$*-universal family with respect to* $(\mathcal{P}^{\mathrm{BAL}}_{\mathbf{U},f,g}, \mathcal{Q}^{\mathrm{CBAL}}_{\mathbf{U},f,g})$*, whose correction factor is upper bounded by* $2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}} c$.

3. *With respect to* $\mathcal{F}$ *and any* $g'$ *be such that* $(f,g')$ *is a* $(t,k,p',b)$*-splitting function pair (for some* $p' \leq p$*) where* $g' \leq g$*,* MEMBERSHIP *is a* $T^{p'}_{\mathrm{con}}$*-membership query procedure for*
$$T^{p'}_{\mathrm{con}} = (\frac{dk}{dk-p})^{k-p}(\frac{dk}{p})^{p-p'} \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}} c.$$

4. *With respect to* $\mathcal{F}$*,* DISJOINTNESS *is a* $T_{\mathrm{dis}}$*-disjointness query procedure for*
$$T_{\mathrm{dis}} = (\frac{dk}{p})^p \cdot 2^{\mathcal{O}(\sqrt{k}\log k)} \cdot \log^{\sqrt{k}} c.$$

Lastly, we give a corollary for Lemma 3.10 with slightly different parameters than Corollary 3.4 and where correctness of expectation is not assumed. Specifically, here we substitute $\alpha = \beta = \frac{\epsilon}{4}$ (where $b = 2$ and $t = \sqrt{k}$ as before). Then, since $0 < \epsilon < 1$, $(1-\alpha)(1-\beta) = (1-\frac{\epsilon}{4})^2 = 1 - \frac{\epsilon}{2} + \frac{\epsilon^2}{16} \geq 1 - \epsilon$ and $(1+\alpha)(1+\beta) = (1+\frac{\epsilon}{4})^2 = 1 + \frac{\epsilon}{2} + \frac{\epsilon^2}{16} \leq 1 + \epsilon$, and we have the following corollary.

**Corollary 5.6.** *Let* $\Pi$ *be a splittable problem such that the number of solutions of the split version of* $\Pi$ *can be approximately counted with multiplicative error* $(1 \pm \frac{\epsilon}{4})$ *in time* $T$ *and with success probability at least* $1 - \frac{1}{c'}$*. Then, for any* $c \in \mathbb{N}$ *such that* $4k(4\sqrt{k})^{\sqrt{k}} \ln(nc) \cdot \frac{1}{c'} \leq \frac{1}{c}$*, the number of solutions of* $\Pi$ *can be approximately counted with multiplicative error between* $\frac{1}{4}$ *and* $2\frac{1}{4}$ *in time* $\mathcal{O}((2^{\mathcal{O}(\sqrt{k}\log k)} \cdot T + n) \cdot \frac{1}{\epsilon^2} k \ln(nc))$ *and with success probability at least* $1 - \frac{1}{c}$*.*

## 5.3 An Algorithm for #Multilinear Monomial Detection on General Circuits

We now show how the algorithm in Section 4.1, with slight modification, solves #Multilinear Monomial Detection on general circuits in the desired time.

**Theorem 5.3.** *For any* $0 < \epsilon < 1$ *, the* #Multilinear Monomial Detection *problem can be approximated with factor* $(1 \pm \epsilon)$ *and success probability at least* $\frac{9}{10}$ *in time* $\mathcal{O}((3.841^k + s(C)^{o(1)}) \cdot \frac{1}{\epsilon^6} \cdot s(C))$*.*

As in Section 4.1, but now using Corollary 5.6 rather than Corollary 3.5 (to avoid the requirement to prove that expectation is correct, which we cannot do), the correctness follows from Lemma 5.8 below.

**Lemma 5.8.** *For any* $0 < \epsilon < 1$*, the splittable version of* #Multilinear Monomial Detection *problem can be computed exactly in expectation and approximated with factor* $(1 \pm \frac{\epsilon}{4})$ *and success probability at least* $1 - \frac{1}{(1000\sqrt{k})^{\sqrt{k}} \ln s(C)}$ *in time* $3.8104^k \cdot 2^{o(k)} \cdot \frac{1}{\epsilon^4} \cdot s(C) \cdot \log^{\mathcal{O}(\sqrt{k})} s(C)$*.*

*Proof.* The algorithm is the same as the algorithm in the proof of Lemma 4.1 with the following minor differences. First, we compute the universal families $\mathcal{F}_{p,g}$ as before but with smaller $\epsilon' = \frac{\ln\frac{3}{2}}{5(2k)^2}$ and $c = (1000\sqrt{k})^{\sqrt{k}} \ln(s(C)\frac{1}{\epsilon}) \cdot s(C) k^{100(\sqrt{k})}$. Then, we compute additional universal families by using Corollary 5.5 (where the constants $d_{k,p,p'}, \widehat{d}_{k,p,p'} = \mathcal{O}(1)$ specified ahead will be fixed later) as follows. For all $p \in \{1,2,\ldots,k\}, p_1 \in \{1,2,\ldots,p\}$ and $g$ such that $(f,g)$

is a $(\sqrt{k},k,p,2)$-splitting pair, we compute with success probability at least $1-\frac{1}{c}$, a family $\mathcal{F}_{p,g,p_1}\subseteq 2^X$ of $(\overline{\mathbf{U}},2,f,g,\frac{\ln\frac{3}{2}}{5(2k)^2},c,d_{k,p,p_1})$-universal family sampling, where $c$ is as defined earlier in this proof, that satisfies all of the following conditions.

1. $|\mathcal{F}_{p,g,p_1}|\leq\dfrac{(d_{k,p,p_1}k)^k}{p^p(d_{k,p,p'}k-p)^{k-p}}\cdot 2^{\mathcal{O}(\sqrt{k}\log k)}\cdot\log^{\sqrt{k}}s(C)$.
2. $\mathcal{F}_{p,g,p_1}$ is an $\frac{\ln\frac{3}{2}}{5(2k)^2}$-parsimonious $(n,p,k-p)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g},\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g})$.
3. With respect to $\mathcal{F}_{p,g,p_1}$ and any $g'$ be such that $(f,g')$ is a $(\sqrt{k},k,p',2)$-splitting function pair (for some $p'\leq p$) where $g'\leq g$, MEMBERSHIP is a $T^{p'}_{\mathrm{con}}$-membership query procedure for
$$T^{p'}_{\mathrm{con}}=(\frac{d_{k,p,p_1}k}{d_{k,p,p_1}k-p})^{k-p}(\frac{d_{k,p,p_1}k}{p})^{p-p'}\cdot 2^{\mathcal{O}(\sqrt{k}\log k)}\cdot\log^{\sqrt{k}}s(C).$$

Additionally, for all $p\in\{1,2,\ldots,k\}$, $g$ such that $(f,g)$ is a $(\sqrt{k},k,p,2)$-splitting pair, $p_1\in\{1,2,\ldots,p\}$ and $g_1\leq g$ such that $(f,g_1)$ is a $(\sqrt{k},p,p_1,2)$-splitting pair, we compute with success probability at least $1-\frac{1}{c}$, a family $\mathcal{H}_{p,g,p_1,g_1}\subseteq 2^X$ of $(\overline{\mathbf{U}},2,g,g_1,\frac{\ln\frac{3}{2}}{5(2k)^2},c,\widehat{d}_{k,p,p_1})$-universal family sampling, where $c$ is as defined earlier in this proof, that satisfies all of the following conditions.

1. $|\mathcal{H}_{p,g,p_1,g_1}|\leq\dfrac{(\widehat{d}_{k,p,p_1}k)^k}{p^p(\widehat{d}_{k,p,p_1}k-p)^{k-p}}\cdot 2^{\mathcal{O}(\sqrt{k}\log k)}\cdot\log^{\sqrt{k}}s(C)$.
2. $\mathcal{H}_{p,g,p_1,g_1}$ is an $\frac{1}{2}$-parsimonious $(n,p,k-p)$-universal family with respect to $(\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g},\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g})$, whose correction factor is upper bounded by $2^{\mathcal{O}(\sqrt{k}\log k)}\cdot\log^{\sqrt{k}}s(C)$.
3. With respect to $\mathcal{H}_{p,g,p_1,g_1}$, DISJOINTNESS is a $T_{\mathrm{dis}}$-disjointness query procedure for
$$T_{\mathrm{dis}}=(\frac{\widehat{d}_{k,p,p_1}k}{p})^p\cdot 2^{\mathcal{O}(\sqrt{k}\log k)}\cdot\log^{\sqrt{k}}s(C).$$

The order of computation, the computation for leaf and addition nodes, and the computation of the output, all remain unchanged. Notice that we do not compute the table $N$, because this is, in the worst case, useless. With respect to multiplication nodes, we do not compute $\mathfrak{C}_{(p_1,p_2,g_1,g_2)}$ (but instead a representative $\widehat{\mathfrak{C}}_{(p_1,p_2,g_1,g_2)}$) and thus neither $\mathfrak{C}$ (but instead a representative $\mathfrak{C}^\star$) explicitly as before, and invoke Corollary 5.2 before Corollary 5.3 along the way. For the sake of clarity, we present the full computation in this case below.

**Multiplication Node.** Let $v_1$ and $v_2$ be the two outgoing neighbors of $v$. Let $\mathcal{I}$ be the set consisting of all quadruples $(p_1,p_2,g_1,g_2)$ such that $p_1,p_2\in\mathbb{N},p_1+p_2=p,(f,g_1)$ and $(f,g_2)$ are $(\sqrt{k},k,p_1,2)$ and $(\sqrt{k},k,p_2,2)$-splitting function pairs, respectively, and for every $i\in\{1,2,\ldots,\sqrt{k}\}$, it holds that $g_1(i)+g_2(i)=g(i)$. First, for every quadruple $(p_1,p_2,g_1,g_2)\in\mathcal{I}$, let $\widehat{\mathfrak{C}}_{(p_1,p_2,g_1,g_2)}:\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}\rightarrow\mathbb{N}_0$ be the counter obtained by invoking Corollary 5.2 with respect to $X$ as the universe, $\epsilon'$ (denoted by $\epsilon$ in the statement but here denoted by $\epsilon'$ to avoid overloading notation) being $\frac{\ln(1+\epsilon)}{(2k)^2}$, $\delta$ being $\frac{\ln\frac{3}{2}}{(2k)^2}$, $p_1,p_2,q=k-p$, $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$ as $\mathcal{P}$ and $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$ as $\mathcal{Q}$, $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},g,g_1}$ as $\mathcal{P}_1$, $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},g,g_2}$ as $\mathcal{P}_2$, the counter stored at $M[v_1,p_1,g_1]$ as $\mathfrak{C}_1$, the counter stored at $M[v_2,p_2,g_2]$ as $\mathfrak{C}_2$, $\mathcal{F}_{p,g,p_1}$ as $\mathcal{F}$, $\mathcal{H}_{p,g,p_1,g_1}$ as $\mathcal{H}$ and $c$ as defined earlier in this proof. Second, the counter $\mathfrak{C}^\star:\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}:\rightarrow\mathbb{N}_0$ is defined as follows. For every set $P\in\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$, define $\mathfrak{C}^\star(P) = \sum_{(p_1,p_2,g_1,g_2)\in\mathcal{I}} \widehat{\mathfrak{C}}_{(p_1,p_2,g_1,g_2)}(P)$. Third, the entry $M[v,p,g]$ stores the counter $\widehat{\mathfrak{C}} : \mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g} \rightarrow \mathbb{N}_0$ obtained by applying Corollary 5.3 with respect to $X$ as the universe, $\epsilon'$ (denoted by $\epsilon$ in the statement but here denoted by $\epsilon'$ to avoid overloading notation) being $\frac{\ln(1+\epsilon)}{(2k)^2}$, $\delta = \frac{\ln\frac{3}{2}}{(2k)^2}$, $p_1, p_2, q = k-p$, $\mathcal{P}^{\mathrm{BAL}}_{\overline{\mathbf{U}},f,g}$ as $\mathcal{P}$ and $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$ as $\mathcal{Q}$, $\mathfrak{C}^\star$ as $\mathfrak{C}$, $\mathcal{F}_{p,g}$ as $\mathcal{F}$, and $c$ as defined earlier in this proof.

**Time Complexity and Correctness.** Time complexity analysis and correctness follow the same lines as in the proof of Lemma 4.1, where the obtained expressions in the time complexity analysis are (essentially) the same as in [FLPS17] and hence bounded in the exact same way (which results in the time bound stated in the lemma, and also where the constants analogous to $d_{k,p,p_1}$ and $\widehat{d}_{k,p,p_1}$ are fixed). Thus, repetition of these details are omitted. For the sake of clarity, we only state the main claim regarding correctness (where $\mathfrak{B}_{v,p,g}$ is defined as in the proof of Lemma 4.1), which is proved by induction similarly to the proof of Claim 4.1.

**Claim 5.2.** *For every node $v$ of $C$, $p \in \{1,2,\ldots,k\}$, and $g$ such that $(f,g)$ is a $(\sqrt{k},k,p,2)$-splitting pair, the following holds: Under the assumption that all calls to the algorithms in Corollaries 5.5, 5.2 and 5.3 are successful, the counter $\widehat{\mathfrak{C}}$ stored at $M[v,p,g]$ $((1-\frac{\ln(1+\epsilon)}{(2k)^2})^{(2p)^2}, (1+\frac{\ln(1+\epsilon)}{(2k)^2})^{(2p)^2}, k-p)$-represents $\mathfrak{B}_{v,p,g}$ with respect to $\mathcal{Q}^{\mathrm{CBAL}}_{\overline{\mathbf{U}},f,g}$.*

□

# 6 Conclusion and Open Problems

In this paper, we presented a general tool to design FPT-approximation schemes for counting problems. Specifically, we introduced the notion of a representative function where our main contribution is a novel sampling procedure to compute representative functions of small support efficiently. Along the way, we developed a data structure to efficiently query membership and disjointness in approximately universal families, which is of independent interest. We have demonstrated the wide applicability of our tool by developing a $\mathcal{O}((2.619^k + |I|^{o(1)}) \cdot \frac{1}{\epsilon^2} \cdot |I|)$-time algorithm for #Multilinear Monomial Detection on skewed circuits, #$k$-Path, and several other problems as well (including #$q$-Set $p$-Packing with $k = qp$, #$q$-Dimensional $p$-Matching with $k = (q-1)p$, # Graph Motif, and #Subgraph Isomorphism for pattern graphs of constant treewidth). Additionally, we developed a $\mathcal{O}((3.841^k + |I|^{o(1)}) \cdot \frac{1}{\epsilon^6} \cdot |I|)$-time algorithm for #Multilinear Monomial Detection on general (monotone) circuits.

We conclude our paper with a few open problems.

- Does the #$k$-Path problem admit an FPT-approximation scheme with running time $2^k(\frac{1}{\epsilon})^{\mathcal{O}(1)}n^{\mathcal{O}(1)}$?
- Does the #Multilinear Monomial Detection problem admit an FPT-approximation scheme with running time substantially better than $3.841^k(\frac{1}{\epsilon})^{\mathcal{O}(1)}n^{\mathcal{O}(1)}$? In particular, can the time bound $2.619^k(\frac{1}{\epsilon})^{\mathcal{O}(1)}n^{\mathcal{O}(1)}$ given for #$k$-Path be matched?
- Can our result for the #Multilinear Monomial Detection problem be extended to non-monotone arithmetic circuits (where subtraction is allowed)?
- Are there relations between techniques based on exterior algebra, Hadamard product, warring rank and representative functions?
- Can we compute representative functions efficiently with respect to linear matroids rather than only set systems (i.e., uniform matroids)? For more information on representation with respect to a matroid, we refer to [FLPS16].

- We remark that results on bounded skewness by themselves may be of interest in this context. Can we derive a general theorem about the problems that admits them? What can be said in this context on VP-circuits and homomorphism polynomials?

**Acknowledgements.** We thank one of the reviewers of a previous version of the paper for telling us about monotone arithmetic circuits.

## References

[ACDM19] Vikraman Arvind, Abhranil Chatterjee, Rajit Datta, and Partha Mukhopadhyay. Fast exact algorithms using hadamard product of polynomials. In Arkadev Chattopadhyay and Paul Gastin, editors, *39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2019, December 11-13, 2019, Bombay, India*, volume 150 of *LIPIcs*, pages 9:1–9:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. 5, 10

[ADH+08] Noga Alon, Phuong Dao, Iman Hajirasouliha, Fereydoun Hormozdiari, and Süleyman Cenk Sahinalp. Biomolecular network motif counting and discovery by color coding. In *Proceedings 16th International Conference on Intelligent Systems for Molecular Biology (ISMB), Toronto, Canada, July 19-23, 2008*, pages 241–249, 2008. 1, 2, 3, 10

[AG09] Noga Alon and Shai Gutner. Balanced hashing, color coding and approximate counting. In *Parameterized and Exact Computation, 4th International Workshop, IWPEC 2009, Copenhagen, Denmark, September 10-11, 2009, Revised Selected Papers*, pages 1–16, 2009. 2, 3, 10

[AG10] Noga Alon and Shai Gutner. Balanced families of perfect hash functions and their applications. *ACM Trans. Algorithms*, 6(3):54:1–54:12, 2010. 1, 3, 10

[AR02] Vikraman Arvind and Venkatesh Raman. Approximation algorithms for some parameterized counting problems. In *Algorithms and Computation, 13th International Symposium, ISAAC 2002 Vancouver, BC, Canada, November 21-23, 2002, Proceedings*, pages 453–464, 2002. 1, 2

[AYZ95] Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *J. ACM*, 42(4):844–856, 1995. 2

[BDH18] Cornelius Brand, Holger Dell, and Thore Husfeldt. Extensor-coding. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 151–164, 2018. 1, 2, 3, 5

[BHKK09] Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Counting paths and packings in halves. In *Algorithms - ESA 2009, 17th Annual European Symposium, Copenhagen, Denmark, September 7-9, 2009. Proceedings*, pages 578–586, 2009. 10

[BHKK17] Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Narrow sieves for parameterized paths and packings. *J. Comput. Syst. Sci.*, 87:119–139, 2017. 2

[Bjö14] Andreas Björklund. Determinant sums for undirected hamiltonicity. *SIAM J. Comput.*, 43(1):280–299, 2014. 2

[BKK17] Andreas Björklund, Petteri Kaski, and Lukasz Kowalik. Counting thin subgraphs via packings faster than meet-in-the-middle time. *ACM Trans. Algorithms*, 13(4):48:1–48:26, 2017. 10

[BKKZ17] Andreas Björklund, Vikram Kamat, Lukasz Kowalik, and Meirav Zehavi. Spotting trees with few leaves. *SIAM J. Discrete Math.*, 31(2):687–713, 2017. 2

[BLSZ19] Andreas Björklund, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. Approximate counting of k-paths: Deterministic and in polynomial space. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, volume 132 of *LIPIcs*, pages 24:1–24:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. 1, 2, 3, 10

[Bol65] B. Bollobás. On generalized graphs. *Acta Math. Acad. Sci. Hungar*, 16:447–452, 1965. 2

[BP20] Cornelius Brand and Kevin Pratt. An algorithmic method of partial derivatives. *CoRR*, abs/2005.05143, 2020. 5

[Bra19] Cornelius Brand. Patching colors with tensors. In Michael A. Bender, Ola Svensson, and Grzegorz Herman, editors, *27th Annual European Symposium on Algorithms, ESA 2019, September 9-11, 2019, Munich/Garching, Germany*, volume 144 of *LIPIcs*, pages 25:1–25:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. 1

[CDM17] Radu Curticapean, Holger Dell, and Dániel Marx. Homomorphisms are a good basis for counting small subgraphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, pages 210–223, New York, NY, USA, 2017. ACM. 1, 10

[CFK$^+$15] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. 1, 2

[CKL$^+$09a] J. Chen, J. Kneis, S. Lu, D. Mölle, S. Richter, P. Rossmanith, S. Sze, and F. Zhang. Randomized divide-and-conquer: Improved path, matching, and packing algorithms. *SIAM Journal on Computing*, 38(6):2526–2547, 2009. 2

[CKL$^+$09b] Jianer Chen, Joachim Kneis, Songjian Lu, Daniel Molle, Stefan Richter, Peter Rossmanith, Sing-Hoi Sze, and Fenghui Zhang. Randomized divide-and-conquer: Improved path, matching, and packing algorithms. *SIAM Journal on Computing*, 38(6):2526—2547, 2009. 2

[CM14] Radu Curticapean and Dániel Marx. Complexity of counting subgraphs: Only the boundedness of the vertex-cover number counts. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 130–139, 2014. 1, 10

[Cur13] Radu Curticapean. Counting matchings of size $k$ is W[1]-hard. In Fedor V. Fomin, Rusins Freivalds, Marta Z. Kwiatkowska, and David Peleg, editors, *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part I*, volume 7965 of *Lecture Notes in Computer Science*, pages 352–363. Springer, 2013. 1

[Cur18] Radu Curticapean. Counting problems in parameterized complexity. In Christophe Paul and Michal Pilipczuk, editors, *13th International Symposium on Parameterized and Exact Computation, IPEC 2018, August 20-24, 2018, Helsinki, Finland*, volume 115 of *LIPIcs*, pages 1:1–1:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. 1

[CX15] Radu Curticapean and Mingji Xia. Parameterizing the permanent: Genus, apices, minors, evaluation mod 2k. In Venkatesan Guruswami, editor, *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 994–1009. IEEE Computer Society, 2015. 1

[DF13] Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. 1

[DLM20] Holger Dell, John Lapinskas, and Kitty Meeks. Approximately counting and sampling small witnesses using a colourful decision oracle. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 2201–2211. SIAM, 2020. 1, 10

[DSG$^+$08] Banu Dost, Tomer Shlomi, Nitin Gupta, Eytan Ruppin, Vineet Bafna, and Roded Sharan. Qnet: A tool for querying protein interaction networks. *Journal of Computational Biology*, 15(7):913–925, 2008. 2

[FG04] Jörg Flum and Martin Grohe. The parameterized complexity of counting problems. *SIAM J. Comput.*, 33(4):892–922, 2004. 1, 10

[FG06] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006. 1

[FGPS19] Fedor V. Fomin, Petr A. Golovach, Fahad Panolan, and Saket Saurabh. Editing to connected f-degree graph. *SIAM J. Discrete Math.*, 33(2):795–836, 2019. 2

[FLPS16] Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *J. ACM*, 63(4):29:1–29:60, 2016. 2, 34, 39, 55

[FLPS17] Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Representative families of product families. *ACM Trans. Algorithms*, 13(3):36:1–36:29, 2017. 2, 55

[FLSZ19] Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization: Theory of Parameterized Preprocessing*. Cambridge University Press, 2019. 2

[HWZ08] Falk Hüffner, Sebastian Wernicke, and Thomas Zichner. Algorithm engineering for color-coding with applications to signaling pathway detection. *Algorithmica*, 52(2):114–132, 2008. 2

[Kou08] Ioannis Koutis. Faster algebraic algorithms for path and packing problems. In *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I: Tack A: Algorithms, Automata, Complexity, and Games*, pages 575–586, 2008. 1, 4

[KS17] Stefan Kratsch and Manuel Sorge. On kernelization and approximation for the vector connectivity problem. *Algorithmica*, 79(1):96–138, 2017. 2

[KW12] Stefan Kratsch and Magnus Wahlström. Representative sets and irrelevant vertices: New tools for kernelization. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 450–459. IEEE Computer Society, 2012. 2

[KW16a] Ioannis Koutis and Ryan Williams. Algebraic fingerprints for faster algorithms. *Commun. ACM*, 59(1):98–105, 2016. 3, 5, 39

[KW16b] Ioannis Koutis and Ryan Williams. LIMITS and applications of group algebras for parameterized problems. *ACM Trans. Algorithms*, 12(3):31:1–31:18, 2016. 1, 2, 4, 39

[Mar06] Dániel Marx. Parameterized coloring problems on chordal graphs. *Theor. Comput. Sci.*, 351(3):407–424, 2006. 2

[Mar09] Dániel Marx. A parameterized view on matroid optimization problems. *Theor. Comput. Sci.*, 410(44):4471–4479, 2009. 1, 2

[Mon85] B. Monien. How to find long paths efficiently. In *Analysis and design of algorithms for combinatorial problems (Udine, 1982)*, volume 109 of *North-Holland Math. Stud.*, pages 239–254. North-Holland, Amsterdam, 1985. 1, 2

[MSOI$^+$02] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002. 3

[Pra19] Kevin Pratt. Waring rank, parameterized and exact algorithms. In David Zuckerman, editor, *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, pages 806–823. IEEE Computer Society, 2019. 2, 3, 4

[RW20] Marc Roth and Philip Wellnitz. Counting and finding homomorphisms is universal for parameterized complexity theory. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 2161–2180. SIAM, 2020. 1

[SI06] Roded Sharan and Trey Ideker. Modeling cellular machinery through biological network comparison. nat. biotechnol. 24, 427-433. *Nature biotechnology*, 24:427–33, 05 2006. 2

[SIKS06] Jacob Scott, Trey Ideker, Richard M. Karp, and Roded Sharan. Efficient algorithms for detecting signaling pathways in protein interaction networks. *Journal of Computational Biology*, 13(2):133–144, 2006. 2

[SSRS06] Tomer Shlomi, Daniel Segal, Eytan Ruppin, and Roded Sharan. Qpath: a method for querying pathways in a protein-protein interaction network. *BMC Bioinformatics*, 7:199, 2006. 2

[SZ16] Hadas Shachnai and Meirav Zehavi. Representative families: A unified tradeoff-based approach. *J. Comput. Syst. Sci.*, 82(3):488–502, 2016. 2, 34

[Tsu19] Dekel Tsur. Faster deterministic parameterized algorithm for $k$-path. *Theor. Comput. Sci.*, 790:96–104, 2019. 2

[Val79] Leslie G. Valiant. The complexity of computing the permanent. *Theor. Comput. Sci.*, 8:189–201, 1979. 1

[Wil09] Ryan Williams. Finding paths of length $k$ in $O^*(2^k)$) time. *Inf. Process. Lett.*, 109(6):315–318, 2009. 2, 4, 5

[WW13] Virginia Vassilevska Williams and Ryan Williams. Finding, minimizing, and counting weighted subgraphs. *SIAM J. Comput.*, 42(3):831–854, 2013. 10

[Zeh15] Meirav Zehavi. Mixing color coding-related techniques. In *Algorithms - ESA 2015 - 23rd Annual European Symposium, Patras, Greece, September 14-16, 2015, Proceedings*, pages 1037–1049, 2015. 2