Electronic Colloquium on Computational Complexity, Report No. 15 (2026)

A Theory for Probabilistic Polynomial-Time Reasoning

Lijie Chen* Jiatu Li† Igor C. Oliveira‡ Ryan Williams§

February 9, 2026

Abstract

In this work, we propose a new bounded arithmetic theory, denoted APX_1, designed to formalize a broad class of probabilistic arguments commonly used in theoretical computer science. Under plausible assumptions, APX_1 is strictly weaker than previously proposed frameworks, such as the theory APC_1 introduced in the seminal work of Jeřábek (2007). From a computational standpoint, APX_1 is closely tied to approximate counting and to the central question in derandomization, the prBPP versus prP problem, whereas APC_1 is linked to the dual weak pigeonhole principle and to the existence of Boolean functions with exponential circuit complexity.

A key motivation for introducing APX_1 is that its weaker axioms expose finer proof-theoretic structure, making it a natural setting for several lines of research, including unprovability of complexity conjectures and reverse mathematics of randomized lower bounds. In particular, the framework we develop for APX_1 enables the formulation of precise questions concerning the provability of $\mathsf{prBPP} = \mathsf{prP}$ in *deterministic* feasible mathematics. Since the (un)provability of P versus NP in bounded arithmetic has long served as a central theme in the field, we expect this line of investigation to be of particular interest.

Our technical contributions include developing a comprehensive foundation for probabilistic reasoning from weaker axioms, formalizing non-trivial results from theoretical computer science in APX_1, and establishing a tailored witnessing theorem for its provably total TFNP problems. As a byproduct of our analysis of the minimal proof-theoretic strength required to formalize statements arising in theoretical computer science, we resolve an open problem regarding the provability of AC^0 lower bounds in PV_1, which was considered in earlier works by Razborov (1995), Krajíček (1995), and Müller and Pich (2020).

*University of California at Berkeley. Email: lijiechen@berkeley.edu

†Massachusetts Institute of Technology. Email: jiatuli@mit.edu. This work was supported by National Science Foundation under Grant No. CCF 2420092.

‡University of Warwick. Email: igor.oliveira@warwick.ac.uk. This work was supported in part by the UKRI Frontier Research Guarantee Grant EP/Y007999/1 and the Centre for Discrete Mathematics and its Applications (DIMAP) at the University of Warwick.

§Massachusetts Institute of Technology. Email: rrw@mit.edu. Parts of this work were completed while the author was visiting the Institute for Advanced Study, Princeton, NJ. This material is based upon work supported by the National Science Foundation under grants DMS-2424441 (at IAS) and CCF-2420092 (at MIT).

ISSN 1433-8092

Contents

1 Introduction

1.1 Overview

Bounded arithmetic extends traditional complexity theory by capturing not only the computational resources (e.g., running time or circuit size) required by algorithms, but also the complexity of proving their correctness. By integrating computational and proof complexity within a unified framework, it opens new angles on foundational questions in theoretical computer science. The area has a long history (see [HP93, Kra95, CN10, Kra19] and references therein) and has seen renewed momentum through new formalizations [BKKK20, Gay24, Kha24, AAdRK25]; unprovability results [PS21, LO23, ABM23, CLO25, CRT25, Tha25]; connections to TFNP [LLR24], complex analysis [Jeř23], reverse mathematics [CLO24, AT25], complexity lower bounds [GC25, CKK⁺25], and propositional proof complexity [Kra25]; and applications in cryptography [JJ22, JKLV24, JKLM25, JJMP25], among other developments. We refer to [Bus97, Oli25] for background and for connections to algorithms and complexity theory.

Two central and extensively studied theories are Cook's PV_1 [Coo75, KPT91] (see also [Kra19, Li25]) and Jeřábek's APC_1 [Jeř04, Jeř05, Jeř07a]. The theory PV_1 formalizes polynomial-time reasoning and captures many classical results in algorithms and complexity. Since it is unclear whether randomized algorithms can, in general, be derandomized, PV_1 is not well-suited for reasoning about probabilities or analyzing randomized algorithms. The theory APC_1 extends PV_1 by adding the dual weak pigeonhole principle $\mathsf{dWPHP}(\mathsf{PV})$, yielding a convenient framework for reasoning about probabilities and randomized constructions. In particular, APC_1 (and its mild extensions) is sufficient to formalize several nontrivial results, including the correctness of randomized algorithms for graph problems [LC11], polynomial identity testing [AT25], and circuit lower bounds [MP20]. However, the axioms of APC_1 may be stronger than necessary: many results of interest could plausibly be provable in a weaker theory closer to PV_1.

There are concrete reasons to expect APC_1 to exceed the minimal strength required for probabilistic polynomial-time reasoning. On the one hand, APC_1 is tied to $\mathsf{dWPHP}(\mathsf{PV})$ and to the *existence* of functions of exponential circuit complexity; from a computational perspective, the *explicit construction* of such functions (i.e., circuit lower bounds) is a widely used derandomization assumption that may be stronger than the derandomization of prBPP (see, e.g., [For01, Gol11, CT23]). On the other hand, even if $\mathsf{prBPP} = \mathsf{prP}$ with a "feasible" proof, APC_1 need not collapse to PV_1; indeed, under plausible cryptographic assumptions, APC_1 is strictly stronger than PV_1 [ILW23].

The search for a weaker theory that still supports the broad class of probabilistic arguments used across theoretical computer science is motivated by several considerations:

- *Unprovability of complexity-theoretic conjectures.* A central objective in this area is to identify frameworks that both formalize existing tools in complexity theory and remain amenable to unprovability results. APC_1 is likely strictly stronger than PV_1 by [ILW23], and its witnessing functions cannot in general be made deterministic even if $\mathsf{prBPP} = \mathsf{prP}$, which complicate unprovability arguments and pose significant challenges (see, e.g., [LO23, CKKO21]). In particular, the introduction of the strong principle $\mathsf{dWPHP}(\mathsf{PV})$ is the main obstacle to extending unprovability of complexity lower bounds in PV_1 [PS21] to APC_1 [LO23].

- *Bounded reverse mathematics with probabilistic reasoning.* Following recent developments such as [CLO24, AT25] (see also [CN10] for related background), one can hope to pursue a systematic reverse mathematics of algorithms and complexity theory that classifies "probabilistic proofs" by the axioms they use. Similarly, it suggests the possibility of classifying randomized algorithms by the complexity of their *correctness proofs*, supplementing the standard classification via space (see, e.g., [Nis92]) or circuit complexity (see, e.g., [AW89]). This perspective is potentially insightful for derandomization, namely, derandomization based on the proof complexity of correctness proofs. Because $\mathsf{dWPHP}(\mathsf{PV})$ is itself strong, APC_1 is an overly powerful base theory for fine-grained correspondences weaker than $\mathsf{dWPHP}(\mathsf{PV})$.

- *Correctness proofs in cryptography.* Jain and Jin [JJ22] and subsequent papers [JKLV24, JKLM25, JJMP25, MDS25] explore PV_1 and its connection to propositional proofs to help construct $i\mathcal{O}$ and other cryptographic primitives, highlighting that the logical complexity of proving certain statements can play an important role in cryptographic systems and their efficiency. In particular, [JJMP25, JKLM25] rely heavily on cryptographic primitives with PV_1 proofs of *correctness*. However, existing work typically considers "perfect correctness" because PV_1 cannot natively talk about approximate counting and randomness, whereas APC_1 seems both too strong and inconvenient for this purpose.

- *Feasible provability of probabilistic statements.* It is natural to formulate precise, feasible notions of the provability of $\mathsf{prBPP} = \mathsf{prP}$ and related questions. Yet even formulating $\mathsf{prBPP} = \mathsf{prP}$ feasibly is nontrivial, as it seems to require defining probabilistic computation within the theory in the first place. Given that the (un)provability of $\mathsf{P} = \mathsf{NP}$ in PV_1 has long been central to bounded arithmetic (see, e.g., [CK07, Oli25]), this direction holds significant potential for advancing the study of the interplay between randomized computations and mathematical proofs.

These considerations point to a common objective: designing a *minimal theory* for reasoning about probabilities and randomized constructions in feasible mathematics.

Summary of contributions. We propose a theory corresponding to "probabilistic polynomial-time reasoning" in a strong sense. Our main conceptual and technical contributions are:

1. **Theory APX_1 and its relative strength.** We introduce APX_1, establish its basic properties, and develop core probabilistic tools. The theory extends PV_1 and is contained in APC_1, in the sense that all of its consequences in the language of APC_1 are also provable in APC_1. Moreover, under plausible assumptions, APX_1 is strictly weaker than APC_1.

2. **Advanced formalizations.** We formalize in APX_1 several nontrivial results from algorithms and complexity, including the Blum-Luby-Rubinfeld linearity testing, Schwartz-Zippel lemma[1], and an average-case AC^0 lower bound for Parity. Additionally, as a byproduct of our refined analysis of AC^0 circuits in bounded arithmetic, we describe a matching worst-case lower bound in PV_1. The latter formalization addresses a problem considered by Razborov [Raz95], Krajíček [Kra95, Section 15.2], and Müller-Pich [MP20], which was only known for stronger theories.

3. **Tailored witnessing theorem.** We show that the provably total NP relations of APX_1 deterministically reduce to a natural TFZPP problem[2] we introduce, $\mathsf{Refuter}(\mathsf{Yao})$. Moreover, if $\mathsf{prBPP} = \mathsf{prP}$, then APX_1 admits deterministic polynomial-time witnessing.

4. **Feasible derandomization.** Using the new framework, we put forward a natural formalization of the fundamental question: Is $\mathsf{prP} = \mathsf{prBPP}$ feasibly provable? In other words, is there a deterministic feasible proof of general derandomization?

5. **Reverse mathematics of randomness.** Finally, we show that APX_1 serves as a suitable base theory for developing the reverse mathematics of average-case and randomized lower bounds, illustrated here through the study of randomized communication protocols and their communication complexity.

Before presenting our results in more detail, we provide additional context and background.

Dual use of $\mathsf{dWPHP}(\mathsf{PV})$ in APC_1. Why does APC_1, until now the weakest known theory capable of formalizing probabilities and randomized algorithms, appear stronger than necessary? By looking into the construction of APC_1 [Jeř04, Jeř05, Jeř07a], we observe two different reasons for introducing $\mathsf{dWPHP}(\mathsf{PV})$.

[1] It is worth noting that the standard proof of Schwartz-Zippel lemma (see, e.g., [AB09, Lemma A.36]) is not known to be formalizable even in APC_1. In this work, we formalize an alternative proof due to Atserias and Tzameret [AT25] in APX_1.

[2] A TFNP problem $R(x, y)$ is said to be in TFZPP if, for every input x, at least an inverse-polynomial fraction of strings y are valid solutions, i.e., $R(x, y) = 1$. It is clear that TFZPP problems admit simple zero-error randomized algorithms running in polynomial time.

- First, it is used to implement approximate counting. Jeřábek [Jeř04] shows that dWPHP(PV) proves the existence of an exponentially hard Boolean function, and by formalizing a form of correctness of the Nisan-Wigderson PRG [NW94] in the theory PV_1, we can approximately compute the acceptance probability of circuits by instantiating the PRG with the hard Boolean function.
- Second, it also serves as a counting principle to derive tools in combinatorics and probability theory, including the inclusion-exclusion principle, union bound, and Chernoff bound [Jeř07a, Section 2].

The first role appears essential, as approximate counting is the foundation for the formalization of probabilistic polynomial-time algorithms. However, dWPHP(PV), as a counting principle, appears to be overly powerful and not necessary for many applications.

Remark 1.1 (Computational Aspects of dWPHP). To add more context, dWPHP(PV) asserts that for any function f implemented by circuits whose co-domain is much larger than its domain, say $f : \{0,1\}^n \to \{0,1\}^{n+1}$, there exists a string in the co-domain that does not have a pre-image. The computational aspect of the principle, namely the search problem of finding such a string given a function f, has recently drawn attention in computational complexity (see [Kor25] for a survey). This problem, which is now called the *Range Avoidance Problem* [KKMP21, Kor21, RSW22], is known to be hard even for nondeterministic search algorithms under plausible assumptions [ILW23, CL24].

Axiomatizing approximate counting in APX_1. Since dWPHP(PV) fulfills two essential functions in APC_1, devising a weaker theory is nontrivial — one must find a way to relax the counting principle without sacrificing the capacity to formalize approximate counting.

Our approach, which in hindsight appears quite natural, is to put *approximate counting* at the foundation, elevating it to a central primitive rather than deriving it from stronger principles such as dWPHP(PV) [Jeř04, Jeř05, Jeř07a]. Starting from PV_1 as the base theory, we directly introduce an oracle that is intended to perform approximate counting, and govern it with appropriate axioms. Through this approach, we decouple *the concept of approximate counting* from *counting principles*.

The main technical challenge is to select an appropriate set of axioms. These axioms should be sufficiently strong to carry out our advanced formalizations, reverse mathematics results, and potentially more results in theoretical computer science. At the same time, the set of axioms should be minimal. The contradictory objectives make it hard to select appropriate axioms; indeed, it is not even a priori clear whether a suitable *finite* set of axioms exists without resorting to variants of dWPHP(PV).

Perhaps surprisingly, we distill four simple and intuitive axioms that suffice to implement all our results, among which the only nontrivial axiom captures the "local" behavior of the approximate counting oracles. Arguably, this makes APX_1 a plausible candidate for the minimal theory of probabilistic polynomial-time reasoning.

Remark 1.2 (Minimal Assumption for Derandomization). The conjectured inclusion $\mathsf{prBPP} \subseteq \mathsf{prP}$ is a central question in derandomization. The celebrated results of Impagliazzo, Nisan, and Wigderson [NW94, IW97] give a positive answer under $\mathsf{E} \not\subseteq \text{i.o.-}\mathsf{SIZE}[2^{\Omega(n)}]$, a plausible worst-case circuit lower bound. Conversely, derandomization results are also known to imply weaker circuit lower bounds such as $\mathsf{NTIME}(n^{\omega(1)}) \not\subseteq \mathsf{P}_{/\mathsf{poly}}$ (see, e.g., [IKW02, Wil14, Tel19]). Yet it has been a longstanding open problem whether the strong circuit lower bounds used in [NW94, IW97] are *necessary* for proving $\mathsf{prBPP} = \mathsf{prP}$. Indeed, there has been significant progress indicating that derandomization may *not* require strong circuit lower bounds, see, e.g., [For01, Gol11, CT21]. Moreover, several *characterizations* of $\mathsf{prBPP} = \mathsf{prP}$ have been recently discovered [LP22, Kor22, CT23, CTW23, LPT24], motivated by the question of understanding the *minimal* assumption required for derandomization.

In a sense, our results attempt to address a similar question in the context of proof complexity. We aim to propose a *minimal theory* that is strong enough to carry out meaningful feasible proofs on probabilistic polynomial-time algorithms. In particular, we provide evidence that dWPHP(PV) and the existence of hard Boolean functions, which are at the foundation of Jeřábek's theory APC_1 [Jeř04, Jeř05, Jeř07a], might not be necessary in a minimal theory for probabilistic polynomial-time reasoning.

1.2 Main Contributions

We now describe our contributions and their implications in detail.

1.2.1 Theory APX_1

As alluded to above, rather than deriving probabilities from stronger combinatorial principles (as in APC_1 via $\mathsf{dWPHP}(\mathsf{PV})$), we axiomatize approximate counting directly. Our aim is a weaker theory in which the probability of any *feasibly definable event*[3] can be named and reasoned about with additive slack, while keeping proof-theoretic strength low.

To achieve this, we introduce a first-order bounded arithmetic theory, APX_1, whose central primitive is an *approximate counting* function P. Intuitively, given a Boolean circuit C on n input bits and a *precision parameter* Δ, the term $\mathsf{P}(C, \Delta)$ returns a rational number in $[0, 1]$ that approximates the acceptance probability of C within additive error $1/|\Delta|$, where $|\Delta|$ denotes the bitlength of the input parameter Δ. For convenience, we often write $\mathsf{P}_\delta(C)$ instead of $\mathsf{P}(C, \Delta)$, where $\delta = 1/|\Delta|$.

The equational core of APX_1, called APX, is obtained by extending Cook's equational theory PV with the oracle symbol P (the new language is denoted $\mathsf{PV}(\mathsf{P})$) and its governing axioms. APX_1 is then the usual first–order closure of APX, i.e., universal closures of APX-equations together with the standard PV-style induction on notation.

Remark 1.3 (PV and PV_1). PV [Coo75] is an equational theory whose intended model is $\mathbb{N}$ with the usual interpretation of basic symbols such as 0, $+$, and $\times$. Its language contains a function symbol for every polynomial-time algorithm $f\colon \mathbb{N}^k \to \mathbb{N}$ (for any fixed k); these symbols and their defining axioms are given via Cobham's characterization of the polynomial-time functions. The theory includes an induction scheme formalizing binary search and, in particular, proves induction for quantifier-free formulas (i.e., polynomial-time predicates). A standard first-order strengthening is PV_1 [KPT91]. While the formal definition of PV_1 is fairly technical, the theory is robust: distinct presentations yield the same theorems. For example, PV_1 has an equivalent axiomatization that avoids Cobham's theorem [Jeř06]; alternatively, it can be presented as the set of all $\forall\Sigma^b_1$-sentences provable in Buss's theory S^1_2 [Bus86]. We refer to [Oli25] for a brief overview and to [Li25] for a detailed introduction.

A key aspect of the definition of APX_1 is to employ "local" constraints governing the behavior of P_δ, which together enforce the "global" desired behavior, i.e., that P_δ approximates the acceptance probability of any input circuit up to an additive error term δ. The entire probabilistic machinery of APX_1 (random variables, expectation, tail bounds, etc.) is built on top of the axioms below.

APX_1 axioms governing P. All axioms are universal $\mathsf{PV}(\mathsf{P})$-equations; below $\beta^{-1} \in \mathsf{Log}$ is a freely available "slack" parameter used to absorb routine finite-precision effects.[4] We sketch the statements at an informal level; the formal version appears in Section 2.

- **Basic Axiom.** For every Boolean circuit C and Δ, the value $\mathsf{P}(C, \Delta)$ is a rational in $[0, 1]$ (encoded in PV) and all $\mathsf{PV}(\mathsf{P})$-provable equations hold. Together with an output-length bound for $\mathsf{P}(C, \Delta)$, this forces feasibility of approximate counting at any requested precision.

- **Boundary Axiom.** If C is syntactically constant (reads no inputs), then $\mathsf{P}_\delta(C) \in \{0, 1\}$ agrees with the output bit of C. Thus P is *exact* on trivial cases.

- **Precision Consistency.** For any two precisions δ_1, δ_2 and circuit C,

$$\left|\mathsf{P}_{\delta_1}(C) - \mathsf{P}_{\delta_2}(C)\right| \ \leq\ \delta_1 + \delta_2 + \beta.$$

[3] In other words, an event $E \subseteq \{0, 1\}^m$ for which there is a polynomial-size Boolean circuit C such that $C(x) = 1$ if and only if $x \in E$.

[4] The expression $\beta^{-1} \in \mathsf{Log}$ is standard notation in bounded arithmetic used to denote that $\beta = 1/|y|$ for some variable y, where $|y|$ is the bitlength of y.

Hence asking for finer precision can only move the reported probability by the sum of the specified error parameters (up to β).

- **Local Consistency.** If C has at least one input bit, and $\mathsf{Fix}_b(C)$ denotes the circuit obtained by fixing the rightmost input bit to $b \in \{0,1\}$, then

$$\left| \mathsf{P}_\delta(C) \;-\; \tfrac{1}{2}\big(\mathsf{P}_\delta(\mathsf{Fix}_0(C)) + \mathsf{P}_\delta(\mathsf{Fix}_1(C))\big) \right| \;\le\; 2\delta + \beta.$$

  Thus the reported acceptance probability of C is (up to additive slack of β) the average of the reported probabilities after fixing a fresh random bit. This aims to capture the intended semantics of counting over the uniform hypercube.[5]

In practice, one can think of APX_1 as the theory extending PV_1 with the symbol P and its governing axioms, together with induction over quantifier-free formulas in the language $\mathsf{PV}(\mathsf{P})$. Everything else – such as random variables, expectation, union bound, etc. – will be introduced and derived from the language and axioms inside APX_1.

Soundness of approximate counting in APX_1. We say that a PV-standard model (i.e., $\mathbb{N}$) where the function symbol P is interpreted by any correct approximate counting function (returning a rational within $\pm 1/|\Delta|$ and exact on syntactically constant circuits) is a *standard model* of APX_1. A simple but central result shows that these are *exactly* the models satisfying the axioms ("admissible models"), yielding semantic soundness for the intended interpretation and a correct axiomatization of approximate counting when the underlying model is $\mathbb{N}$ (see Section 2.3).

Minimality of APX_1. We believe that APX_1 is a good candidate of the *minimal theory* for probabilistic polynomial-time reasoning. This is not a formal assertion from a mathematical perspective. However, the axioms above appear close to the weakest workable base theory that can consistently *define* and *operate on* approximate probabilities of feasibly described events. Specifically:

- Any theory that reasons about probabilistic polynomial time algorithms should be able to *define* the acceptance probability of the algorithms. This requires the capability of approximate counting with an additive error, i.e., the symbol P.

- Because the *Basic Axiom* and the *Boundary Axiom* are rather syntactic promises of the oracle P, we expect them to be available. Arguably, the *Precision Consistency Axiom*, which asserts the consistency of P on different precision parameters, should also be available. Note that these three axioms are not sufficient, as one can easily specify a trivial and incorrect polynomial-time function such that these axioms are provable in PV_1.

- Therefore, we use the *Local Consistency Axiom* to capture the correctness of P — it shows that the approximate counting oracle withstands a simple statistical test with three queries made throughout the proof. It seems unlikely that one can make nontrivial use of an oracle for the purpose of approximate counting that may fail this test; subsequently, the axiom also seems necessary.

Another evidence of the minimality of APX_1 is that, computationally, the function symbol P aligns with the *Circuit Acceptance Probability Problem* (CAPP), which is complete for prBPP (see, e.g., [Vad12]). In contrast, the *Range Avoidance Problem*, which corresponds to $\mathsf{dWPHP}(\mathsf{PV})$ and is relevant for APC_1, is likely hard even against nondeterministic algorithms [ILW23, CL24].

[5] In other words, for every (feasibly definable) set $X \subseteq \{0,1\}^n$, as X is the disjoint union of X_0 and X_1, where $X_b = \{x \in X \mid x_n = b\}$, we expect $|X| \approx |X_0| + |X_1|$.

1.2.2 Probabilistic Reasoning in APX_1

We develop a self-contained "probabilistic calculus" inside APX_1 using the approximate counting function P_δ. As a preliminary step, we show that P_δ behaves in the expected way on feasibly described events. Concretely, APX_1 establishes the following properties (each up to an arbitrarily small additive slack $\beta^{-1} \in \mathsf{Log}$):

- **Semantic invariance.** P_δ respects semantic equivalence, i.e., if APX_1 proves that circuits C and D compute the same function, then $\left|\mathsf{P}_\delta(C) - \mathsf{P}_\delta(D)\right| \leq 2 \cdot \delta + \beta$ (Lemma 3.2).
- **Permutation invariance.** Permuting input bits does not noticeably change the value of P_δ. In other words, for any circuit C and permutation π of input bits, $\left|\mathsf{P}_\delta(C \circ \pi) - \mathsf{P}_\delta(C)\right| \leq 2 \cdot \delta + \beta$ (Lemma 3.5).
- **Existence via the probabilistic method.** Suppose that strings accepted by a circuit C are considered *good*. Then if good strings are abundant, i.e., more than $(\delta + \beta)$-fraction with respect to precision parameter δ, there must exist a good string (Lemma 3.6). A bit more formally,

$$\mathsf{APX}_1 \vdash \forall n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}\ \forall C\ (\beta > 0 \land \mathsf{P}_\delta(C) > \delta + \beta \rightarrow \exists x \in \{0,1\}^n\ C(x) = 1).$$

- **Consistency on concrete circuits.** P_δ agrees with simple tests. For instance, for a naturally defined threshold circuit $C_{<t}(x)$ on n-bit inputs that accepts if and only if x (viewed as an integer) is less than t, $\mathsf{P}_\delta(C_{<t}) \approx t/2^n$ (see Section 3.1.4).

These meta-properties ensure that the definitions and inequalities developed in APX_1 inherit the intended probabilistic behavior with only small, explicitly controlled additive losses.

With these guarantees in place, we now introduce *feasible random variables.* A random variable X is specified by an explicit support $V \subseteq \mathbb{Q}$, a seed length n, and a multi-output sampler circuit $C : \{0,1\}^n \rightarrow V$. Its *approximate expectation* is defined by querying P_δ on the indicator Boolean circuits $\{C_v\}_{v \in V}$, where $C_v(z)$ accepts z if and only if $C(z) = v$. In other words:

$$\mathbb{E}_\delta[X] \triangleq \sum_{v \in V} v \cdot \mathsf{P}_\delta(C_v).$$

We observe that there exists a $\mathsf{PV}(\mathsf{P})$ function $\mathsf{E}(V, n, C, \Delta)$ that computes $\mathbb{E}_{|\Delta|^{-1}}[X]$ for the random variable X defined by (V, n, C). Specifically, E enumerates all $v \in V$, constructs the corresponding circuit C_v, queries the oracle to obtain $p_v \leftarrow \mathsf{P}(C_v, \Delta)$, and outputs the sum $\sum_{v \in V} v \cdot p_v$.

We introduce a central technical tool that provides a general version of the *averaging argument for expectation* (Section 3.2.3): Given random variables $X_1, \ldots, X_m$ on the same seed and coefficients $\lambda_1, \ldots, \lambda_m$, APX_1 can *search* for a suffix z of the seed such that a lower bound on the value $\sum_i \lambda_i \cdot \mathbb{E}_\delta[X_i]$ is approximately preserved after fixing that suffix. This is used repeatedly to move between *global* and *pointwise* statements and underlies the proof of several results. We explain this technique in more detail in Section 1.3.

Using the tool described above, together with some additional ideas, APX_1 derives approximate formulations of several standard probability inequalities. In particular, it establishes the *linearity of expectation* for linear combinations of feasible random variables, the *union bound* for polynomially many events, and *Markov's inequality* for non-negative variables with the usual $1/k$ decay.[6]

Remark 1.4 (Example: Union Bound in APX_1; see Theorem 3.20). APX_1 proves the following statement. Let $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C_1, \ldots, C_m$ be single-output circuits, and $V = \{0,1\}$. Suppose that $\forall x \in \{0,1\}^n$ and $i \in [m]$, $C_i(x) \in V$, and let $Y, X_1, \ldots, X_m$ be random variables defined as follows.

- For each $i \in [m]$, X_i is defined by (V, n, C_i).
- Y is defined by (V, n, S), where $S(x) \in \{0,1\}$ is a circuit such that $S(x) \leq C_1(x) \lor \cdots \lor C_m(x)$.

[6] We note that some results suffer an approximation loss that depends on $\|V\| \triangleq \sum_{v \in V} |v|$ and on the magnitude of the involved coefficients, where here $|\cdot|$ denotes absolute value. In applications where these quantities are polynomially bounded, this can be mitigated by taking sufficiently small parameters δ and β in the application of $\mathbb{E}_\delta[X]$.

Then we have $\mathbb{E}_\delta[Y] \leq \mathbb{E}_\delta[X_1] + \cdots + \mathbb{E}_\delta[X_m] + (2\delta + \beta) \cdot m$.

Additionally, APX_1 defines an *approximate variance* $\mathsf{Var}_\delta[X] \triangleq \mathbb{E}_\delta\big[(X-\mu)^2\big]$, with $\mu \triangleq \mathbb{E}_\delta[X]$, and shows an identity of the form $\mathsf{Var}_\delta[X] \approx \mathbb{E}_\delta[X^2] - \mu^2$, which leads to a natural formulation of *Chebyshev's inequality*. APX_1 also formalizes *(almost) pairwise independence* via approximate covariance, and proves that the variance of a sum is (approximately) the sum of variances for (almost) pairwise independent variables.

Finally, we address *independence and concentration*. We work in APX_1 with *explicit* independence: variables are sampled by disjoint parts of the seed. Under this notion, the theory proves a *multiplication principle*

$$\mathbb{E}_\delta[XY] \;\approx\; \mathbb{E}_\delta[X] \cdot \mathbb{E}_\delta[Y],$$

and, for Bernoulli variables, a convenient product bound

$$\left|\mathbb{E}_\delta\Big[\prod_{i=1}^m X_i\Big] - \prod_{i=1}^m \mathbb{E}_\delta[X_i]\right| \;\leq\; 8\delta \cdot m.$$

These yield *one-sided error reduction*. Moreover, APX_1 proves a *Chernoff bound* for sums of $m = O(\log n)$ i.i.d. Bernoulli random variables; the bound has the standard exponential tail with controlled additive slack.

Remark 1.5 (Example: One-Sided Error Reduction in APX_1; see Theorem 3.33). For a Boolean circuit $C : \{0,1\}^n \to \{0,1\}$, let $C^{\vee k} : \{0,1\}^{nk} \to \{0,1\}$ be the circuit defined as $C^{\vee k}(x_1, \ldots, x_k) \triangleq \bigvee_{i \in [k]} C(x_i)$. The following statement is provable in APX_1. For any $n, k, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and $C : \{0,1\}^n \to \{0,1\}$, if $\mathsf{P}_\delta(\neg C) \leq \varepsilon$ then $\mathsf{P}_\delta(\neg C^{\vee k}) \leq (\delta + \beta + \varepsilon)^k + \delta + \beta$.

We refer to Section 3 for a detailed description of how these different notions and results are implemented in APX_1.

1.2.3 Theoretical Computer Science in APX_1

As explained above, approximate counting – as axiomatized in APX_1 – suffices to build the typical probabilistic toolkit (such as existence arguments, linearity of expectation, averaging argument, union bound, Markov, Chebyshev, limited independence, error reduction, and a version of Chernoff for logarithmically many samples). This lightweight yet robust framework can be exploited to formalize several nontrivial results. We illustrate this point through a set of detailed formalizations of influential results from different areas of theoretical computer science:

- Yao's distinguisher-to-predictor transformation via the hybrid argument, a central tool in computational pseudorandomness (see Theorem 4.1);
- the Schwartz-Zippel Lemma (as stated in [AT25]), an algebraic result for polynomial identity testing with broad applications in randomness and complexity (see Theorem 4.4);
- the classical lower bound for the parity function against bounded-depth polynomial-size circuits in circuit complexity (see Section 4.4);
- the correctness of the Blum-Luby-Rubinfeld linearity test from sublinear time algorithms and property testing (see Section 4.5).

For concreteness and in order to contrast our results with previous work, we focus here on the formalization of circuit lower bounds for the n-bit parity function, denoted $\oplus_n$. In fact, we show that a stronger *average-case* lower bound against depth-d Boolean circuits (AC^0_d) can be proved in APX_1.

Theorem 1.6 (Average-Case AC^0 Lower Bound for $\oplus_n$ in APX_1). *For all constants $k, d \geq 1$, there exists a constant $n_0 \geq 1$ such that APX_1 proves the following statement. Let $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $n > n_0$, and*

$C : \{0,1\}^n \to \{0,1\}$ *be an* AC^0_d *circuit of size at most* n^k*. Let* $T_C : \{0,1\}^n \to \{0,1\}$ *be the circuit that, given* $x \in \{0,1\}^n$*, outputs* 1 *if and only if* $C(x) = \oplus_n(x)$*. Then*

$$\mathsf{P}_\delta(T_C) \leq \frac{1}{2} + \frac{1}{n^k} + \delta + \beta. \tag{1.1}$$

The main technical challenge is to avoid "encoding-based counting arguments" (pigeonhole-principle variants) unavailable in APX_1, such as those used in Razborov's proof of the switching lemma [Raz95]. Instead, our proof builds on a technique of Furst, Saxe, and Sipser [FSS84]. The approach was refined by [AAI+01] (see also [Agr01]), who gave a deterministic polynomial-time algorithm that outputs an appropriate restriction supplied by the switching lemma. One of our contributions is to show that the correctness of the algorithm in [AAI+01] can be established within APX_1. Combined with the probabilistic tools above and other ideas, this yields the *average-case* lower bound in APX_1.

As a consequence of our refined proof-theoretic framework, and with some additional effort, we can extract from the above formalization a *worst-case* lower bound within the weaker theory PV_1.

Theorem 1.7 (Worst-Case AC^0 Lower Bound for $\oplus_n$ in PV_1)**.** *For all constants* $k, d \geq 1$*, there exists a constant* $n_0 \geq 1$ *such that* PV_1 *proves the following statement. For every* $n \in \mathsf{Log}$*,* $n > n_0$*, and* AC^0_d *circuit* $C : \{0,1\}^n \to \{0,1\}$ *of size at most* n^k*, there exists a string* $x \in \{0,1\}^n$ *such that* $C(x) \neq \oplus_n(x)$*.*

Earlier formalizations of the worst-case parity lower bound for bounded-depth circuits required stronger theories. In particular, [MP20] and [Kra95] formalize different proofs in $\mathsf{APC}_1 = \mathsf{PV}_1 + \mathsf{dWPHP}(\mathsf{PV})$, while [Raz95] works in PV_1 but in the LogLog regime – i.e., with n of doubly logarithmic order – so the proof can manipulate exponentially large objects (see [MP20] for details).

These formalizations reinforce the intuition that a substantial portion of results in algorithms and complexity theory are already captured within PV_1 or its mild extensions, and that establishing unprovability results would therefore be of considerable significance (see [Oli25] for related discussions).

In Section 1.3 below, we elaborate on the proofs of Theorem 1.6 and Theorem 1.7. For further details about these and other formalizations, see Section 4.

1.2.4 Witnessing, Relative Strength of APX_1, and Provability of $\mathsf{prBPP} = \mathsf{prP}$

We now discuss relations between theories PV_1, APX_1, and APC_1, and connections to the prBPP versus prP problem. We also introduce a new computational problem called $\mathsf{Refuter}(\mathsf{Yao})$, and provide a tailored witnessing theorem for the $\forall\Sigma^b_1(\mathsf{PV})$-consequences of APX_1 (i.e. provably total TFNP problems in APX_1).

APX_1 versus APC_1. By construction, every sentence provable in PV_1 is also a theorem of APX_1. It is also possible to show that if φ is a sentence in the language of PV_1 (i.e., without the approximate counting symbol P) provable in APX_1, then it is provable in APC_1 (see Corollary 5.10). This means that, modulo the difference in languages (i.e. APC_1 does not have the symbol P), APX_1 is a sub-theory of APC_1.[7]

On the other hand, under plausible computational assumptions, there are sentences provable in APC_1 that are not provable in APX_1 (see Corollary 5.14).[8] This is obtained by adapting a technique from [ILW23]. In other words, PV_1 is contained in APX_1, while APX_1 is likely strictly weaker than APC_1.

Subsequently, a fundamental research direction is to determine whether APX_1 is stronger than PV_1, a question closely connected to the prBPP versus prP problem and to understanding the role of randomness in feasible proofs.

[7] Indeed, there is a conservative extension of APC_1 known as HARD^A [Jeř07a] that contains APX_1 in a stronger sense — the symbol P can be simulated by a term in HARD^A such that all axioms governing P are provable (see Theorem 5.9).

[8] More formally, there is a $\forall\Sigma^b_2(\mathsf{PV})$-sentence provable in APC_1 that is not provable in APX_1, under the existence of indistinguishability obfuscation and coNP not contained infinitely often in $\mathsf{NP}_{/\mathsf{poly}}$ (see Corollary 5.14).

PV_1 versus APX_1 and feasible derandomization. From a meta-mathematical standpoint, it is natural to ask whether $\mathsf{prBPP} = \mathsf{prP}$ is *(un)provable* in a weak arithmetic theory such as PV_1. A key obstacle is formalization: the language of PV_1 is tailored to *deterministic* polynomial-time functions, whereas the statement $\mathsf{prBPP} = \mathsf{prP}$ quantifies over acceptance probabilities of circuits on an exponentially large space. We propose the following question.

Open Problem 1. Is there a PV function symbol $\widetilde{\mathsf{P}}(C, \Delta)$ for which the *basic*, *boundary*, *precision consistency*, and *local consistency* axioms (Section 1.2.1) are provable in PV_1?

An unconditional positive answer seems out of reach at present, as it would immediately imply $\mathsf{prBPP} = \mathsf{prP}$ by the soundness of the approximate counting axioms and the polynomial running time of $\widetilde{\mathsf{P}}$ (see Theorem 2.5). Intuitively, this would amount to a *deterministic polynomial-time proof* of the collapse. At the moment, it is unclear whether a positive or a negative answer is more plausible.

A weaker possibility is that, even if no such PV function symbol $\widetilde{\mathsf{P}}(C, \Delta)$ exists with the axioms provable in PV_1, adding the approximate counting oracle P might nonetheless be conservative for *deterministic* statements in the base language. Formally:

Open Problem 2. Is APX_1 conservative over PV_1? Equivalently, does every first-order sentence in the language of PV that is provable in APX_1 already have a proof in PV_1?

A positive answer to Open Problem 1 would imply a positive answer here. The relationship between Open Problem 2 and $\mathsf{prBPP} = \mathsf{prP}$ appears incomparable. If $\mathsf{prBPP} = \mathsf{prP}$ holds but only via a non-feasible proof, APX_1 need not be conservative over PV_1. Conversely, even if APX_1 is conservative over PV_1, it is not clear to us whether $\mathsf{prBPP} = \mathsf{prP}$ follows. At a high level, we are interested in the relationship between *derandomization of computations* and *derandomization of proofs*. While we are currently unable to provide definite answers, we believe these questions are fundamental and merit further study. We refer to [Kra25] and references therein for related questions in the context of APC_1 versus PV_1.

A Witnessing Theorem for APX_1: Reductions to $\mathsf{Refuter}(\mathsf{Yao})$. A key characteristic of bounded theories is to have a suitable *witnessing theorem* corresponding to certain computational problems (see, e.g., [Bus86, KPT91]). We isolate a certain (total) search problem as a key computational task for producing witnesses for the $\forall\Sigma_1^b(\mathsf{PV})$-consequences of APX_1.

$\mathsf{Refuter}(\mathsf{Yao})$: An instance fixes the following parameters: input length n, multiset size m, predictor circuit description size s, and advantage $\delta > 0$. The input is a *predictor generator*, i.e., a circuit

$$G\colon \{0,1\}^{nm} \to [n] \times \{0,1\}^s$$

which, on a flat distribution $\mathcal{D} \in (\{0,1\}^n)^m$ (i.e., an m-tuple of n-bit strings), returns an index $i \in [n]$ and the description of a predictor circuit $P\colon \{0,1\}^{i-1} \to \{0,1\}$ of size s. A *solution* is any flat distribution $\mathcal{D}$ such that, writing $(i, P) = G(\mathcal{D})$,

$$\Pr_{x \leftarrow \mathcal{D}}\big[P(x_{<i}) = x_i\big] < \tfrac{1}{2} + \delta.$$

Thus a solution $\mathcal{D}$ *refutes* that G can produce predictors for any distribution with advantage δ. When parameters satisfy $(\delta^2/10) \cdot m \geq s + \lceil \log n \rceil + 1$, $\mathsf{Refuter}(\mathsf{Yao})$ lies in TFZPP; in other words, uniformly random distribution is likely a solution. The problem is called $\mathsf{Refuter}(\mathsf{Yao})$, as G is intended to output a predictor like the standard "distinguisher→predictor" transformation of Yao [Yao82].

Remark 1.8 ($\mathsf{Refuter}(\mathsf{Yao})$ and Derandomization). Note that $\mathsf{Refuter}(\mathsf{Yao})$ requires generating a distribution $\mathcal{D}$ that is unpredictable with respect to a *given* predictor generator G – a deterministic procedure that attempts to produce a predictor P for $\mathcal{D}$. The distribution $\mathcal{D}$ need not be pseudorandom (or equivalently, unpredictable) against *all small circuits*; it only needs to fool the specific generator G. This can be viewed as a special case of constructing *targeted pseudorandom generators*, a task known to be prBPP-complete (see [Gol11, CT21, LPT24]).

We establish the following result for the provably total TFNP problems of APX_1.

Theorem 1.9 (Witnessing for APX_1). *Let $\varphi(x, y)$ be a quantifier-free formula in the language of PV_1. If $\mathsf{APX}_1 \vdash \forall x\, \exists y\, \varphi(x, y)$, there exists a deterministic polynomial-time Turing reduction from the search problem defined by φ to $\mathsf{Refuter(Yao)}$ with parameters satisfying $(\delta^2/10) \cdot m \geq s + \lceil \log n \rceil + 1$.*

In Section 1.3 below, we provide an overview of the proof of Theorem 1.9.

Relation to $\mathsf{LossyCode}$ and APC_1. Recall the definition of the search problem $\mathsf{LossyCode}$ [Kor22]: given a compressor circuit $C\colon \{0,1\}^n \to \{0,1\}^{n-1}$ and a decompressor circuit $D\colon \{0,1\}^{n-1} \to \{0,1\}^n$, output x with $D(C(x)) \neq x$. Similarly to $\mathsf{Refuter(Yao)}$, this problem is total and in TFZPP.

We observe the existence of a deterministic polynomial-time mapping reduction from $\mathsf{Refuter(Yao)}$ to $\mathsf{LossyCode}$ whenever the input instances of $\mathsf{Refuter(Yao)}$ satisfy

$$(\delta^2/10) \cdot m \;\geq\; s + \lceil \log n \rceil + 1.$$

Therefore, in the stated regime, derandomizing $\mathsf{LossyCode}$ subsumes derandomizing $\mathsf{Refuter(Yao)}$. Since every APX_1-provably total TFNP problem reduces to $\mathsf{Refuter(Yao)}$, under the above parameter condition it further reduces to $\mathsf{LossyCode}$.

Recall that Wilkie (unpublished) and Thapen [Tha02] (see [Jeř04, Proposition 1.14] and [LPT24, Theorem D.1]) proved that $\mathsf{LossyCode}$ captures the $\forall\Sigma^b_1$-fragment of APC_1. Consequently, these results organize the TFNP landscapes of the two theories: $\mathsf{LossyCode}$ witnesses the $\forall\Sigma^b_1$-consequences of APC_1, while $\mathsf{Refuter(Yao)}$ witnesses those of APX_1.

We return to these topics in Section 5, providing detailed proofs of all results mentioned above and further discussions.

1.2.5 Reverse Mathematics of Randomized and Average-Case Lower Bounds

The *retraction weak pigeonhole principle* ($\mathsf{rWPHP(PV)}$) [Jeř07b, LLR24, CLO24] is one of the most important combinatorial principles known to be provable in APC_1, but whose provability in APX_1 remains unclear. Recall that $\mathsf{rWPHP(PV)}$ asserts that for every $n, m \in \mathsf{Log}$ with $m < n$ and for all (deterministic) circuits $C\colon \{0,1\}^n \to \{0,1\}^m$ ("compressor") and $D\colon \{0,1\}^m \to \{0,1\}^n$ ("decompressor"), there is $x \in \{0,1\}^n$ such that $D(C(x)) \neq x$.

In other words, $\mathsf{rWPHP(PV)}$ captures the combinatorial principle underlying the total search problem $\mathsf{LossyCode}$ discussed above. Its provability in APX_1 would mean that APX_1 and APC_1 prove the same $\forall\Sigma^b_1(\mathsf{PV})$ sentences, and by the witnessing theorem (see Theorem 1.9), this would further imply that $\mathsf{LossyCode}$ and $\mathsf{Refuter(Yao)}$ are equivalent with respect to deterministic polynomial-time Turing reductions.

We study *counting variants* of the retraction weak pigeonhole principle and characterize their *equivalence class* with respect to provability in APX_1. We show that this class encompasses certain *communication complexity lower bounds against randomized protocols*, establishing that these results are all equivalent (over the base theory APX_1) to suitable variants of the retraction pigeonhole principle.

Counting Variants of $\mathsf{rWPHP(PV)}$. We consider the following statements:

- <u>Approximate Counting rWPHP</u>: $\#\mathsf{rWPHP}[m, \varepsilon]$.

  For any *deterministic* compressor-decompressor pair with encoding length $m < n$, an ε-fraction of inputs cannot be correctly decompressed.

- <u>Randomized Compression rWPHP</u>: $\mathsf{rrWPHP}[m, \varepsilon]$.

  For a *randomized* compressor and a deterministic decompressor with encoding length $m < n$, there is some input on which the pair has error probability at least ε.

These principles are formalized in a natural way using the probabilistic framework provided by APX_1.

One-Way Communication Complexity. We prove an equivalence result involving communication complexity (CC) lower bounds against randomized *one-way protocols* with either *public randomness* or *private randomness*. Recall that the Set Disjointness function $\mathsf{SetDisj}(x, y)$ outputs 1 if and only if for every index $i \in [n]$, either $x_i = 0$ or $y_i = 0$, i.e., x and y have no common 1-index. The following statements, presented informally for clarity, are relevant to our result:

- Public Randomized CC Lower Bound for Set Disjointness: $\mathsf{pub\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{SetDisj}}[m, \varepsilon]$.

  Every public-coin one-way protocol computing $\mathsf{SetDisj}$ with communication complexity m must have error probability at least ε on some input pair (x, y).

- Private Randomized CC Lower Bound for Set Disjointness: $\mathsf{priv\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{SetDisj}}[m, \varepsilon]$.

  Every private-coin one-way protocol computing $\mathsf{SetDisj}$ with communication complexity m must have error probability at least ε on some input pair (x, y).

- Public Randomized CC Lower Bound for Some Function: $\mathsf{pub\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{some}}[m, \varepsilon]$.

  For every $n \in \mathsf{Log}$, there exists $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ such that $\mathsf{pub\text{-}r}\underrightarrow{\mathsf{LB}}^{f}[m, \varepsilon]$ holds.

- Private Randomized CC Lower Bound for Some Function: $\mathsf{priv\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{some}}[m, \varepsilon]$.

  For every $n \in \mathsf{Log}$, there exists $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ such that $\mathsf{priv\text{-}r}\underrightarrow{\mathsf{LB}}^{f}[m, \varepsilon]$ holds.

We leave the details about the formalization of the corresponding lower bound sentences to Section 6. We note that APX_1 is able to show that some concrete functions admit low-cost communication protocols. For instance, using linear hashing, it proves that $\mathsf{Equality}$ admits public-randomness one-way communication protocols of cost $O(\log n)$.

We can state an informal version of our equivalence result as follows.[9]

Theorem 1.10 (Main Equivalence Result (Informal); see Theorem 6.7)**.** *The following statements are equivalent over* APX_1*, for suitable relations between the constants* $k \geq 1$ *and* $0 < \varepsilon < 1$*, quantified outside the theory:*

(1) $\#\mathsf{rWPHP}[n-1, n^{-k}]$

(2) $\#\mathsf{rWPHP}[n^{\varepsilon}, n^{-k}]$

(3) $\mathsf{rrWPHP}[n-1, n^{-k}]$

(4) $\mathsf{rrWPHP}[n^{\varepsilon}, n^{-k}]$

(5) $\mathsf{pub\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{SetDisj}}[n-1, n^{-k}]$

(6) $\mathsf{pub\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{SetDisj}}[n^{\varepsilon}, n^{-k}]$

(7) $\mathsf{priv\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{SetDisj}}[n-1, n^{-k}]$

(8) $\mathsf{priv\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{SetDisj}}[n^{\varepsilon}, n^{-k}]$

(9) $\mathsf{pub\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{some}}[n-1, n^{-k}]$

(10) $\mathsf{pub\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{some}}[n^{\varepsilon}, n^{-k}]$

(11) $\mathsf{priv\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{some}}[n-1, n^{-k}]$

(12) $\mathsf{priv\text{-}r}\underrightarrow{\mathsf{LB}}^{\mathsf{some}}[n^{\varepsilon}, n^{-k}]$

As a consequence, one of these statements is provable in APX_1 if and only if every statement in Theorem 1.10 is provable in APX_1. This result provides evidence that APX_1 can serve as a suitable base theory for developing the reverse mathematics of average-case and randomized lower bounds.

For more details and additional discussion, we refer to Section 6.

1.3 Techniques

We next outline some of the main techniques used in our proofs, starting with a recurring argument that establishes basic probabilistic inequalities in APX_1.

[9] In particular, the simplified formulation given here omits considerations about the number of random bits employed in the randomized protocols, which plays a role in the parameters of some statements.

1.3.1 Probabilistic Reasoning in APX_1: The "Pointwise to Global" Technique (Section 1.2.2)

At the core of our probabilistic reasoning is a simple but powerful seed-fixing lemma that lets us pass from *global* inequalities to *pointwise* statements about suitably chosen restrictions of the randomness. Recall that a feasible random variable X is specified by an explicit support $V \subseteq \mathbb{Q}$, a seed length n, and a multi-output circuit $C : \{0,1\}^n \to V$. Its approximate expectation is

$$\mathbb{E}_\delta[X] \triangleq \sum_{v \in V} v \cdot \mathsf{P}_\delta(C_v),$$

where C_v is the indicator circuit for the event $C(x) = v$. This is a $\mathsf{PV}(\mathsf{P})$-computable quantity.

A general averaging argument for expectation (Theorem 3.17). The following holds in APX_1. Let $X_1, \ldots, X_m$ be random variables over the same seed with support V, and fix coefficients $\lambda_1, \ldots, \lambda_m \in \mathbb{Q}$. Write

$$\mu \triangleq \sum_{i=1}^{m} \lambda_i \, \mathbb{E}_\delta[X_i], \qquad \mu \restriction z \triangleq \sum_{i=1}^{m} \lambda_i \, \mathbb{E}_\delta[X_i \mid z],$$

where $X_i \mid z$ denotes X_i after fixing a suffix of the seed to z. Then, for every desired suffix length k, there exists $z \in \{0,1\}^k$ such that

$$\mu \restriction z \;\geq\; \mu \;-\; (2\delta + \beta) \cdot \|\lambda\|_1 \cdot \|V\|_1. \tag{3.7}$$

Thus a lower bound on μ can be *witnessed* (up to controlled additive slack) by conditioning on a partial assignment of the seed.

The proof iteratively fixes one seed bit at a time. By a form of Local Consistency for expectation, the average of $\mu \restriction (0 \circ z)$ and $\mu \restriction (1 \circ z)$ is close to $\mu \restriction z$; hence it is possible to prove that one of the two extensions preserves the current value up to an additive loss $O(\eta \cdot \|\lambda\|_1 \cdot \|V\|_1)$, where η is an auxiliary parameter in the proof. Greedily repeating this for k steps yields a k-bit suffix with total loss $O(k \cdot \eta \cdot \|\lambda\|_1 \|V\|_1)$. A precision-smoothing argument (switching from η to δ via precision consistency) then gives the stated $(2\delta + \beta)$-type bound, independent of k. The greedy construction is formally captured by a $\mathsf{PV}(\mathsf{P})$-procedure $\mathsf{AvgSampler}$. Conceptually, $\mathsf{AvgSampler}$ searches for a good suffix using calls to the approximate counting oracle P, and its correctness is established using (polynomial) induction on k over a quantifier-free $\mathsf{PV}(\mathsf{P})$-formula, which is available in APX_1.

Remark 1.11 (Example: Consistency of Complementation (Corollary 3.18).). Let $E_1, E_2 : \{0,1\}^n \to \{0,1\}$ be complementary predicates, i.e., $E_1 = \neg E_2$ as Boolean circuits. Let X_1, X_2 be their indicator variables. We argue in APX_1. Given an arbitrary $\beta^{-1} \in \mathsf{Log}$, we set $\eta \triangleq \beta / C$, for a large enough constant C. Pointwise, for *every full assignment* ρ to the seed, using the relation between E_1 and E_2 we have

$$\mathbb{E}_\eta[X_1 \mid \rho] + \mathbb{E}_\eta[X_2 \mid \rho] - 1 \;=\; 0.$$

Set $\lambda \triangleq (1, 1, -1)$ and consider $\mu \triangleq \mathbb{E}_\eta[X_1] + \mathbb{E}_\eta[X_2] - 1$. Applying Theorem 3.17 with $k = n$, we obtain a ρ such that $\mu \leq \mu \restriction \rho + O(\eta)$. But $\mu \restriction \rho$ is exactly 0 by the pointwise identity above, yielding $\mathbb{E}_\eta[X_1] + \mathbb{E}_\eta[X_2] - 1 = O(\eta)$. Similarly, one can show that $-\mathbb{E}_\eta[X_1] - \mathbb{E}_\eta[X_2] + 1 = O(\eta)$. Translating expectations back to probabilities via the indicator correspondence, and applying a standard precision-smoothing argument, one can conclude that P_δ is consistent with complementation, i.e.,

$$\left|\mathsf{P}_\delta(E_1) + \mathsf{P}_\delta(E_2) - 1\right| \;\leq\; 2\delta + \beta.$$

To summarize, Theorem 3.17 provides a way to fix randomness while preserving lower bounds on linear combinations of expectations. It turns equalities or inequalities that hold *for each seed* into quantitative global bounds in APX_1 with explicit additive slack depending only on δ, β and the natural ℓ_1 norms of the supports and coefficients. This mechanism is the engine that drives many of our probability inequalities and applications in Section 3.

Remark 1.12. The bit-by-bit fixing trick is a standard technique in computational complexity theory. For instance, it is used in the search-to-decision reduction for SAT [AB09, Section 2.5] and to derive circuit lower bounds from derandomization [AvM11], among other results. In particular, our approach is inspired by a new proof of $\mathsf{BPP} \subseteq \mathsf{MA} \subseteq \Sigma_2^p$ via a bit-by-bit "dueling argument" [LPT24, Lemma A.10].

1.3.2 Provability of Circuit Lower Bounds (Theorem 1.6 and 1.7)

Theorem 1.6 gives an *average-case* lower bound for the parity function against depth-d AC^0 circuits, formalized in APX_1, while Theorem 1.7 gives a corresponding *worst-case* lower bound, formalized in the weaker theory PV_1. As alluded to above, the challenge is to avoid encoding-based arguments that rely on pigeonhole principles (or frameworks that build on them), since they are unavailable in these theories. The first result showcases how APX_1 approximate-probability calculus supports average-case arguments in a more sophisticated setting, while the second shows that, with additional derandomization work, we can carry a corresponding worst-case lower bound argument entirely within PV_1.

At a high level, the formalizations implement a simplification and derandomization [AAI+01] of the Furst–Saxe–Sipser [FSS84] *random restriction approach* to AC^0 lower bounds. Recall that the argument proceeds in stages, where at each stage we fix a suitable partial restriction $\rho\colon [n] \to \{0, 1, *\}$ that sets all input variables in $T \triangleq \rho^{-1}(\{0, 1\}) \subseteq [n]$. The crucial point is that a depth-d circuit C *simplifies* when restricted by ρ, leading to a not much larger circuit $C \restriction \rho$ of depth $d - 1$, while the parity function *retains its hardness*.

Two central lemmas employed in the specification of ρ drive the proofs of Theorem 1.6 and Theorem 1.7.

Subset Selection Lemma (Lemma 4.8). The first step is to algorithmically choose the set $T \subseteq [n]$ of variables with a "*narrow-or-wide*" guarantee: for every bounded-width CNF/DNF F at the bottom of the circuit C, after fixing the variables in T, F either already depends on few literals (narrow) or contains many disjoint subclauses supported on T (wide). Crucially, this subset selection is *constructed and proved correct* in PV_1 using a delicate potential-function argument that simulates the method of derandomization via conditional expectations. This makes the selection of the subset $T \subseteq [n]$ feasible in our theories, not merely existential.

Restriction Selection Lemmas (Lemma 4.9 and 4.10). Given the narrow-or-wide structure exposed by the subset selection step, we then *choose values for variables in* T so that all relevant gates in the circuit C simultaneously simplify after applying the resulting restriction ρ. There are two versions, matching our two theorems:

- In the APX_1 setting of Theorem 1.6, we consider a *random assignment of bits* (Lemma 4.9) and then fix a good selection of the values via APX_1's "pointwise-to-global" *averaging argument for expectation* explained above. This lets us form a partial restriction ρ and obtain a corresponding circuit $C \restriction \rho$ that approximately retains the relative advantage of C when computing parity.
- In the more constrained PV_1 setting of Theorem 1.7, we *derandomize* the same choice (Lemma 4.10). Again, this is implemented by a potential argument that feasibly simulates the method of conditional expectations within PV_1. A crucial aspect of the proof that facilitates the formalization is that the relevant expectations depend on at most $O(\log n)$ input coordinates and thus can be efficiently computed by PV_1 terms.

In both settings, the circuit lower bound is obtained by an inductive application of the restriction technique, as in the standard proof of the result. The details appear in Section 4.4.

The novelty is not in the combinatorics of AC^0 versus Parity but in the underlying *proof-theoretic framework*. APX_1 supplies a minimal yet sufficient probabilistic infrastructure that lets us carry out the average-case lower bound argument internally. In our proofs, this lets us define and reason about the agreement tester T_C in the statement of Theorem 1.6, define and analyze appropriate events, quantify advantage and pass from randomized restrictions to concrete choices, and keep track of the small additive losses accumulated across iterations — all within APX_1.

Moreover, this streamlined setup and the perspective it provides clarify the boundary with the weaker theory PV_1. The APX_1 framework and our formalization isolate exactly where probabilistic reasoning is used and where the argument is purely combinatorial. This separation indicates which components can be replaced by deterministic potential-based arguments available in PV_1, thereby guiding the adaptation that yields our worst-case formalization in PV_1.

1.3.3 The Witnessing Theorem (Theorem 1.9)

Theorem 1.9 states that every $\forall\Sigma_1^b(\mathsf{PV})$-sentence provable in APX_1 admits a deterministic polynomial-time Turing reduction to the total search problem $\mathsf{Refuter}(\mathsf{Yao})$, with parameters obeying $(\delta^2/10)\cdot m \geq s + \lceil \log n \rceil + 1$. Recall that an instance of $\mathsf{Refuter}(\mathsf{Yao})$ gives a *predictor generator* G and asks for a flat distribution $\mathcal{D}$ such that the predictor $(i, P) = G(\mathcal{D})$ fails to predict the i-th bit of $\mathcal{D}$ with advantage δ. (For the stated parameter range, $\mathsf{Refuter}(\mathsf{Yao})$ is in TFZPP and map-reduces to $\mathsf{LossyCode}$.)

Suppose that $\mathsf{APX}_1 \vdash \forall x\, \exists y\, \varphi(x, y)$, where φ is a quantifier-free PV-formula. Given x of length n, we describe a predictor generator G_x such that a solution $\mathcal{D}$ to $\mathsf{Refuter}(\mathsf{Yao})$ over G_x allows us to compute y such that $\varphi(x, y)$ holds.

Starting from an APX_1-proof of $\forall x\, \exists y\, \varphi(x, y)$, we first apply Herbrand's theorem over the universal axiomatization of APX_1 to obtain finitely many $\mathsf{PV}(\mathsf{P})$-terms $t_1, \ldots, t_c$ such that $\bigvee_i \varphi(x, t_i(x))$ holds, and encode this disjunction by a single term t_φ with the equational core APX proving $t_\varphi(x, t_1(x), \ldots, t_c(x)) = 1$. In standard models (Section 1.2.1), these terms are polynomial-time oracle algorithms for the approximate-counting oracle P.

Next, we describe the construction of G_x. Given a candidate flat distribution $\mathcal{D}$ (say, over n'-bit strings and of support size m', where $n', m' = \mathsf{poly}(n)$ are large enough), we *simulate* each oracle call $\mathsf{P}(C, \Delta)$ inside any t_i by *empirical counting on* $\mathcal{D}$:

$$\mathsf{P}(C, \Delta) \triangleq \Pr_{u \leftarrow \mathcal{D}} \big[C(u_{\leq \ell}) \big],$$

where $C \colon \{0,1\}^\ell \to \{0,1\}$ and $\ell \leq n'$. Let $t_i^{\mathcal{D}}(x)$ and $t_\varphi^{\mathcal{D}}(x, \cdot)$ denote the resulting outputs. Note that these can be computed in *deterministic* polynomial time, since $\mathcal{D}$ is explicitly given as a collection of $\mathsf{poly}(n)$ strings of length $\mathsf{poly}(n)$, and t_φ, t_1, …, t_c run in polynomial time.

Predictor Extraction Lemma (Lemma 5.6). The key technical step says: if under this simulation the APX-provable equation fails, i.e., $t_\varphi^{\mathcal{D}}(x, \cdot) \neq 1$, then we can algorithmically extract a small predictor P of size s' that achieves advantage at least δ' for an explicitly computed bit position of $\mathcal{D}$, where $(\delta'^2/10)\cdot m' \geq s' + \lceil \log n' \rceil + 1$.

Conceptually, an APX proof asserts: for *every* interpretation of P, either the equation holds or one of the approximate-counting axioms (*Basic*, *Boundary*, *Precision Consistency*, *Local Consistency*) is violated. Under our empirical interpretation, the first three axioms continue to hold, so any failure must exhibit a *Local Consistency* violation of the form

$$\Big| \Pr_{u \leftarrow \mathcal{D}}[C(u_{\leq \ell})] - \tfrac{1}{2}\Big(\Pr_{u \leftarrow \mathcal{D}}[C(u_{<\ell}0)] + \Pr_{u \leftarrow \mathcal{D}}[C(u_{<\ell}1)] \Big) \Big| \; > \; \tfrac{2}{|\Delta|} + \tfrac{1}{|B|},$$

for a circuit C and strings Δ, B produced in the APX proof.

Similarly to the analysis of Yao's distinguisher-to-predictor lemma, such a gap yields a predictor for the next bit via a deterministic transformation of C; here the "signal" comes not from distinguishing $\mathcal{D}$ from uniform, but from detecting a *local inconsistency* of empirical counts across bit-fixings. Thus predictors arise not only from distinguishers, but also from the ability to spot local inconsistencies when $\mathcal{D}$ is used as a random source for approximate counting – a viewpoint that might be of independent interest.[10]

[10]In particular, Yao's distinguisher to predictor transformation requires *randomness* (unless we have $\mathsf{prBPP} = \mathsf{prP}$) [LPT24]. The construction of predictors from local inconsistency, however, is deterministic.

Formally, the lemma is established by a *proof-theoretic analysis*, proceeding by induction on the steps of the APX proof and tracking the parameters (n', m', s', δ') through the final rule used.

Wrapping up the argument, for an input x, the reduction outputs the predictor generator G_x obtained from Lemma 5.6. Note that any solution $\mathcal{D}$ to the resulting Refuter(Yao) instance *cannot* trigger a successful predictor extraction, since by definition G_x fails to produce a predictor on $\mathcal{D}$. Hence given a solution $\mathcal{D}$ to this instance of Refuter(Yao), it must make the simulated identity true, i.e.,

$$t_\varphi(x, t_1^{\mathcal{D}}(x), \ldots, t_c^{\mathcal{D}}(x)) = 1.$$

In other words, some $t_i^{\mathcal{D}}(x)$ is a valid witness y for $\varphi(x, y)$. Finally, as observed above, because $\mathcal{D}$ is explicit, all simulated calls and $t_i^{\mathcal{D}}(x)$ are computable in deterministic polynomial time.

This completes the sketch of the proof of Theorem 1.9. For the details, see Section 5.

1.4 Related Work

Below we provide a representative, though not exhaustive, list of related developments and references.

Probabilistic arguments in bounded arithmetic. Paris, Wilkie, and Woods [PWW88] (see also Pudlák [Pud90]) observed that many probabilistic arguments can be formalized using variants of the weak pigeonhole principle rather than exact counting. An early explicit link between the weak pigeonhole principle and randomized algorithms is due to Wilkie (cf. [Kra95]), who showed that randomized polynomial-time algorithms witness all $\forall\Sigma_1^b$-consequences of $\mathsf{S}_2^1 + \mathsf{dWPHP}(\mathsf{PV})$.

Ojakian [Oja04] undertakes a general study of how probabilistic methods from combinatorics can be formalized in bounded arithmetic. While such proofs can often be recast as purely counting-based arguments, the naive translation still leaves exponentially many objects to count. The central idea, again, is to use the weak pigeonhole principle to simulate the probabilistic counting argument and thereby avoid this blow-up. The formalizations are carried out in S_2^1 augmented with suitable variants of the pigeonhole principle.

Jeřábek [Jeř04] showed that within PV_1 one can compare the sizes of two bounded P/poly-definable sets by constructing a surjection from one onto the other; he used this to formalize descriptions of algorithms in ZPP and RP. He further showed [Jeř04, Jeř05] that $\mathsf{APC}_1 = \mathsf{PV}_1 + \mathsf{dWPHP}(\mathsf{PV})$ is strong enough to formalize sophisticated derandomization results. In [Jeř07a], Jeřábek developed a more systematic framework, showing in particular that for any bounded P/poly-definable set, APC_1 proves that a suitable pair of surjective counting functions exists that approximates its cardinality up to a polynomially small error. (The notation APC_1 follows the terminology of [BKT14].)

Built on Jeřábek's framework, Lê [Lê14] formalizes more results in APC_1 and its extensions, including randomized matching algorithms, the Lovász Local Lemma, and the Goldreich-Levin theorem. Throughout these formalizations, Lê provides formulations of concepts in APC_1 such as *expectation*, *Markov inequalities*, and *pairwise-independence*, which we also consider in this project. The formalization of random variables and expectation in [Lê14] heavily relies on the machinery of APC_1 and is thus inadequate for our purposes.

Remark 1.13. A concrete open problem is to state and prove a stronger form of the Chernoff bound in APX_1. In this work, we show that the Chernoff bound with $O(\log n)$ variables (i.e., strings of length n are considered feasible) can be formalized in APX_1. It is unclear whether we could state a clean and meaningful Chernoff bound with $O(n)$ variables: the error probability will be exponentially small, which could be much smaller than the approximate counting error of the function P. Moreover, even if a meaningful formalization exists, it is unclear whether existing proofs of the Chernoff bound can be formalized in APX_1. Note that a strong form of Chernoff bound with $O(n)$ variables can be formalized in Jeřábek's theory APC_1 (see [Jeř07a, Proposition 2.18]).

The proof complexity of dWPHP(PV). There is evidence that the pigeonhole-based axioms used throughout these frameworks exceed what can be proved in purely polynomial-time theories: while dWPHP(PV) is available in T_2^2, relativized variants are unprovable already in S_2^2 [Rii93]. As noted above, under cryptographic assumptions, PV_1 does not prove dWPHP(PV) [ILW23]. This supports the common stance that PV_1

is too weak to derive the WPHP-style principles exploited in the above formalizations. For a comprehensive investigation of dWPHP(PV) and its provability in bounded arithmetic, see [Kra25].

Jeřábek's approximate-counting toolbox includes general principles such as inclusion-exclusion and strong Chernoff-type estimates, all formalized inside APC_1. By contrast, the development of APX_1 deliberately starts from weaker primitives: while we recover Markov/Chebyshev-style reasoning, error reduction, and other basic probabilistic tools, we do not reprove all of Jeřábek's strongest concentration bounds here. It remains an interesting direction to test the limits of APX_1: which stronger probabilistic inequalities (e.g., full-strength Chernoff) are intrinsically beyond its axioms?

Beyond approximate counting with additive error. Certain combinatorial proofs—e.g., of Ramsey's theorem—typically require counting *sparse sets*, which is unavailable both in our framework and in Jeřábek's theory APC_1. In our setting, for $X \subseteq \{0,1\}^n$ we can estimate $|X|$ only to within an additive error that is a polynomial fraction of 2^n, whereas these arguments require accuracy within a polynomial fraction of $|X|$. Such counting becomes possible in theories stronger than APC_1, as developed in [Jeř09].

In a concurrent work, Thapen [Tha24] introduces a framework to formulate stronger complexity classes (such as $\oplus\mathsf{P}$ and $\#\mathsf{P}$) in the theory of TFNP in a way that is similar in spirit to our axiomatization of approximate counting. For instance, given an oracle that is intended to compute $\oplus\mathsf{P}$, Thapen considered the relativized TFNP problem that searches for a "local inconsistency" of the oracle. Note that the TFNP framework in [Tha24] considers only *query complexity*, while we additionally consider proofs in bounded arithmetic. Nevertheless, it is conceivable that results in these two directions may have analogues in each framework given the similarity in the setup.

Theories with explicit counting. In [Jeř05, Chapter 6], Jeřábek studies bounded theories with explicit counting, revisiting the Impagliazzo-Kapron [IK06] second-order logic for formalizing cryptographic reasoning. The logic is multi-sorted: first-order variables range over strings, while second-order variables of sort $k > 0$ range over k-ary (intended polynomial-time) functions. In particular, functions are second-order objects rather than function symbols in the language. The theory includes recursive counting constructs for expressing the sizes of definable bounded sets.

The same chapter also introduces a feasible theory of approximate counting using a 3-valued semantics based on Kleene's logic, equipped with an LPF-style implication to support induction-like reasoning. Counting is approximate: the semantics distinguishes between having many versus few solutions (while using 3-valued logic to allow an explicit indeterminate region). The resulting "Σ_1^c-consequences" admit probabilistic polynomial-time witnessing (see [Jeř05, Theorem 6.2.20]).

The counting framework in these theories is considerably different from ours, relying on exact counting terms or approximate counting quantifiers in different logical settings. We refer to these references for details.

Bounded reverse mathematics. Cook and Nguyen [CN10] provide a thorough exposition of the bounded reverse mathematics program, systematically developing theories of bounded arithmetic and presenting formalizations of key combinatorial and algorithmic results, with the goal of identifying the weakest axioms sufficient to prove them.

Finally, we refer to [Lê14, Pic14, MP20, AT25] and references therein for numerous examples of results from theoretical computer science that can be formalized in bounded arithmetic. It would be interesting to further investigate which of these formalizations can be carried out in APX_1.

Acknowledgements. We would like to thank Jan Krajíček for discussions related to the $\forall\Sigma_1^b$-conservativity of APC_1 over PV_1 and for bringing some references to our attention. We also thank Dimitrios Tsintsilidas for comments on an earlier version of the paper. We would also like to thank Surya Mathialagan, Shuo Pang, and Hanlin Ren for helpful discussions. Finally, we thank the anonymous STOC reviewers for useful comments about the presentation.

2 Formal Definition of the Theory

In this section, we formally define the equational theory APX and its first-order counterpart APX_1. We assume basic familiarity with Cook's Theory PV [Coo75]. The necessary background can be found in [Kra95, Chapter 12], [Kra19, Chapter 12], and [Li25].

2.1 Notation

Base Theory. Let $\mathsf{PV}(\mathsf{P})$ be the theory PV relative to a fresh function symbol P with the axiom

$$\mathsf{ITR}(\mathsf{P}(C,\Delta), C \# \Delta \# \Delta) = \varepsilon$$

that bounds the output length of the function symbol P. Intuitively, the axiom means that the output length of $\mathsf{P}(C,\Delta)$ given strings C and Δ as its input is at most $|C| \cdot |\Delta|^2$. This axiom ensures that $\mathsf{PV}(\mathsf{P})$-terms are feasible functions in the standard model. Interested readers are referred to [Jeř04, Jeř07a, Jeř07b] for more examples of relativized PV.

Slightly different from Cook's original notation, we will define $\mathsf{PV}(\mathsf{P})$ with constant symbol ε (rather than 0) and replace the initial functions $s_1(x)$ and $s_2(x)$ by $s_1(x)$ and $s_0(x)$, respectively. Other functions TR, ITR, $\circ$, and $\#$ are defined as in Cook's original definition. Let $\hat{\mathsf{P}}$ be a function over Boolean strings. The standard model of $\mathsf{PV}(\mathsf{P})$ with respect to $\hat{\mathsf{P}}$, denoted by $\mathbb{M}(\hat{\mathsf{P}})$, is defined as follows:

- The universe consists of all Boolean strings of finite length.
- The constant symbol ε is interpreted as the empty string.
- $s_b(x)$ (for $b \in \{0,1\}$) is interpreted as the function that appends b to the right of the string x.
- $\mathsf{TR}(x)$ is interpreted as the function that trims the rightmost bit of x; $\mathsf{ITR}(x,y)$ is interpreted as the function that trims x for $|y|$ times.
- $\circ$ is interpreted as string concatenation, while $\#(x,y)$ is interpreted as the function that concatenates $|y|$ copies of x.
- The function symbol P is interpreted as $\hat{\mathsf{P}}$.

A function introduced by one of the rules in PV (i.e. introduction by terms or introduction by limited recursion on notation) is interpreted as the unique function over the universe that satisfies its introduction rule.

Circuits. We define a few PV functions that manipulate Boolean circuits. Let $\mathsf{IsCkt}(C,z)$ be the PV function that outputs 1 if C is a circuit with input length $|z|$, and outputs 0 otherwise; $\mathsf{IsConst}(C)$ be the PV function that outputs 1 if C is a circuit that does not read its input (i.e., there is no path from the output gate to an input variable); $\mathsf{Bool}(C)$ outputs 1 if $\mathsf{IsConst}(C)$ and C outputs 1 and outputs 0 if $\mathsf{IsConst}(C)$ and C outputs 0 (otherwise, outputs, e.g., ε); $\mathsf{Fix}_b(C)$ be the function that, given a circuit C, output the circuit obtained from C by fixing the rightmost input bit to be $b \in \{0,1\}$; $\mathsf{Eval}(C,x)$ be the function that evaluate the circuit C on the input x. One may think of any straightforward implementations of these functions in PV as PV is a robust theory.

For simplicity, we use the following abbreviations:

- For $n \in \mathsf{Log}$, $C \in B_n$ denotes $\mathsf{IsCkt}(C, 1^n)$, i.e., C is a circuit with n input bits. Moreover, $\forall C \in B_n\ \varphi(C)$ denotes $\forall C\ (\mathsf{IsCkt}(C,1^n) \to \varphi(C))$ and $\exists C \in B_n\ \varphi(C)$ denotes $\exists C\ (\mathsf{IsCkt}(C,1^n) \land \varphi(C))$.
- For a circuit C, $C(x)$ denotes $\mathsf{Eval}(C,x)$.
- We use Null_n to denote the circuit with n input bits that does not read its input bits and outputs 0, and True_n to denote the circuit with n input bits that does not read its input bits and outputs 1.

PV-Terms and Functions. We say that a $\mathsf{PV}(\mathsf{P})$-term is a PV-term if its construction indicates that it does not call the P-oracle. Formally, the set of PV-terms is the minimum set that contains all base functions and is close under composition and the function formulation rules in PV, that is:

- Base functions $s_i(x)$, $\mathsf{TR}(x)$, $\mathsf{ITR}(x)$, $\circ(x, y)$ are PV-terms.
- If t is a PV-term, the function f_t introduced with the defining axiom $f_t = t$ is also a PV term.
- A term formulated from PV-terms by composition is a PV term.
- If g, h_0, h_1, k_0, k_1 are PV-terms, the function f_Π constructed by limited recursion on notation from $\Pi = (g, h_0, h_1, k_0, k_1)$ is a PV term.

We say that a function symbol f is a PV-function if it is a PV-term.

By the Cook-Levin theorem (see [Pic15b] for a formalization in PV), PV terms can be converted into polynomial-size Boolean circuits on any given input length $n \in \mathsf{Log}$, and the correctness can be proved in PV. Similarly, $\mathsf{PV}(\mathsf{P})$ terms can be converted into polynomial-size P-oracle circuits on any given input length $n \in \mathsf{Log}$ with $\mathsf{PV}(\mathsf{P})$-provable correctness.

Encoding Conventions and Arithmetic Operations. For functions and multi-output circuits, we will treat ε as `false` and any other value as `true` when we define the acceptance probability of the circuit. Let $\mathsf{Bool}(x)$ be the PV function that outputs 0 if x is ε and outputs 1 otherwise.

We assume that natural numbers are encoded in binary in a straightforward way. For instance, one can encode a natural number in dyadic notation as in [Coo75] so that basic arithmetic operations such as addition, multiplication, and comparison can be defined naturally. We assume that the encoding can be verified efficiently, i.e., there is a PV function symbol $\mathsf{IsNumber}(x)$ that outputs 1 if x is the encoding of a natural number, and outputs 0 otherwise, and use $[x \in \mathbb{N}]$ as the shorthand of $\mathsf{IsNumber}(x)$. We use $[x]_\mathbb{N}$ to denote the natural number encoded by x when we want to be explicit about the interpretation of x as a natural number.

Elementary arithmetic operations, such as addition, multiplication, and comparison, can be defined naturally. Moreover, basic properties of the PV function symbols representing these operations can be established in PV (whenever the operations involve a feasible number of elements).

We specify a standard encoding of rational numbers in PV: We use the pair (x, y) to denote the rational number

$$\sum_{i=1}^{|x|} x_i 2^{i-1} + \sum_{j=1}^{|y|} y_j 2^{-j}.$$

Similarly to the encoding of natural numbers, we assume a PV function symbol $\mathsf{IsRational}(x)$ that tests whether x encodes a rational number, and we use $[x \in \mathbb{Q}]$ as shorthand for $\mathsf{IsRational}(x)$. We write $[x]_\mathbb{Q}$ to denote the rational number encoded by x. We might directly treat x as a rational number if this is clear from the context.

Data Structures and Explicit Sets. We assume a straightforward encoding of *explicit* sets (and multi-sets), i.e., sets of feasible size, that supports operations such as selection, union, intersection, and membership query. An explicit set S may be encoded as a list containing all the elements in it. Note that this is different from the *feasibly definable* sets in [Jeř04], which may be of infeasible size. When discussing explicit sets, we use $|S|$ to denote the size of S, i.e., the number of elements contained in S. When $S = \{q_1, \dots, q_\ell\}$ is a set of rational numbers, we use $||S|| = \sum_{i=1}^{\ell} |q_i|$ to denote its ℓ_1-norm, i.e., the sum of the absolute values of the elements in S.[11]

Moreover, for an explicit set S and a quantifier-free formula $\varphi(x)$ in the language of PV_1, we can define the universal quantification over S, denoted by $\forall x \in S : \varphi(x)$, as a *quantifier-free* formula in PV_1 that is true if and only if every element $x \in S$ satisfies $\varphi(x)$ (in the standard model). This is possible as S is explicitly encoded, and thus there is a straightforward feasible algorithm that given the encoding of S, enumerates S and checks whether there is an $x \in S$ such that $\varphi(x)$ is false. Similarly, we can define the existential quantification over S, denoted by $\exists x \in S : \varphi(x)$. All relevant deduction rules about quantification over sets

[11] While we abuse notation and employ $|\cdot|$ to denote both length and absolute value, the meaning will be clear in each context.

should be admissible in PV assuming standard encoding, e.g.,

$$(\exists_i): \quad \frac{\Gamma \vdash \varphi[y/t] \quad \Gamma \vdash t \in S}{\Gamma \vdash \exists y \in S : \varphi} \tag{2.1}$$

$$(\exists_e): \quad \frac{\Gamma \vdash \exists y \in S : \varphi \quad \Gamma, z \in S, \varphi[y/z] \vdash \psi}{\Gamma \vdash \psi} \tag{2.2}$$

where in $(\exists_i)$ t is an arbitrary term, and in $(\exists_e)$ z must be a fresh variable that has no occurrence in Γ, φ, ψ, y. This ensures that most natural mathematical proofs regarding explicit sets can be easily formalized in PV; see [Li25, Chapter 4] for more discussions. In the rest of the paper, we will only informally describe the proof and pinpoint the key idea to formalize it in PV if it is unclear.

2.2 Theory APX

Intuitively, we will define the theory as PV(P) together with additional axioms intended to formalize that P approximately computes the acceptance probability of a given circuit up to a specified precision. In other words, for every deterministic circuit C and any Δ, $\mathsf{P}(C,\Delta)$ outputs the encoding of a rational number in $[0,1]$ guaranteed to lie within the interval $[p-1/|\Delta|, p+1/|\Delta|]$, where p denotes the acceptance probability of C. For simplicity, we will also denote $\mathsf{P}(C,\Delta)$ by $\mathsf{P}_\delta(C)$, where $\delta^{-1} \triangleq |\Delta| \in \mathsf{Log}$ is the precision of counting.

Language of APX. APX is an equational theory whose language extends that of PV by including the new function symbol P and every additional function symbol that can be introduced through the usual function symbol introduction rules of PV (including composition and limited recursion on notation).

Although PV is an equational theory operating over strings, propositional connectives, arithmetic operations, and arithmetic relations (e.g., comparison between rational numbers) can be encoded by appropriate equations with desired properties (see, e.g., [Li25]). This allows us to formulate the following axioms.

Axioms of APX. The axioms involve only universally quantified variables and can therefore be expressed as PV(P) equations:

- (*Basic Axiom*). Any provable equation in PV(P) is an axiom of APX. Moreover, $[\mathsf{P}(C,\Delta) \in \mathbb{Q}] = 1$, $\mathsf{P}(C,\Delta) \le 1$, $0 \le \mathsf{P}(C,\Delta)$ are axioms of APX, where "$x \le y$" is formalized by an appropriate PV equation that is valid if and only if $[x]_\mathbb{Q} \le [y]_\mathbb{Q}$.
- (*Boundary Axiom*). For any $C \in B_n$, $\mathsf{IsConst}(C) \to \mathsf{P}_\delta(C) = \mathsf{Bool}(C)$. This axiom indicates that the acceptance probability of a syntactically constant circuit[12] that always outputs $b \in \{0,1\}$ is equal to b.
- (*Precision Consistency Axiom*). For every $n, \delta_1^{-1}, \delta_2^{-1}, \beta^{-1} \in \mathsf{Log}$ and every $C \in B_n$,

$$|\mathsf{P}_{\delta_1}(C) - \mathsf{P}_{\delta_2}(C)| \le \delta_1 + \delta_2 + \beta. \tag{2.3}$$

  Intuitively, this axiom states that the approximate counting function P should be consistent with different precision parameters.
- (*Local Consistency Axiom*). For every $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and every $C \in B_n$,

$$\left|\mathsf{P}_\delta(C) - \frac{\mathsf{P}_\delta(\mathsf{Fix}_0(C)) + \mathsf{P}_\delta(\mathsf{Fix}_1(C))}{2}\right| \le 2 \cdot \delta + \beta. \tag{2.4}$$

  Intuitively, this axiom states that the approximate counting function P should be self-consistent in the sense that the acceptance probability of a circuit C is close to the average acceptance probability of the circuit obtained by randomly fixing the rightmost input bit of C.

[12] In other words, there is no path from the output gate to an input variable, i.e., the relevant part of the circuit consists of Boolean operations applied to constant input bits.

Rules of APX. Finally, the theory contains the following derivation rules:

- (*Logical Rules*). We include the logical rules of PV:
  1. $t_1 = t_2 \vdash t_2 = t_1$
  2. $t_1 = t_2, t_2 = t_3 \vdash t_1 = t_3$
  3. $t_1 = t_2 \vdash t_1(x/t) = t_2(x/t)$
  4. $u = v \vdash t(x/u) = t(x/v)$
- (*Structural Induction Rule*). Let $f_1(x, \vec{y})$ and $f_2(x, \vec{y})$ be PV(P) functions. For PV(P) functions $g(\vec{y})$, $h_0(x, \vec{y}, z)$, and $h_1(x, \vec{y}, z)$, if the following equations are provable for $j \in \{1, 2\}$ and $i \in \{0, 1\}$

$$f_j(\varepsilon, \vec{y}) = g(\vec{y}) \quad (2.5)$$

$$f_j(s_i(x), \vec{y}) = h_i(x, \vec{y}, f_j(x, \vec{y})) \quad (2.6)$$

  then we can deduce the equation $f_1(x, \vec{y}) = f_2(x, \vec{y})$. This rule is analogous to the original induction rule in PV. Intuitively, it means that if f_1 and f_2 are both identical to the function recursively defined from g, h_0, h_1, they are the same function.

Remark 2.1 (Nested Probability Symbols). We stress that the function symbol P does not take P-oracle circuits as input, and therefore sentences involving nested probability symbols such as

$$\Pr_x[\Pr_y[\varphi(x, y)] > \varepsilon] > \delta$$

cannot be formalized directly in APX. Nevertheless, for predicates $\varphi(x, p)$ and $\psi(y)$ that do not share inputs, the nested probability

$$\Pr_x\left[\psi\left(x, \Pr_y[\varphi(y)]\right)\right]$$

can be expressed in APX, as we can define a P-oracle algorithm that first calculates $p = \Pr_y[\varphi(y)]$ by calling the P-oracle, and then calculates $\Pr_x[\psi(x, p)]$ by calling the P-oracle again.

Remark 2.2 (Elementary Functions and Precision Issues). In our formalizations, we sometimes employ elementary functions over the reals, such as $\sqrt{x}$, $\ln(x)$, or $\exp(x)$ (typically involving constants or for an x of the form a/b with $a, b \in \mathsf{Log}$). As the output of these functions may be an irrational number, to implement them in PV, we need to define each function by taking an additional parameter that determines the number of digits of precision. When the functions are defined appropriately, appropriate formulations of basic inequalities (e.g., $\exp(x) \geq 1 + x$) can be proved in PV by directly formalizing a standard mathematical proof. In this paper, in the context of the use of such values and inequalities, we always have a margin to tolerate any potential precision issue (e.g., the error term $\beta \in \mathsf{Log}^{-1}$ in axioms). For this reason, and following standard practice, we will not elaborate on the actual implementation of such functions and their basic properties.

2.3 Models of APX

Let $\mathbb{M}$ be the standard model of PV, i.e., the universe is $\{0, 1\}^*$, ε is interpreted as the empty string, and $s_0(x)$, $s_1(x)$ are interpreted as the functions that append 0 and 1 to x, respectively. For every function $\hat{\mathsf{P}} : \{0, 1\}^* \times \{0, 1\}^* \to \{0, 1\}^*$, $\mathbb{M}(\hat{\mathsf{P}})$ is the model of PV(P) where the function symbol P is interpreted as the function $\hat{\mathsf{P}}$.

Definition 2.3 (Standard Models)**.** Let $\hat{\mathsf{P}} \colon \{0, 1\}^* \times \{0, 1\}^* \to \{0, 1\}^*$ be any *correct approximate counting function*, i.e.,

1. $\hat{\mathsf{P}}(C, \Delta)$ outputs (the encoding of) a rational number $q \in [0, 1]$ within the interval $[p - 1/|\Delta|, p + 1/|\Delta|]$ for every circuit $C : \{0, 1\}^* \to \{0, 1\}$ and $\Delta \in \{0, 1\}^*$, where p is the acceptance probability of C and q is of length at most $|C| \cdot |\Delta|^2$; and

2. $\hat{\mathsf{P}}(C,\Delta)$ outputs the correct value in $\{0,1\}$ whenever the input circuit C satisfies $\mathsf{IsConst}(C)$.

We say that $\mathbb{M}(\hat{\mathsf{P}})$ is *a standard model* of APX.

Definition 2.4 (Admissible Models)**.** Let $\hat{\mathsf{P}}\colon \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ be a function. We say that $\mathbb{M}(\hat{\mathsf{P}})$ is *an admissible model* of APX if it satisfies all axioms and rules of APX.

The crucial observation is that a model is standard if and only if it is admissible. The proof of the theorem is highly constructive; indeed, similar induction arguments occur multiple times in the development of basic probability theory in APX_1 (see Section 3).

Theorem 2.5. *Let $\hat{\mathsf{P}} : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ be any function. Then $\mathbb{M}(\hat{\mathsf{P}})$ is a standard model if and only if it is an admissible model.*

Proof. We will only prove the ($\Leftarrow$) direction, as the converse is straightforward. Suppose, towards a contradiction, that $\mathbb{M}(\hat{\mathsf{P}})$ is admissible but is not standard. Then there is a circuit $C : \{0,1\}^n \to \{0,1\}$ and $\Delta \in \{0,1\}^*$ such that

$$\hat{\mathsf{P}}(C,\Delta) \notin p(C) \pm \frac{1}{|\Delta|}, \tag{2.7}$$

where $p(C) \triangleq \Pr_{x\in\{0,1\}^n}[C(x)=1]$ is the acceptance probability of C. Note that C and Δ are encoded by finite strings (in the standard model $\mathbb{M}$ of PV), and $n \in \mathbb{N}$.

As $\mathbb{M}(\hat{\mathsf{P}})$ is admissible, it must satisfy the Boundary Axiom. Subsequently, Equation (2.7) does not hold when $n = 0$. It suffices to consider the case that $n > 0$. Suppose for contradiction we have

$$\left|\hat{\mathsf{P}}(C,\Delta) - p(C)\right| > \frac{1}{|\Delta|} + \varepsilon, \tag{2.8}$$

where $\varepsilon > 0$. Let $\Xi \triangleq 1^{10(n+1)/\varepsilon}$. As $\hat{\mathsf{P}}$ satisfies the Precision Consistency Axiom, we have

$$\left|\hat{\mathsf{P}}(C,\Xi) - p(C)\right| \geq \left|\hat{\mathsf{P}}(C,\Delta) - p(C)\right| - \left(\frac{1}{|\Delta|} + \frac{2}{|\Xi|}\right) > \varepsilon - \frac{2}{|\Xi|}. \tag{2.9}$$

Let C_0, C_1 be the circuits obtained by fixing the rightmost input bit of C to be 0 and 1, respectively. As $\hat{\mathsf{P}}$ must satisfy the Local Consistency Axiom, we have

$$\left|\hat{\mathsf{P}}(C,\Xi) - \frac{\hat{\mathsf{P}}(C_0,\Xi) + \hat{\mathsf{P}}(C_1,\Xi)}{2}\right| \leq \frac{3}{|\Xi|}, \tag{2.10}$$

Also, $p(C) = (p(C_0) + p(C_1))/2$ by its definition. Subsequently, there exists $\sigma \in \{0,1\}$ such that

$$\left|\hat{\mathsf{P}}(C_\sigma,\Xi) - p(C_\sigma)\right| \geq \left|\hat{\mathsf{P}}(C,\Xi) - p(C)\right| - \frac{3}{|\Xi|}. \tag{2.11}$$

Recall that $n \in \mathbb{N}$ is a standard integer. Let $C^{(0)} \triangleq C$, and $C^{(1)} \triangleq C_\sigma$. By the procedure defined above, for every $1 \leq i \leq n$, we can define $C^{(i)}$ as the circuit obtained from $C^{(i-1)}$ by fixing the rightmost input bit such that

$$\left|\hat{\mathsf{P}}(C^{(i)},\Xi) - p(C^{(i)})\right| \geq \left|\hat{\mathsf{P}}(C^{(i-1)},\Xi) - p(C^{(i-1)})\right| - \frac{3}{|\Xi|},$$

and therefore by Equation (2.9) we eventually have

$$\left|\hat{\mathsf{P}}(C^{(n)},\Xi) - p(C^{(n)})\right| \geq \varepsilon - \frac{6\cdot(n+1)}{|\Xi|} > 0.$$

Note that the circuit $C^{(n)}$ has input length 0 and as a consequence computes a constant function. The value $p(C^{(n)}) \in \{0,1\}$ is its acceptance probability. Since $\mathbb{M}(\hat{\mathsf{P}})$ is admissible, it satisfies the Boundary Axiom, and consequently $\hat{\mathsf{P}}(C^{(n)},\Xi) = p(C^{(n)})$. This contradicts the above inequality. $\square$

Definition 2.6. Among the standard models of APX, the one that interprets P by the exact counting function is called the *exact standard model* of APX, denoted by $\mathbb{M}^*$.

Proposition 2.7 (Soundness of APX)**.** *Provable equations in* APX *are true in any standard model of* APX.

Proof. This can be verified by induction on the proof. ☐

2.4 First-Order Theory APX_1

In analogy with the first-order theory PV_1, we will introduce a first-order theory APX_1 that includes all APX provable equations as well as convenient deduction rules.

Language of APX_1. The language of the first-order theory APX_1 includes all $\mathsf{PV}(\mathsf{P})$ symbols.

Axioms of APX_1. The theory is axiomatized by the standard first-order logic with equality together with the following non-logical axiom schemes:

- For any provable equation $s(\vec{x}) = t(\vec{x})$ of APX, $\forall\vec{x}\ s(\vec{x}) = t(\vec{x})$ is an axiom of APX_1.
- $\forall x\ \forall y\ (x = y \leftrightarrow s_i(x) = s_i(y))$, $i \in \{0,1\}$, is an axiom of APX_1.
- $\forall x\ \varepsilon \neq s_i(x)$ is an axiom of APX_1.
- $\forall x\ s_0(x) \neq s_1(x)$ is an axiom of APX_1.
- (n-induction). Let φ be a quantifier-free formula and $x_1, \ldots, x_n, \vec{y}$ be variables. Suppose that φ does not contain free variables other than $x_1, \ldots, x_n$ and $\vec{y}$. Then

$$\forall\vec{y}\ \left(\bigwedge_{j\in[n]} \varphi(x_j/\varepsilon) \wedge \forall\vec{x}\left(\bigwedge_{\vec{\sigma}\in\{0,1\}^n} (\varphi \to \varphi_{\vec{\sigma}})\right) \to \forall\vec{x}\ \varphi\right),$$

  where $\varphi_{\vec{\sigma}}$ denotes the formula φ with all free occurrences of x_i substituted by $s_{\sigma_i}(x_i)$ for each $i \in [n]$, is an axiom of APX_1.

We observe that APX_1 satisfies the following properties.

Proposition 2.8. APX_1 *admits a universal axiomatization.*

Proof Sketch. This is essentially the same as the proof that PV admits a universal axiomatization (see [Coo75, Kra95]). Note that all axioms of APX_1 are universal sentences except for the n-induction axiom scheme, which is a $\forall\exists$-sentence. In more detail, the n-induction axiom scheme is logically equivalent to the following sentence: For every $\vec{y}$ and $\vec{x} = (x_1, \ldots, x_n)$ satisfying that

- $\varphi(x_j/\varepsilon)$ for every $j \in [n]$, and
- $\neg\varphi$,

there exists an $\vec{x}' = (x'_1, \ldots, x'_n)$ and $\vec{\sigma} = (\sigma_1, \ldots, \sigma_n) \in \{0,1\}^n$ such that $\varphi(\vec{x}/\vec{x}')$ is true but $\varphi(\vec{x}/\vec{x}'_\sigma)$ is false, where $\vec{x}'_\sigma \triangleq (s_{\sigma_1}(x'_1), \ldots, s_{\sigma_n}(x'_n))$. Nevertheless, there is a straightforward polynomial-time algorithm that outputs such $\vec{x}'$ given $\vec{x}$ and $\vec{y}$ by considering prefixes of $\vec{x}$, and the correctness of the algorithm can be proved in $\mathsf{PV}(\mathsf{P})$. This can be used to show that the n-induction axiom scheme can be derived from other axiom schemes and APX_1 admits a universal axiomatization. ☐

Proposition 2.9. APX_1 *is conservative over* APX.

Proof Sketch. The proof is essentially the same as the proof that PV_1 is conservative over PV. We refer interested readers to [Coo75, Bus86, Kra95] for more details. ☐

Similar to PV_1 (see [KPT91]), we can show that a form of induction principle on quantifier-free formulas is provable in APX_1. In order to state this result, we need to make some remarks about notation.

Recall that we assume a straightforward encoding of natural numbers, such as the dyadic encoding in [Coo75], and use $[x]_\mathbb{N}$ to denote the natural number encoded by x. Let $\mathsf{Less}_\mathbb{N}(x, y)$ be the PV function that outputs 1 if (in the standard model) $[x]_\mathbb{N} < [y]_\mathbb{N}$ and outputs 0 otherwise. We use $[x < y]$ as a shorthand for $\mathsf{Less}_\mathbb{N}(x, y)$. We use $\forall x < y\ \varphi(x)$ as a shorthand for $\forall x\ ([x < y] = 1 \to \varphi(x))$, and $\exists x < y\ \varphi(x)$ as a shorthand for $\exists x\ ([x < y] = 1 \wedge \varphi(x))$.

Theorem 2.10. *Let $\varphi(x, \vec{y})$ be a quantifier-free formula. Then APX_1 proves*

$$\forall \vec{y}\ \forall b\ \ (\varphi(0, \vec{y}) \wedge \forall x < b\ (\varphi(x, \vec{y}) \to \varphi(x+1, \vec{y})) \to \varphi(b, \vec{y}))\,,$$

where 0 is the PV-term encoding $0 \in \mathbb{N}$ and $+$ is the PV-function for addition of natural numbers.

Proof Sketch. The proof is essentially the same as the admissibility proof of such induction scheme in PV_1, following a binary search argument. We refer interested readers to [Coo75, Bus86, KPT91, Kra95] for more details. □

Models of APX_1. Any model $\mathcal{M}(\mathcal{P})$ of APX induces a model of APX_1 with the same universe and interpretation for $\mathsf{PV(P)}$ terms. In particular, a model of APX_1 is said to be a *standard* model if it is derived from any standard model of APX. A first-order sentence φ in the language of APX_1 is said to be a *true sentence* if it is true in *any* standard model. We provide two examples:

- Let $C_{n,m} \equiv 0$ be a constant circuit that takes $(x, y) \in \{0,1\}^n \times \{0,1\}^m$. The sentence

  $$\forall n \in \mathsf{Log}\ \forall m \in \mathsf{Log}\ \forall x \in \{0,1\}^n\ \forall \delta^{-1} \in \mathsf{Log}\ \mathsf{P}_\delta(C(x, \cdot)) \leq 2\delta$$

  in suitable formalization, is a true sentence as it holds when $\mathsf{P}_\delta(\cdot)$ is interpreted as any valid approximate counting oracle with additive error δ.
- For the same circuit $C_{n,m}$, the sentence

  $$\forall n \in \mathsf{Log}\ \forall m \in \mathsf{Log}\ \forall x \in \{0,1\}^n\ \forall \delta^{-1} \in \mathsf{Log}\ \mathsf{P}_\delta(C(x, \cdot)) = 0$$

  is true in the exact standard model, but is not true in the standard model where $\mathsf{P}_\delta(C(x, \cdot)) \triangleq \delta$ and $\mathsf{P}_\delta(D) \triangleq 0$ when $D \neq C(x, \cdot)$. Therefore it is not a true sentence.

Proposition 2.11 (Soundness of APX_1)**.** *Any provable sentence φ in APX_1 is a true sentence.*

Proof. This can be verified by induction on the proof. □

3 Probabilistic Reasoning in APX_1

In this section, we prove meta-theorems that exhibit the robustness of the approximate counting function in APX_1 and develop basic concepts such as (approximate) expectation and variance for feasibly defined random variables.

3.1 Consistency of Approximate Counting

We now state a couple of meta-theorems indicating that the approximate counting functionality provided in APX is consistent in a strong sense.

3.1.1 Global Consistency of Approximate Counting

Monotonicity. Suppose that there are two circuits $C_1, C_2 : \{0,1\}^n \to \{0,1\}$ satisfying that $C_1(x) \le C_2(x)$ for every $x \in \{0,1\}^n$. Then the acceptance probability of $C_1(x)$ is at most that of $C_2(x)$. Therefore, if P is a function for approximate counting, the acceptance probability of C_1 reported by P should be no larger than the reported acceptance probability of C_2 plus twice the precision of counting. Formally:

Lemma 3.1. APX_1 *proves that*

$$\forall n \in \mathsf{Log}\ \forall C_1, C_2 \in B_n\ \forall \delta^{-1} \in \mathsf{Log}\ \forall \beta^{-1} \in \mathsf{Log}$$
$$\big((\forall x \in \{0,1\}^n\ C_1(x) \le C_2(x) \to \mathsf{P}_\delta(C_1) \le \mathsf{P}_\delta(C_2) + 2 \cdot \delta + \beta\big).$$

Proof. We argue in APX_1. Fix $\vec{y}$ and $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and circuits $C_1, C_2 \in B_n$. Suppose that $\forall x\ C_1(x) \le C_2(x)$. We will prove that

$$\mathsf{P}_\delta(C_1) \le \mathsf{P}_\delta(C_2) + 2 \cdot \delta + \beta.$$

Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later, and $C_1^{k,x}, C_2^{k,x} : \{0,1\}^{n-k} \to \{0,1\}$ be the circuits obtained by fixing the rightmost k bits of C_1, C_2 by $x \in \{0,1\}^k$, respectively.

We will prove that $\mathsf{P}_\eta(C_1) \le P_\eta(C_2) + 6 \cdot n \cdot \eta$. This suffices as we can pick $\eta = 1/(100 \cdot n \cdot \beta)$ and apply the Precision Consistency Axiom.

Towards a contradiction, assume that $\mathsf{P}_\eta(C_1) > P_\eta(C_2) + 6 \cdot n \cdot \eta$. We will design a P-oracle algorithm that, for any such C_1, C_2 and $k \le n$, outputs a string x of length k that satisfies the invariant

$$\mathsf{P}_\eta(C_1^{k,x}) > \mathsf{P}_\eta(C_2^{k,x}) + 6 \cdot (n-k) \cdot \eta.$$

Moreover, the correctness of the algorithm can be proved in APX_1. The algorithm is an iterative algorithm that considers $k = 0, 1, \ldots, n$:

- For $k = 0$, the algorithm outputs ε. This is correct as $C_1^{0,\varepsilon} = C_1$, $C_2 = C_2^{0,\varepsilon}$, and as a consequence the required statement follows from the assumption that $\mathsf{P}_\eta(C_1) > P_\eta(C_2) + 6 \cdot n \cdot \eta$.
- Now suppose the algorithm could output a string $x \in \{0,1\}^k$ such that

  $$\mathsf{P}_\eta(C_1^{k,x}) > \mathsf{P}_\eta(C_2^{k,x}) + 6 \cdot (n-k) \cdot \eta.$$

  Our goal is to output a string $x' \in \{0,1\}^{k+1}$ such that

  $$\mathsf{P}_\eta(C_1^{k+1,x'}) > \mathsf{P}_\eta(C_2^{k+1,x'}) + 6 \cdot (n-k-1) \cdot \eta.$$

  Note that by the Local Consistency Axiom[13], we know that

  $$\mathsf{P}_\eta(C_1^{k,x}) \ge (1/2) \cdot (\mathsf{P}_\eta(\mathsf{Fix}(C_1^{k,x}, 0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_1^{k,x}, 1))) - 3 \cdot \eta;$$
  $$\mathsf{P}_\eta(C_2^{k,x}) \le (1/2) \cdot (\mathsf{P}_\eta(\mathsf{Fix}(C_2^{k,x}, 0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_2^{k,x}, 1))) + 3 \cdot \eta.$$

  Subsequently, there must be $\sigma \in \{0,1\}$ such that $\mathsf{P}_\eta(\mathsf{Fix}(C_1^{k,x}, \sigma)) > \mathsf{P}_\eta(\mathsf{Fix}(C_2^{k,x}, \sigma)) + 6 \cdot (n-k-1) \cdot \eta$. The algorithm queries the P-oracle, finds such $\sigma \in \{0,1\}$, and outputs $x' \triangleq \sigma \circ x$. This satisfies the invariant, as $\mathsf{Fix}(C_1^{k,x}, \sigma) = C_1^{k,\sigma \circ x}$ and $\mathsf{Fix}(C_2^{k,x}, \sigma) = C_2^{k,\sigma \circ x}$.

It is clear that the correctness of the algorithm can be proved in APX_1 using induction for open formulas.[14] It follows that as we fix $k = n$, the algorithm provably outputs $x^\star \in \{0,1\}^n$ such that $\mathsf{P}_\eta(C_1^{n,x^\star}) > \mathsf{P}_\eta(C_2^{n,x^\star})$, under the assumption that $\mathsf{P}_\eta(C_1) > P_\eta(C_2) + 6 \cdot n \cdot \eta$. Note that $C_1^{n,x}$ and $C_2^{n,x}$ are circuits that do not read their inputs and output $C_1(x^\star)$ and $C_2(x^\star)$, respectively. This violates the Boundary Axiom, since $C_1(x) \le C_2(x)$ for every $x \in \{0,1\}^n$. ☐

[13] Here, the parameter β in the Local Consistency Axiom is set to η.

[14] Formally, we use n-induction for $n = 1$, and rely on the fact that open formulas are expressive enough, i.e., the language of APX_1 includes the function symbol P and oracle polynomial-time functions with access to P.

Global Consistency. A corollary of monotonicity is that if two circuits are provably identical, their acceptance probabilities given by the oracle P should not differ significantly. Formally:

Lemma 3.2. APX_1 *proves that*

$$\forall n \in \mathsf{Log}\ \forall C_1, C_2 \in B_n\ \forall \delta^{-1}, \beta^{-1} \in \mathsf{Log}$$
$$\big((\forall x \in \{0,1\}^n\ C_1(x) = C_2(x)) \to |\mathsf{P}_\delta(C_1) - \mathsf{P}_\delta(C_2)| \le 2 \cdot \delta + \beta\big).$$

Proof. We can prove by Lemma 3.1 that if $C_1(x) = C_2(x)$ for any $x \in \{0,1\}^n$, it follows that $\mathsf{P}_\delta(C_1) - \mathsf{P}_\delta(C_2) \le 2 \cdot \delta + \beta$ and $\mathsf{P}_\delta(C_2) - \mathsf{P}_\delta(C_1) \le 2 \cdot \delta + \beta$. Subsequently, we will have $|\mathsf{P}_\delta(C_1) - \mathsf{P}_\delta(C_2)| \le 2 \cdot \delta + \beta$ as long as the absolute value function is properly defined. □

Remark 3.3. The standard way to formalize approximate counting for a polynomial-time decidable property in APX_1 is to first translate it to a circuit C then query $\mathsf{P}_\delta(C)$. The global consistency property shows that the approximate counting oracle P is robust with respect to the translation of PV functions into circuits, provided that we can prove in APX_1 that the translation is functionally correct. The latter can be done already in PV_1 (see, e.g., [Pic14, Section 2.4]).

3.1.2 Permutational Symmetry of Approximate Counting

Next we show that the approximate counting oracle P is *permutational symmetric*, in the sense that a permutation of input variables does not change the acceptance probability given by P significantly.

Local Symmetry. As a first step, we show that swapping two adjacent input bits of a circuit does not change the acceptance probability significantly. Concretely:

Lemma 3.4. APX_1 *proves that*

$$\forall n \in \mathsf{Log}\ \forall i \in [n-1]\ \forall C \in B_n\ \forall \delta^{-1}, \beta^{-1} \in \mathsf{Log}\ |\mathsf{P}_\delta(C) - \mathsf{P}_\delta(\mathsf{Swap}(C,i))| \le 2 \cdot \delta + \beta,$$

where $\mathsf{Swap}(C,i)$ *is a* PV*-function that outputs a circuit* C' *obtained by swapping the* i*-th and the* $(i+1)$*-th input bits (from the rightmost bit) of* C*.*

Proof. We argue in APX_1. Fix $n \in \mathsf{Log}$, $i \in [n-1]$, $C \in B_n$, $\delta^{-1} \in \mathsf{Log}$, and $\beta^{-1} \in \mathsf{Log}$. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later.

For $C \in B_n$ and $|x| < n$, we define $\mathsf{Fix}(C, x)$ as the PV function that outputs a circuit obtained by fixing the last $|x|$ input bits of C to x, i.e., it outputs $C_x \in B_{n-|x|}$ such that $C_x(u) = C(u \circ x)$. Note that for properly defined PV functions Swap and Fix, we can prove in PV that if $|x| = i-1$, $\sigma = (\sigma_1, \sigma_2) \in \{0,1\}^2$, $C^{12}_{x,\sigma} \triangleq \mathsf{Fix}(C, \sigma_1 \circ \sigma_2 \circ x)$ is functionally equivalent to $C^{21}_{x,\sigma} \triangleq \mathsf{Fix}(\mathsf{Swap}(C,i), \sigma_2 \circ \sigma_1 \circ x)$. Moreover, we may assume that it is provable in PV that for any circuit $C \in B_n$, $x \in \{0,1\}^k$, and $z \in \{0,1\}^{n-k-1}$, $\mathsf{Eval}(\mathsf{Fix}(C,x), s_i(z)) = \mathsf{Eval}(\mathsf{Fix}(C, s_i(x)), z)$, and that $\mathsf{Fix}(C, \varepsilon) = C$.

Therefore, by the *Global Consistency* of approximate counting, we have that

$$\forall x \in \{0,1\}^{i-1}\ \forall \sigma \in \{0,1\}^2\ |\mathsf{P}_\eta(C^{12}_{x,\sigma}) - \mathsf{P}_\eta(C^{21}_{x,\sigma})| \le 3 \cdot \eta. \tag{3.1}$$

That is, for $x \in \{0,1\}^{i-1}$, if we arbitrarily fix the *rightmost* two bits of $\mathsf{Fix}(C,x)$ and $\mathsf{Fix}(\mathsf{Swap}(C,i),x)$, their acceptance probabilities are close. By applying Local Consistency Axiom twice and subsequently the *Global Consistency* of approximate counting, we can prove that

$$\forall x \in \{0,1\}^{i-1}\ \left|\mathsf{P}_\eta(\mathsf{Fix}(C,x)) - (1/4) \sum_{\sigma \in \{0,1\}^2} \mathsf{P}_\eta(C^{12}_{x,\sigma})\right| \le 10 \cdot \eta;$$

$$\forall x \in \{0,1\}^{i-1}\ \left|\mathsf{P}_\eta(\mathsf{Fix}(\mathsf{Swap}(C,i),x)) - (1/4) \sum_{\sigma \in \{0,1\}^2} P_\eta(C^{21}_{x,\sigma})\right| \le 10 \cdot \hat{\eta}.$$

Subsequently, we know from Equation (3.1) that

$$\forall x \in \{0,1\}^{i-1}\ |\mathsf{P}_\eta(\mathsf{Fix}(C,x)) - \mathsf{P}_\eta(\mathsf{Fix}(\mathsf{Swap}(C,i),x))| \le 20 \cdot \eta. \tag{3.2}$$

This shows that the acceptance probabilities of C and $\mathsf{Swap}(C,i)$ are close when the rightmost $i-1$ bits of them are both fixed by $x \in \{0,1\}^{i-1}$.

We will now prove that

$$|\mathsf{P}_\eta(C) - \mathsf{P}_\eta(\mathsf{Swap}(C,i))| \le 20 \cdot (n+1) \cdot \eta. \tag{3.3}$$

This suffices as we can pick $\eta \triangleq \beta/(100 \cdot (n+1))$ and apply the PRECISION CONSISTENCY AXIOM.

Suppose, towards a contradiction, that Equation (3.3) does not hold. We design an iterative algorithm that given C, i, and $k \le i-1$, outputs a string x of length k such that

$$|\mathsf{P}_\eta(\mathsf{Fix}(C,x)) - \mathsf{P}_\eta(\mathsf{Fix}(\mathsf{Swap}(C,i),x))| > 20 \cdot (n+1-k) \cdot \eta.$$

The algorithm is essentially the same as the algorithm in the proof of Lemma 3.1, i.e., it extends the string by one bit in each iteration by querying the approximate counting oracle. In particular, the base case $k=0$ is satisfied, as Equation (3.3) does not hold. Therefore, for $k = i-1$, the algorithm outputs a string $x \in \{0,1\}^{i-1}$ such that

$$|\mathsf{P}_\eta(\mathsf{Fix}(C,x)) - \mathsf{P}_\eta(\mathsf{Fix}(\mathsf{Swap}(C,i),x))| > 20 \cdot (n+1-i) \cdot \eta \ge 20 \cdot \eta.$$

This violates Equation (3.2) and thus completes the proof. □

Permutational Symmetry. We can then state and prove the *permutational symmetry* of approximate counting by decomposing a permutation into a sequence of transformations $C \mapsto \mathsf{Swap}(C,i)$.

We assume a straightforward encoding of permutations of $[n]$ for $n \in \mathsf{Log}$, and write $\pi \in S_n$ as an abbreviation of "π is a permutation of $[n]$" encoded by a straightforward PV function. Let $C \in B_n$ and $\pi \in S_n$ be a permutation of $[n]$. We define $\mathsf{Permute}(C,\pi)$ be the PV function that outputs a circuit $C \circ \pi \in B_n$ defined as $(C\circ\pi)(x) = C(x_{\pi_n} \circ \cdots \circ x_{\pi_1})$. Then we have that:

Lemma 3.5. $\mathsf{APX}_1 \vdash \forall n \in \mathsf{Log}\ \forall \pi \in S_n\ \forall C \in B_n\ \forall \delta^{-1}, \beta^{-1} \in \mathsf{Log}\ |\mathsf{P}_\delta(C) - \mathsf{P}_\delta(C \circ \pi)| \le 2\cdot\delta + \beta.$

Proof Sketch. Under a straightforward encoding of permutations of $[n]$, we can prove in PV that there is a list $\ell = (i_1, \ldots, i_k)$ for some $k \in \mathsf{Log}$ such that $C \circ \pi$ is functionally equivalent to C_k defined as

$$C_0 \triangleq C, \quad C_j \triangleq \mathsf{Swap}(C_{j-1}, i_j) \quad (j \in [k]).$$

By induction on j, we can prove by applying Lemma 3.4 that for any $\eta \in \mathsf{Log}$, $|\mathsf{P}_\eta(C) - \mathsf{P}_\eta(C_j)| \le 3\cdot j\cdot\eta$. This, together with the *Global Consistency* of approximate counting, implies that

$$|\mathsf{P}_\eta(C) - \mathsf{P}_\eta(C\circ\pi)| \le 3\cdot(k+1)\cdot\eta.$$

We then prove the lemma by taking $\eta \triangleq \beta/(10\cdot(k+1))$ and applying the PRECISION CONSISTENCY AXIOM. □

3.1.3 Existence Lemma for Approximate Counting

An important counting principle is that if a mathematical object can be sampled with non-zero probability, then it must *exist*. This simple result is the bedrock of the celebrated probabilistic method in combinatorics (see, e.g., [AS16]). The following lemma formalizes the principle in the context of approximate counting:

Lemma 3.6. $\mathsf{APX}_1 \vdash \forall n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}\ \forall C \in B_n\ (\beta > 0 \land \mathsf{P}_\delta(C) > \delta + \beta \to \exists x \in \{0,1\}^n\ C(x) = 1.$

Proof. We argue in APX_1. Fix any $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and n-input circuit $C \in B_n$. Suppose that $\beta > 0$ and $\mathsf{P}_\delta(C) > \delta + \beta$, and let $\eta^{-1} \in \mathsf{Log}$ be determined later.

By the Precision Consistency Axiom, we can see that $\mathsf{P}_\eta(C) \geq \beta - \eta$. Suppose, towards a contradiction, that for every $x \in \{0,1\}^n$, $C(x) = 0$. In such case, $C(\vec{x})$ is equivalent to the circuit Null_n that always outputs 0. By the *Global Consistency* of approximate counting, we have that

$$\mathsf{P}_\eta(\mathsf{Null}_n) \geq \mathsf{P}_\eta(C) - 3\eta \geq \beta - 4\eta.$$

Let $\eta = \beta/10$. We have that $\mathsf{P}_\eta(\mathsf{Null}_n) > 3\eta$, which leads to a contradiction with the Boundary Axiom. $\square$

We note that a more general version of the principle will be proved in Section 3.2.3 following a similar but more complicated argument, which will be later used to prove the *linearity of approximate expectation*.

3.1.4 Approximate Counting for Concrete Circuits

In this subsection, we consider the behavior of the approximate counting oracle on concrete circuits: the "less-than-t" circuit that parses its input as a number and outputs 1 if it is less than a fixed threshold, and circuits with a small number of inputs.

"Less-than-t" Circuits. Let $t \in \{0, 1, \ldots, 2^n\}$, $C_{<t} : \{0,1\}^n \to \{0,1\}$ be the circuit that parses its input as the binary encoding of a number $x \in \{0, 1, \ldots, 2^n - 1\}$ and accepts if and only if $x < t$. The following lemma shows in APX_1 that the acceptance probability of $C_{<t}$ is approximately $t/2^n$, as expected.

Lemma 3.7 (Less-than-t Circuits)**.** $\mathsf{APX}_1 \vdash \forall n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}\ \forall t \in \{0, 1, \ldots, 2^n\}\ |\mathsf{P}_\delta(C_{<t}) - t/2^n| \leq \delta + \beta$.

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1}, t \in \{0, 1, \ldots, 2^n\}$. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Note that when $t = 2^n$, $C_{<t}$ is functionally equivalent to True_n, and thus the lemma immediately follows from the *Global Consistency* and Boundary Axiom.[15] In the rest of the proof, we assume $t < 2^n$.

Suppose, towards a contradiction, that $|\mathsf{P}_\delta(C_{\leq t}) - t/2^n| > \delta + \beta$. By the Precision Consistency Axiom, we have that $|\mathsf{P}_\eta(C_{<t}) - \mathsf{P}_\delta(C_{<t})| \leq \delta + 2\eta$, and subsequently

$$|\mathsf{P}_\eta(C_{<t}) - t/2^n| > \beta - 2\eta.$$

We may assume that the rightmost bit of t is the most significant bit; this is without loss of generality by the *Permutational Symmetry* of approximate counting.

We will design an P-oracle iterative algorithm that, in the i-th iteration, outputs $t_i \in \{0, 1, \ldots, 2^{n-i} - 1\}$ satisfying the following condition:

- Let $C_i : \{0,1\}^{n-i} \to \{0,1\}$ be the circuit that parses its input as a number $x \in \{0, 1, \ldots, 2^{n-i} - 1\}$ and outputs 1 if $x < t_i$. Then $\left|\mathsf{P}_\eta(C_i) - t_i/2^{n-i}\right| > \beta - 20\eta \cdot (i+1)$.

The algorithm starts with $t_0 \triangleq t$ (and thus $C_0 \triangleq C_{<t}$). In the i-th iteration, the algorithm considers the rightmost bit (i.e. the most significant bit) of t_i. Recall that $\mathsf{Fix}(C, b)$ outputs the circuit obtained from C by fixing the rightmost input bit to be b.

If the rightmost bit of t_i is 0, the algorithm outputs $t_{i+1} \triangleq t_i$. It is clear that $\mathsf{Fix}(C_i, 1)$ is functionally equivalent to the Null_{n-i-1}, and thus by the *Global Consistency* and Boundary Axiom, $\mathsf{P}_\eta(\mathsf{Fix}(C_i, 1)) \leq 3\eta$. Let C_{i+1} be the circuit that parses its input as a number $x \in \{0, 1, \ldots, 2^n - 1\}$ and outputs 1 if $x < t_{i+1}$. It follows that $\mathsf{Fix}(C_i, 0)$ is functionally equivalent to C_{i+1}, and thus by the *Global Consistency* of approximate

[15] For this step to hold, we need for $C_{<t}$ to be provably equivalent to True_n. This will hold in PV_1 for a natural implementation of the circuits $C_{<t}$.

counting, $|\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) - \mathsf{P}_\eta(C_{i+1})| \le 3\eta$. Subsequently:

$$\begin{aligned}
&\left|\mathsf{P}_\eta(C_{i+1}) - t_{i+1}/2^{n-i-1}\right| \\
\ge\ &\left|\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) - t_{i+1}/2^{n-i-1}\right| - 3\eta \\
\ge\ &\left|\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_i,1)) - t_{i+1}/2^{n-i-1}\right| - 6\eta \\
=\ &2\cdot\left|\frac{\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_i,1))}{2} - \frac{t_{i+1}}{2^{n-i}}\right| - 6\eta \\
\ge\ &2\cdot\left|\mathsf{P}_\eta(C_i) - t_i/2^{n-i}\right| - 12\eta && (\text{Local Consistency Axiom}) \\
>\ &\beta - 20\eta\cdot(i+1) - 12\eta \ge \beta - 20\eta\cdot(i+2).
\end{aligned}$$

If the rightmost bit of t_i is 1, the algorithm outputs $t_{i+1} = t_i - 2^{n-i-1}$. It is clear that $\mathsf{Fix}(C_i,0)$ is functionally equivalent to True_{n-i+1}, and thus by the *Global Consistency* and Boundary Axiom, $|\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) - 1| \le 3\eta$. Let C_{i+1} be the circuit that parses its input as a number $x \in \{0,1,\dots,2^n-1\}$ and outputs 1 if $x < t_{i+1}$. It follows that $\mathsf{Fix}(C_i,1)$ is functionally equivalent to C_{i+1}, and thus by the *Global Consistency* of approximate counting, $|\mathsf{P}_\eta(\mathsf{Fix}(C_i,1)) - \mathsf{P}_\eta(C_{i+1})| \le 3\eta$. Subsequently:

$$\begin{aligned}
&\left|\mathsf{P}_\eta(C_{i+1}) - t_{i+1}/2^{n-i-1}\right| \\
\ge\ &\left|\mathsf{P}_\eta(\mathsf{Fix}(C_i,1)) - t_{i+1}/2^{n-i-1}\right| - 3\eta \\
\ge\ &\left|\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) - 1 + \mathsf{P}_\eta(\mathsf{Fix}(C_i,1)) - t_{i+1}/2^{n-i-1}\right| - 6\eta \\
=\ &2\cdot\left|\frac{\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_i,1))}{2} - \frac{t_{i+1}+2^{n-i-1}}{2^{n-i}}\right| - 6\eta \\
=\ &2\cdot\left|\frac{\mathsf{P}_\eta(\mathsf{Fix}(C_i,0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_i,1))}{2} - \frac{t_i}{2^{n-i}}\right| - 6\eta \\
\ge\ &2\cdot\left|\mathsf{P}_\eta(C_i) - t_i/2^{n-i}\right| - 12\eta && (\text{Local Consistency Axiom}) \\
>\ &\beta - 20\eta\cdot(i+1) - 12\eta \ge \beta - 20\eta\cdot(i+2).
\end{aligned}$$

It is clear that the correctness of the algorithm follows from the induction principle for polynomial-time verifiable properties allowed by the n-Induction Axiom of APX_1. Therefore, after n iterations, the algorithm will output $t_n = 0$ such that the acceptance probability of the circuit $C_n \equiv \mathsf{Null}_0$ is at least $\beta - 20\eta\cdot(n+1)$. By setting $\eta \triangleq \beta/(40n+40)$, we can conclude a contradiction using the Boundary Axiom. □

Circuits with Short Inputs. For circuits $C : \{0,1\}^n \to \{0,1\}$ such that $n \in \mathsf{LogLog}$ it is feasible to enumerate all inputs of C. The following lemma shows that $\mathsf{P}_\delta(C)$ is consistent with its acceptance probability computed via the brute-force algorithm.

Lemma 3.8 (Brute Force Counting Lemma)**.** APX_1 *proves the following statement. For every $n \in \mathsf{LogLog}$, circuit $C \in B_n$, and $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, let t be the number of accepting inputs of C. Then $|\mathsf{P}_\delta(C) - t/2^n| \le \delta + \beta$. In particular, if $\delta \le 2^{-n-1}$, t is the nearest integer to $\mathsf{P}_\delta(C)\cdot 2^n$.*

The proof of the lemma employs the Local Consistency Axiom and the Precision Consistency Axiom. We will formalize the argument using the following general tool: If there is a sequence of circuits serving as an approximating counting algorithm for a circuit C, in the sense that it satisfies the boundary condition and is locally consistent, then the acceptance probability estimated by the algorithm is necessarily close to $\mathsf{P}_\delta(C)$. Formally:

Lemma 3.9 (Dueling Lemma)**.** APX_1 *proves the following statement. Let $n \in \mathsf{Log}$, $C \in B_n$ be a circuit, and $P_0, P_1, \dots, P_n$ be circuits that output rational numbers such that P_i is of input length i. Let $\delta^{-1}, \eta^{-1} \in \mathsf{Log}$. Suppose that for every $i < n$ and every $x \in \{0,1\}^i$,*

$$\left|P_i(x) - \frac{P_{i+1}(x\circ 0) + P_{i+1}(x\circ 1)}{2}\right| \le \eta;$$

and that for every $x \in \{0,1\}^n$*,* $P_n(x) = C(x)$*. Then* $|\mathsf{P}_\delta(C) - P_0| \le \delta + 4\eta \cdot (n+1)$*.*

Proof. We argue in APX_1. Fix $n \in \mathsf{Log}$, $C \in B_n$, the circuits $P_0, P_1, \dots, P_n$, and $\delta^{-1}, \eta^{-1} \in \mathsf{Log}$. Suppose that it satisfies the two conditions in the lemma. Assume for contradiction that $|\mathsf{P}_\delta(C) - P_0| > \delta + 3\eta \cdot (n+1)$. By the Precision Consistency Axiom, we have that $|\mathsf{P}_\eta(C) - P_0| > 4\eta \cdot n + 3\eta$.

Let C^x be the circuit obtained by fixing the rightmost $|x|$ bits of C to be x. Consider the following algorithm that, in the i-th iteration, outputs $x_i \in \{0,1\}^i$ such that $|\mathsf{P}_\eta(C^{x_i}) - P_i(x_i)| > 4\eta \cdot (n-i) + 3\eta$. The algorithm starts with $x_0 \triangleq \varepsilon$. In the i-th iteration, we know by the Local Consistency Axiom that

$$\begin{aligned}
|\mathsf{P}_\eta(C^{x_i}) - P_i(x_i)| &\le \left| \frac{\mathsf{P}_\eta(C^{0\circ x_i}) + \mathsf{P}_\eta(C^{1\circ x_i})}{2} - P_i(x_i) \right| + 3\eta \\
&\le \left| \frac{\mathsf{P}_\eta(C^{0\circ x_i}) + \mathsf{P}_\eta(C^{1\circ x_i})}{2} - \frac{P_{i+1}(0\circ x_i) + P_{i+1}(1\circ x_i)}{2} \right| + 4\eta \\
&\le \frac{1}{2}\Big(|\mathsf{P}_\eta(C^{0\circ x_i}) - P_{i+1}(0\circ x_i)| + |\mathsf{P}_\eta(C^{1\circ x_i}) - P_{i+1}(1\circ x_i)| \Big) + 4\eta.
\end{aligned}$$

This means that for some $\sigma \in \{0,1\}$, $|\mathsf{P}_\eta(C^{\sigma\circ x_i}) - P_{i+1}(\sigma \circ x_i)| > 4\eta \cdot (n-i-1) + 3\eta$. The algorithm then proceeds by setting $x_{i+1} = \sigma \circ x_i$.

It is clear that the correctness of the algorithm can be proved by induction on a quantifier-free formula, which is available in APX_1. Therefore, in the n-th iteration, the algorithm outputs a string $x_n \in \{0,1\}^n$ such that $|\mathsf{P}_\eta(C^{x_n}) - P_n(x_n)| > 3\eta$. However, this violates the Boundary Axiom as $P_n(x_n) = C(x_n)$, and the circuit C^{x_n} is a constant circuit that outputs $C(x_n)$. ☐

Proof of Lemma 3.8. We argue in APX_1. Fix $n \in \mathsf{LogLog}$ and $C \in B_n$, $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, and let t be the number of accepting inputs of C. Let $P_0, \dots, P_n$ be the circuits such that $P_i(x)$ takes an i-bit input x and outputs the acceptance probability of $C^x : \{0,1\}^{n-i} \to \{0,1\}$ defined as $C^x(z) = C(z \circ x)$. In particular, $P_0 = t/2^n$. Let $\eta^{-1} \in \mathsf{Log}$ be determined later.

Since $n \in \mathsf{LogLog}$, it is provable in PV_1 that $P_0, \dots, P_n$ satisfy the conditions in the *Dueling Lemma*. Then $|\mathsf{P}_\delta(C) - P_0| = |\mathsf{P}_\delta(C) - t/2^n| \le \delta + 4\eta \cdot (n+1)$. The lemma follows by setting $\eta \triangleq \beta/(4n+4)$. ☐

3.2 Approximate Expectation and its Basic Theory

We now develop a theory of feasible random variables and their approximate expectation.

3.2.1 Definition of Random Variables and Approximate Expectation

We first define the approximate expectation of a discrete random variable taking values in $\mathbb{Q}$. Recall that explicit sets are sets encoded by an explicit list and, in particular, the size of explicit sets are always feasible. To have the expectation being feasibly computable (approximately), we restrict to the setting where the support of random variables are given as an *explicit set*.

Let $n \in \mathsf{Log}$, V be an explicit set of rational numbers, and C be a multi-output circuit such that APX_1 proves that

$$\forall x \in \{0,1\}^n \; C(x) \in V.$$

We say that (V, n, C) defines a random variable $X \triangleq C(\mathcal{U}_n)$, and define the expectation of X as

$$\sum_{v \in V} v \cdot \Pr_x[C(x) = v],$$

where the probability can be implemented by the approximate counting quantifier P in APX. This leads to the following formal definition of random variables and expectation.

Definition 3.10 (Random Variable)**.** Let V be an explicit set of rational numbers, $n \in \mathsf{Log}$, and C be a multi-output circuit. We say that (V, n, C) defines a *random variable* X over V if $\forall x \in \{0,1\}^n \; C(x) \in V$. The set V is called the *support* of X, C is called the *sampler* of X, and n is called the *seed length* of X.

Definition 3.11 (Approximate Expectation)**.** Let (V, n, C) be a tuple defining a random variable X over V, and $\delta^{-1} \in \mathsf{Log}$. We define the *approximate expectation* of X, denoted by $\mathbb{E}_\delta[X]$, as

$$\sum_{v \in V} v \cdot \mathsf{P}_\delta(C_v),$$

where C_v is the n-input circuit that given $x \in \{0,1\}^n$, output 1 (resp. 0) if $C(x) = v$ (resp. $C(x) \neq v$), "$\cdot$" denotes the multiplication of rational numbers, and $\sum$ denotes the summation of rational numbers.

We note that there is a $\mathsf{PV}(\mathsf{P})$ function $\mathsf{E}(V, n, C, \Delta)$ computing $\mathbb{E}_{|\Delta|^{-1}}[X]$ for the random variable X defined from (V, n, C); it enumerates over $v \in V$, constructs the circuit C_v, calls the oracle $p_v \leftarrow \mathsf{P}(C_v, \Delta)$, and sums over $v \cdot p_v$ for all $v \in V$. To see that this algorithm is feasible, notice that V is an explicit set of feasible size, and under the encoding specified in Section 2.1, the total length of all rational numbers in V is PV-provably feasible.

For simplicity, we will use the notation $C : \{0,1\}^n \to \mathbb{Q}$ to denote that C is a multi-output circuit whose output is parsed as a rational number.

Moreover, one may think of the acceptance probability of a circuit $C : \{0,1\}^n \to \{0,1\}$ as the expectation of the indicating random variable $I_C \triangleq (\{0,1\}, n, C)$ up to a small additive error, as shown by the following proposition. Therefore, the properties of expectation we will prove next also translate to properties of approximate counting.

Proposition 3.12. APX_1 *proves the following statement. Let* $n, \delta^{-1} \in \mathsf{Log}$ *and* $C : \{0,1\}^n \to \{0,1\}$ *be a Boolean circuit. Let* $I_C \triangleq (\{0,1\}, n, C)$ *be the indicator random variable for* $C(x) = 1$*. Then for any* $\beta^{-1} \in \mathsf{Log}$*,* $|\mathsf{P}_\delta(C) - \mathbb{E}_\delta[I_C]| \leq 2\delta + \beta$*.*

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and a circuit $C : \{0,1\}^n \to \{0,1\}$. By the definition of approximate expectation, we know that $\mathbb{E}_\delta[I_C] = 1 \cdot \mathsf{P}_\delta[C_1]$, where $C_1(x), C(x)$ are functionally equivalent circuits. By the Global Consistency of approximate counting, we have that

$$|\mathsf{P}_\delta(C) - \mathbb{E}_\delta[I_C]| = |\mathsf{P}_\delta(C) - \mathsf{P}_\delta[C_1]| \leq 2\delta + \beta. \qquad \square$$

3.2.2 Basic Properties of Approximate Expectation

Precision Consistency. Similar to the Precision Consistency Axiom for approximate counting, the definition of approximate expectation is consistent with respect to different precisions as shown in the proposition below.

Proposition 3.13. APX_1 *proves the following statement. Let* $n, \delta_1^{-1}, \delta_2^{-1} \in \mathsf{Log}$*,* $C : \{0,1\}^n \to \mathbb{Q}$*,* $V \subseteq \mathbb{Q}$ *be an explicit set such that* $\forall x \in \{0,1\}^n\ C(x) \in V$*. Let* X *be the random variable defined by* (V, n, C)*. Then for every* $\beta^{-1} \in \mathsf{Log}$*,*

$$|\mathbb{E}_{\delta_1}[X] - \mathbb{E}_{\delta_2}[X]| \leq (\delta_1 + \delta_2 + \beta) \cdot \|V\|,$$

where $\|V\| \triangleq \sum_{v \in V} |v|$ *is the* ℓ_1*-norm of* V *and* $|v|$ *denotes the absolute value of the rational* v*.*

Proof. We argue in APX_1. Recall that C_v is the circuit that outputs 1 if $C(x) = v$, and outputs 0 otherwise. By the definition of approximate counting, we can see that

$$\begin{aligned}
|\mathbb{E}_{\delta_1}[X] - \mathbb{E}_{\delta_2}[X]| &= \left| \sum_{v \in V} v \cdot \mathsf{P}_{\delta_1}(C_v) - \sum_{v \in V} v \cdot \mathsf{P}_{\delta_2}(C_v) \right| \\
&\leq \left| \sum_{v \in V} v \cdot (\mathsf{P}_{\delta_1}(C_v) - \mathsf{P}_{\delta_2}(C_v)) \right| \\
&\leq \left| \sum_{v \in V} v \cdot (\delta_1 + \delta_2 + \beta) \right| \leq (\delta_1 + \delta_2 + \beta) \cdot \|V\|,
\end{aligned}$$

where the second last inequality follows from the Precision Consistency Axiom. $\square$

Local Consistency. Similarly, we can prove that approximate expectation is locally consistent by fixing the rightmost bit of the random seed to be 0 or 1 randomly. Formally:

Proposition 3.14. APX_1 *proves the following statement. Let* $n, \delta^{-1} \in \mathsf{Log}$*,* $C : \{0,1\}^n \to \mathbb{Q}$*,* $V \subseteq \mathbb{Q}$ *be an explicit set such that* $\forall x \in \{0,1\}^n\ C(x) \in V$*. Let* X *be the random variable defined by* (V, n, C)*. Then for every* $\beta^{-1} \in \mathsf{Log}$*,*

$$\left|\mathbb{E}_\delta[X] - \frac{\mathbb{E}_\delta[X|_0] + \mathbb{E}_\delta[X|_1]}{2}\right| \le (2\delta + \beta) \cdot \|V\|,$$

where $\|V\| \triangleq \sum_{v \in V} |v|$ *is the* ℓ_1 *norm of* V*, and for* $b \in \{0,1\}$*,* $X|_b$ *denotes the random variable defined by* $(V, n-1, \mathsf{Fix}_b(C))$*.*

Proof. We argue in APX_1. Let $\eta^{-1} \in \mathsf{Log}$ be determined later. Recall that $\mathsf{Fix}_b(C)$ outputs the circuit obtained by fixing the rightmost input bit of C to be b, where $b \in \{0,1\}$. By the Local Consistency Axiom, we know that for every $v \in V$,

$$\left|\mathsf{P}_\eta(C_v) - \frac{\mathsf{P}_\eta(\mathsf{Fix}(C_v, 0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_v, 1))}{2}\right| \le 3\eta.$$

By the definition of approximate counting, we can calculate that

$$\begin{aligned}
&\left|\mathbb{E}_\eta[X] - \frac{\mathbb{E}_\eta[X|_0] + \mathbb{E}_\eta[X|_1]}{2}\right| \\
&= \left|\sum_{v \in V} v \cdot \mathsf{P}_\eta(C_v) - \frac{1}{2}\left(\sum_{v \in V} v \cdot \mathsf{P}_\eta((\mathsf{Fix}_0(C))_v) + \sum_{v \in V} v \cdot \mathsf{P}_\eta((\mathsf{Fix}_1(C))_v)\right)\right| \\
&\le \left|\sum_{v \in V} v \cdot \mathsf{P}_\eta(C_v) - \frac{1}{2}\left(\sum_{v \in V} v \cdot \mathsf{P}_\eta(\mathsf{Fix}_0(C_v)) + \sum_{v \in V} v \cdot \mathsf{P}_\eta(\mathsf{Fix}_1(C_v))\right)\right| + 3\eta \cdot \|V\| \qquad (\textit{Global Consistency}) \\
&= \sum_{v \in V} |v| \cdot \left|\mathsf{P}_\eta(C_v) - \frac{\mathsf{P}_\eta(\mathsf{Fix}(C_v, 0)) + \mathsf{P}_\eta(\mathsf{Fix}(C_v, 1))}{2}\right| + 3\eta \cdot \|V\| \\
&\le 6\eta \cdot \|V\|.
\end{aligned}$$

The result follows from the *Precision Consistency of Expectation* by taking $\eta \triangleq \beta/10$. ☐

Consistency in Support Extension. Suppose that X is a random variable defined by the tuple (V, n, C). Consider an explicit set V' such that $V \subseteq V'$. We can define another random variable X' that is essentially the same as X by considering the tuple (V', n, C). The following proposition shows that the expectation of X' and X are nearly the same, i.e., a support extension does not affect the expectation of a random variable significantly.

Proposition 3.15. APX_1 *proves the following statement. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$*,* $C : \{0,1\}^n \to \mathbb{Q}$*,* $V, V' \subseteq \mathbb{Q}$ *be explicit sets such that* $\forall x \in \{0,1\}^n\ C(x) \in V \subseteq V'$*. Let* X *be the random variable defined by* (V, n, C)*, and* X' *be the random variable defined by* (V', n, C)*. Then:*

$$|\mathbb{E}_\delta[X] - \mathbb{E}_\delta[X']| \le (2\delta + \beta) \cdot \|V' \setminus V\| \le (2\delta + \beta) \cdot \|V'\|.$$

where $\|V' \setminus V\| = \sum_{v \in V' \setminus V} |v|$ *is the* ℓ_1*-norm of* $V' \setminus V$*.*

Proof. We argue in APX_1. By the definition of approximate expectation, we know that

$$|\mathbb{E}_\delta[X] - \mathbb{E}_\delta[X']| = \left|\sum_{v \in V' \setminus V} v \cdot \mathsf{P}_\delta(C_v)\right| \le \sum_{v \in V' \setminus V} |v| \cdot \mathsf{P}_\delta(C_v), \tag{3.4}$$

where $C_v(x)$ outputs 1 if $C(x) = v$, and 0 otherwise. Therefore it suffices to prove that $\mathsf{P}_\delta(C_v) \le 2\delta + \beta$ for $v \in V' \setminus V$. Fix any $v \in V' \setminus V$. Note that since $C(x) \in V$ for $x \in \{0,1\}^n$, we know that $C(x) \neq v$ and thus C_v is (provably) functionally equivalent to Null_n. The desired bound then follows from the *Global Consistency* of approximate counting using that $\mathsf{P}_\delta(\mathsf{Null}_n) = 0$ by the BOUNDARY AXIOM. □

Permutational Symmetry. Suppose that X, X' are random variables defined by the tuples (V, n, C) and $(V, n, C \circ \pi)$, where $\pi \in S_n$ denotes a permutation of the input bits. Similar to the *Permutational Symmetry* of approximate counting, we will show that $\mathbb{E}[X] \approx \mathbb{E}[X']$.

Proposition 3.16. APX_1 *proves the following statement. Let $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C : \{0,1\}^n \to \mathbb{Q}$, $V \subseteq \mathbb{Q}$ be an explicit set such that $\forall x \in \{0,1\}^n\ C(x) \in V$. Let $\pi \in S_n$ be a permutation of the input bits. Let X, X' be the random variables defined by (V, n, C) and $(V, n, C \circ \pi)$, respectively. Then:*

$$|\mathbb{E}_\delta[X] - \mathbb{E}_\delta[X']| \le (2\delta + \beta) \cdot \|V\|,$$

where $\|V\|$ is the ℓ_1-norm of V.

Proof. We argue in APX_1. Let $\eta^{-1} \in \mathsf{Log}$ be determined later. We can calculate that

$$\begin{aligned}
|\mathbb{E}_\eta[X] - \mathbb{E}_\eta[X']| &\le \sum_{v \in V} |v| \cdot |\mathsf{P}_\eta(C_v) - \mathsf{P}_\eta((C \circ \pi)_v)| \\
&\le \sum_{v \in V} |v| \cdot |\mathsf{P}_\eta(C_v) - \mathsf{P}_\eta(C_v \circ \pi)| + 3\eta \cdot \|V\| \\
&\le 6\eta \cdot \|V\|.
\end{aligned}$$

Here, the second line follows from *Global Consistency* of approximate counting, and the third line follows from the *Permutational Symmetry* of approximate counting. Subsequently, the proposition follows from the *Precision Consistency of Expectation* by taking $\eta \triangleq \beta/10$. □

3.2.3 Averaging Argument for Expectation

We will prove a general version of the *averaging argument* that allows us to search for a suffix of the seed such that the given linear combination of expectations of random variables $X_1, \ldots, X_m$ is approximately preserved after fixing part of the seed.

Suppose that $X_1, \ldots, X_m$ are random variables supported on V defined by a sequence of circuits $C_1, \ldots, C_m$, each with seed length n, i.e., for each $i \in [m]$ and every $x \in \{0,1\}^n$, $C_i(x) \in V$. Let $\delta^{-1} \in \mathsf{Log}$. Let $\lambda_1, \ldots, \lambda_m \in \mathbb{Q}$ be coefficients, and consider the quantity

$$\mu_{n,m,\delta,\vec{\lambda}} \triangleq \lambda_1 \cdot \mathbb{E}_\delta[X_1] + \lambda_2 \cdot \mathbb{E}_\delta[X_2] + \cdots + \lambda_m \cdot \mathbb{E}_\delta[X_m]. \tag{3.5}$$

Let $z \in \{0,1\}^k$ for $k \in [n]$. We can define the random variable $X_i|_z$ for each $i \in [m]$ from $(V, n-k, \mathsf{Fix}(C_i, z))$, where $\mathsf{Fix}(C_i, z)$ outputs the circuit obtained from C_i by fixing the rightmost k bits to z. That is, $X_i|_z$ is the random variable obtained by fixing the last k input bits of C_i to be z.[16] Let $\mu_{n,m,\delta,\vec{\lambda}}|_z$ be the quantity

$$\mu_{n,m,\delta,\vec{\lambda}}|_z \triangleq \lambda_1 \cdot \mathbb{E}_\delta[X_1|_z] + \lambda_2 \cdot \mathbb{E}_\delta[X_2|_z] + \cdots + \lambda_m \cdot \mathbb{E}_\delta[X_m|_z]. \tag{3.6}$$

Theorem 3.17 (Averaging Argument for Expectation)**.** APX_1 *proves the following statement. Let $n, m, \delta^{-1} \in \mathsf{Log}$, $C_1, \ldots, C_m : \{0,1\}^n \to \mathbb{Q}$ be circuits, $V \subseteq \mathbb{Q}$ be an explicit set such that $\forall x \in \{0,1\}^n\ \forall i \in [m]\ C_i(x) \in V$, and $\vec{\lambda} = (\lambda_1, \ldots, \lambda_m)$ be a list of length m such that $\lambda_i \in \mathbb{Q}$ for $i \in [m]$.*

Then for every $k \in [n]$ and $\beta^{-1} \in \mathsf{Log}$, there is a $z \in \{0,1\}^k$ such that

$$\mu_{n,m,\delta,\vec{\lambda}}|_z \ge \mu_{n,m,\delta,\vec{\lambda}} - (2\delta + \beta) \cdot \|V\| \cdot \|\vec{\lambda}\|, \tag{3.7}$$

where $\|V\| \triangleq \sum_{v \in V} |v|$ and $\|\vec{\lambda}\| \triangleq \sum_{i \in [m]} |\lambda_i|$ are the ℓ_1-norm of V and λ, respectively, and $\mu_{n,m,\delta,\vec{\lambda}}$ and $\mu_{n,m,\delta,\vec{\lambda}}|_z$ are defined by Equation (3.5) and Equation (3.6), respectively.

[16] This is without loss of generality by the *Permutational Symmetry of Expectation*.

Proof. We argue in APX_1. Fix $n, m, \delta^{-1} \in \mathsf{Log}$, $C_1, \dots, C_m : \{0,1\}^n \to \mathbb{Q}$, let $V \subseteq \mathbb{Q}$ be an explicit set, $\vec{\lambda} = (\lambda_1, \dots, \lambda_m) \in \mathbb{Q}$, $k \in [n]$, $\beta^{-1} \in \mathsf{Log}$. Recall that by definition, we have that for each $b \in \{0,1\}$,

$$\mu_{n,m,\eta,\vec{\lambda}}|_z = \lambda_1 \cdot \mathbb{E}_\delta[X_1|_z] + \dots + \lambda_m \cdot \mathbb{E}_\delta[X_m|_z]; \tag{3.8}$$

$$\mu_{n,m,\eta,\vec{\lambda}}|_{z\circ b} = \lambda_1 \cdot \mathbb{E}_\delta[X_1|_{z\circ b}] + \dots + \lambda_m \cdot \mathbb{E}_\delta[X_m|_{z\circ b}]. \tag{3.9}$$

We will design a P-oracle polynomial-time algorithm $\mathsf{AvgSampler}(\pi)$ that takes

$$\pi \triangleq (1^n, 1^m, 1^{\delta^{-1}}, C_1, \dots, C_m, V, \vec{\lambda}, 1^k, 1^{\beta^{-1}})$$

as its input and outputs $z \in \{0,1\}^k$ such that Equation (3.7) holds. Theorem 3.17 then follows if the correctness of $\mathsf{AvgSampler}(\pi)$ can be proved in APX_1.

$\mathsf{AvgSampler}(\pi)$ is an iterative algorithm on k (i.e. the length of z). We will prove the invariant that for any $k \in \{0, 1, \dots, n\}$, the algorithm $\mathsf{AvgSampler}(\pi)$ outputs a string $z \in \{0,1\}^k$ such that

$$\mu_{n,m,\eta,\vec{\lambda}}|_z \geq \mu_{n,m,\eta,\vec{\lambda}} - 3k \cdot \eta \cdot \|V\| \cdot \|\vec{\lambda}\| \tag{3.10}$$

where $\eta^{-1} \in \mathsf{Log}$. We note that if this is possible, we can set $\eta \triangleq \beta/(3n+3)$ so that Equation (3.7) follows from the *Precision Consistency of Expectation*. Specifically, we can see that

$$\begin{aligned} &\left|\mu_{n,m,\eta,\vec{\lambda}}|_z - \mu_{n,m,\delta,\vec{\lambda}}|_z\right| \\ =& \left|\lambda_1 \cdot (\mathbb{E}_\eta[X_1|_z] - \mathbb{E}_\delta[X_1|_z]) + \dots + \lambda_m \cdot (\mathbb{E}_\eta[X_m|_z] - \mathbb{E}_\delta[X_m|_z])\right| \\ \leq& \left|\lambda_1 \cdot (\delta + 2\eta) \cdot \|V\| + \dots + \lambda_m \cdot (\delta + 2\eta) \cdot \|V\|\right| \leq (\delta + 2\eta) \cdot \|\vec{\lambda}\| \cdot \|V\| \end{aligned}$$

and similarly

$$\left|\mu_{n,m,\eta,\vec{\lambda}} - \mu_{n,m,\delta,\vec{\lambda}}\right| \leq (\delta + 2\eta) \cdot \|\vec{\lambda}\| \cdot \|V\|.$$

Equation (3.7) then follows from the triangle inequality.

For $k = 0$, $\mathsf{AvgSampler}(\pi)$ outputs ε, and Equation (3.10) holds as $\mu_{n,m,\vec{y},\eta,\vec{\lambda}}|_z = \mu_{n,m,\vec{y},\eta,\vec{\lambda}}$ by definition. Suppose that it has already obtained a string $z \in \{0,1\}^k$ such that Equation (3.10) holds. Our goal is to choose a bit $b \in \{0,1\}$ such that

$$\mu_{n,m,\eta,\vec{\lambda}}|_{b\circ z} \geq \mu_{n,m,\eta,\vec{\lambda}} - 3(k+1) \cdot \eta \cdot \|V\| \cdot \|\vec{\lambda}\|.$$

For each $i \in [m]$, we know by the *Local Consistency of Expectation* that

$$\left|\mathbb{E}_\eta[X_i|_z] - \frac{\mathbb{E}_\eta[X_i|_{0\circ z}] + \mathbb{E}_\eta[X_i|_{1\circ z}]}{2}\right| \leq 3\eta \cdot \|V\|$$

It then follows that

$$\begin{aligned} &\left|\mu_{n,m,\eta,\vec{\lambda}}|_z - \frac{\mu_{n,m,\eta,\vec{\lambda}}|_{0\circ z} + \mu_{n,m,\eta,\vec{\lambda}}|_{1\circ z}}{2}\right| \\ =& \left|\sum_{i\in[m]} \lambda_i \cdot \left(\mathbb{E}_\eta[X_i|_z] - \frac{\mathbb{E}_\eta[X_i|_{0\circ z}] + \mathbb{E}_\eta[X_i|_{1\circ z}]}{2}\right)\right| \leq 3\eta \cdot \|\vec{\lambda}\| \cdot \|V\|. \end{aligned}$$

Therefore, for some $b \in \{0,1\}$, we will have that $\mu_{n,m,\eta,\vec{\lambda}}|_{b\circ z} \geq \mu_{n,m,\eta,\vec{\lambda}}|_z - 3\eta \cdot \|\vec{\lambda}\| \cdot \|V\|$, which subsequently implies that

$$\mu_{n,m,\eta,\vec{\lambda}}|_{b\circ z} \geq \mu_{n,m,\eta,\vec{\lambda}}|_z - 3\eta \cdot \|\vec{\lambda}\| \cdot \|V\| \geq \mu_{n,m,\eta,\vec{\lambda}} - 3(k+1) \cdot \eta \cdot \|\vec{\lambda}\| \cdot \|V\|.$$

The algorithm $\mathsf{AvgSampler}$ can use the P-oracle to determine b and output $b\circ z$. This completes the proof. $\square$

3.2.4 Complementation

An easy corollary of the *Averaging Argument for Expectation* is *complementary counting*. That is, if X is a random variable over $\{0,1\}$ and $\overline{X} \triangleq 1 - X$, then $\mathbb{E}[\overline{X}] = 1 - \mathbb{E}[X]$. Formally:

Corollary 3.18 (Complementary Counting)**.** APX_1 *proves the following statement. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$*,* $C_1, C_2 \in B_n$ *such that for every* $x \in \{0,1\}^n$*,* $C_1(x) \neq C_2(x)$*. Then* $|\mathsf{P}_\delta(C_1) + \mathsf{P}_\delta(C_2) - 1| \leq 2\delta + \beta$*. Moreover, let* X_1, X_2 *be the indicator random variables of* C_1, C_2 *over* $\{0,1\}$*, respectively. Then* $|\mathbb{E}_\delta[X_1] + \mathbb{E}_\delta[X_2] - 1| \leq 2\delta + \beta$*.*

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C_1, C_2 \in B_n$. Let $\eta^{-1} \in \mathsf{Log}$ be determined later, and X_1, X_2 be the indicator random variables of C_1 and C_2, respectively. It is clear that for any total assignment ρ to the seed, $\mathbb{E}_\eta[X_1|_\rho] + \mathbb{E}_\eta[X_2|_\rho] - 1 = 0$. Therefore, by *Averaging Argument for Expectation* with $k = n$, $\mathbb{E}_\eta[X_1] + \mathbb{E}_\eta[X_2] - 1 \leq 6\eta$. Similarly, we can show that $-\mathbb{E}_\eta[X_1] - \mathbb{E}_\eta[X_2] + 1 \leq 6\eta$. This implies that

$$|\mathbb{E}_\eta[X_1] + \mathbb{E}_\eta[X_2] - 1| \leq 6\eta. \tag{3.11}$$

By Proposition 3.12, we have

$$|\mathsf{P}_\eta(C_1) + \mathsf{P}_\eta(C_2) - 1| \leq 12\eta.$$

Subsequently, by the Precision Consistency Axiom, we have $|\mathsf{P}_\delta(C_1) + \mathsf{P}_\delta(C_2) - 1| \leq 2\delta + 16\eta$. The desired bound then follows by setting $\eta \triangleq \beta/30$. The "Moreover" part follows from Equation (3.11) by the *Precision Consistency of Expectation*. ☐

3.2.5 Linearity of Expectation

We are now ready to prove the (approximate) linearity of expectation, one of the most useful results in probability theory. Let $X_1, \ldots, X_m, Y$ be random variables over an explicit set V. For a random seed z of the random variables, we use $X_i|_z$ and $Y|_z$ to denote the value that X_i and Y evaluate to, respectively. Suppose that for each random seed z, we have that

$$Y|_z = \gamma + \lambda_1 \cdot X_1|_z + \lambda_2 \cdot X_2|_z + \cdots + \lambda_m \cdot X_m|_z,$$

for some $\lambda_1, \ldots, \lambda_m \in \mathbb{Q}$. Then we should be able to obtain that $\mathbb{E}[Y]$ is close to

$$\gamma + \lambda_1 \cdot \mathbb{E}[X_1] + \lambda_2 \cdot \mathbb{E}[X_2] + \cdots + \lambda_m \cdot \mathbb{E}[X_m].$$

Formally, we have that:

Theorem 3.19 (Linearity of Expectation)**.** APX_1 *proves the following: Let* $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$*,* $C_1, \ldots, C_m : \{0,1\}^n \to \mathbb{Q}$ *be a list of circuits,* $\vec{\lambda} = (\lambda_1, \ldots, \lambda_m)$ *be a list of length* m *such that* $\lambda_i \in \mathbb{Q}$ *for* $i \in [m]$*,* $\gamma \in \mathbb{Q}$*, and* $V \subseteq \mathbb{Q}$ *be an explicit set such that:*

- *For any* $x \in \{0,1\}^n$ *and* $i \in [m]$*,* $C_i(x) \in V$*.*
- *For any* $x \in \{0,1\}^n$*,* $\gamma + \lambda_1 \cdot C_1(x) + \lambda_2 \cdot C_2(x) + \cdots + \lambda_m \cdot C_m(x) \in V$*.*

Let X_i *be the random variable defined by* (V, n, C_i) *for* $i \in [m]$*, and* Y *be the random variable defined by* (V, n, S)*, where* $S : \{0,1\}^n \to \mathbb{Q}$ *is a circuit such that*

$$S(x) = \gamma + \lambda_1 \cdot C_1(x) + \lambda_2 \cdot C_2(x) + \cdots + \lambda_m \cdot C_m(x).$$

Then:

$$|\mathbb{E}_\delta[Y] - (\gamma + \lambda_1 \cdot \mathbb{E}_\delta[X_1] + \cdots + \lambda_m \cdot \mathbb{E}_\delta[X_m])| \leq (2\delta + \beta) \cdot \|V\| \cdot \|\vec{\lambda}\|, \tag{3.12}$$

where $\|V\|$ *and* $\|\vec{\lambda}\|$ *are the* ℓ_1*-norm of* V *and* λ*, respectively.*

Proof. We argue in APX_1. We first prove that $\mathbb{E}_\delta[Y]-(\gamma+\lambda_1\cdot\mathbb{E}_\delta[X_1]+\cdots+\lambda_m\cdot\mathbb{E}_\delta[X_m]) \le (2\delta+\beta)\cdot\|V\|\cdot\|\vec{\lambda}\|$. Fix $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C_1, \ldots, C_m : \{0,1\}^n \to \mathbb{Q}$, $\vec{\lambda}$, γ, and V. Let

$$\mu \triangleq 1 \cdot \mathbb{E}_\delta[Y] + (-\lambda_1) \cdot \mathbb{E}_\delta[X_1] + \cdots + (-\lambda_m) \cdot \mathbb{E}_\delta[X_m],$$

and for each $z \in \{0,1\}^n$, we define $Y|_z$ and $X_i|_z$ to be random variable with seed length 0 as

$$Y|_z \triangleq S(z), \quad X_i|_z \triangleq C_i(z), \quad (i \in [m]), \tag{3.13}$$

and $\mu|_z$ as

$$\mu|_z \triangleq 1 \cdot \mathbb{E}_\delta[Y|_z] + (-\lambda_1) \cdot \mathbb{E}_\delta[X_1|_z] + \cdots + (-\lambda_m) \cdot \mathbb{E}_\delta[X_m|_z]. \tag{3.14}$$

By the *Averaging Argument for Expectation*, there is a string $z \in \{0,1\}^n$ such that $\mu|_z \ge \mu-(2\delta+\beta)\cdot\|V\|\cdot\|\vec{\lambda}\|$, which implies that $\mu \le \mu|_z + (2\delta + \beta) \cdot \|V\| \cdot \|\vec{\lambda}\|$.

Notice that $\mathbb{E}_\delta[Y|_z]$ is defined as

$$\mathbb{E}_\delta[Y|_z] = \sum_{v\in V} v \cdot \mathsf{P}_\delta(S_v) = \sum_{v\in V} v \cdot \mathsf{Bool}(S_v) = S(z),$$

where S_v is the circuit with no input that outputs 1 if and only if $S(z) = 1$. The second equality follows from the Boundary Axiom. Similarly, we can prove that for each $i \in [m]$, $\mathbb{E}_\delta[X_i|_z] = C_i(z)$. Subsequently, $\mu|_z = S(z) - (\lambda_1 \cdot C_1(z) + \cdots + \lambda_m \cdot C_m(z)) = \gamma$, which further implies that $\mu \le \gamma + (2\delta + \beta) \cdot \|V\| \cdot \|\vec{\lambda}\|$, i.e.,

$$\mathbb{E}_\delta[Y] - (\gamma + \lambda_1 \cdot \mathbb{E}_\delta[X_1] + \cdots + \lambda_m \cdot \mathbb{E}_\delta[X_m]) \le (2\delta + \beta) \cdot \|V\| \cdot \|\vec{\lambda}\|.$$

Finally, we can apply the same argument to μ' and $\mu'|_z$ defined by

$$\mu' \triangleq (-1) \cdot \mathbb{E}_\delta[Y] + \lambda_1 \cdot \mathbb{E}_\delta[X_1] + \cdots + \lambda_m \cdot \mathbb{E}_\delta[X_m]$$
$$\mu'|_z \triangleq (-1) \cdot \mathbb{E}[Y|_z] + \lambda_1 \cdot \mathbb{E}[X_1|_z] + \cdots + \lambda_m \cdot \mathbb{E}[X_m|_z]$$

to conclude that $\mu = -\mu' \ge \gamma - (2\delta + \beta) \cdot \|V\| \cdot \|\vec{\lambda}\|$. This completes the proof. □

3.3 Probability Inequalities

We now develop several standard inequalities related to (approximate) probability and expectation, including the union bound, Markov's inequality, and Chebyshev's inequality.

3.3.1 Union Bound

Another application of the averaging argument for approximate expectation (see Theorem 3.17) is the union bound. Recall that the acceptance probability of a circuit C can be formalized as the expectation of its indicating random variable $I_C \in \{0,1\}$. Therefore the union bound can be derived from the following principle: Let $X_1, \ldots, X_m, Y$ be Boolean-valued random variables such that for any random seed z, $Y|_z = X_1|_z \vee X_2|_z \vee \cdots \vee X_m|_z$. Then $\mathbb{E}[Y]$ should not be much larger than $\mathbb{E}[X_1]+\mathbb{E}[X_2]+\cdots+\mathbb{E}[X_m]$. Formally:

Theorem 3.20 (Union Bound)**.** APX_1 *proves the following statement. Let $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C_1, \ldots, C_m \in B_n$ be single-output circuits, $V = \{0,1\}$. Suppose that $\forall x \in \{0,1\}^n$ and $i \in [m]$, $C_i(x) \in V$, and let $Y, X_1, \ldots, X_m$ be random variables defined as follows.*

- *For each $i \in [m]$, X_i is defined by (V, n, C_i).*
- *Y is defined by (V, n, S), where $S(x) \in \{0,1\}$ is a circuit such that $S(x) \le C_1(x) \vee \cdots \vee C_m(x)$.*

Then we have $\mathbb{E}_\delta[Y] \le \mathbb{E}_\delta[X_1] + \cdots + \mathbb{E}_\delta[X_m] + (2\delta + \beta) \cdot m$.

Proof. We argue in APX_1. Fix $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C_1, \dots, C_m \in B_n$, and $V = \{0,1\}$. For each $z \in \{0,1\}^n$, we define $Y|_z$ as the random variable with seed length 0 that outputs $S(z)$, and X_i as the random variable with seed length 0 that outputs $C_i(z)$ for $i \in [m]$. Let μ and $\mu|_z$ be defined as

$$\mu \triangleq 1 \cdot \mathbb{E}_\delta[Y] + (-1) \cdot \mathbb{E}_\delta[X_1] + \cdots + (-1) \cdot \mathbb{E}_\delta[X_m], \tag{3.15}$$

$$\mu|_z \triangleq 1 \cdot \mathbb{E}_\delta[Y|_z] + (-1) \cdot \mathbb{E}_\delta[X_1|_z] + \cdots + (-1) \cdot \mathbb{E}_\delta[X_m|_z]. \tag{3.16}$$

By the *Averaging Argument for Expectation*, we can conclude that $\mu \le \mu|_z + (2\delta + \beta) \cdot m$ for some string z.

It then suffices to show that $\mu|_z \le 0$. Similarly to the proof of Theorem 3.19, we can prove by the Boundary Axiom that $\mathbb{E}_\delta[Y|_z] = S(z)$ and $\mathbb{E}_\delta[X_i|_z] = C_i(z)$ for $i \in [m]$. Subsequently, we know by the definition of $\mu|_z$ and the assumption on S that

$$\mu|_z = S(z) - (C_1(z) + \cdots + C_m(z)) \le 0,$$

is provable in APX_1. This completes the proof. ☐

3.3.2 Markov's Inequality

Next, we consider Markov's inequality. For a random variable X over an explicit set V, we should be able to prove that the probability that $X \ge k \cdot \mathbb{E}[X]$ cannot be much larger than $1/k$. This can be naturally formalized as follows:

Theorem 3.21 (Markov's Inequality)**.** *The following statement is provable in APX_1. Let X be a random variable defined by (V, n, C), where V is an explicit set of non-negative rational numbers, $n \in \mathsf{Log}$, and $C : \{0,1\}^n \to \mathbb{Q}$ is a circuit. Let $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $\mu \in \mathbb{Q}$ with $\mu \ge \mathbb{E}_\delta[X]$ and $\mu > 0$, $k \in \mathbb{Q}$ with $k > 0$, and $T(x)$ be the circuit that outputs 1 if $C(x) \ge k \cdot \mu$, and outputs 0 otherwise. Then*

$$\mathsf{P}_\delta(T) \le \delta + k^{-1} \cdot (1 + \delta \cdot \mu^{-1} \cdot \|V\|) + \beta \cdot (\mu^{-1} \cdot \|V\| + 1),$$

where $\|V\|$ is the ℓ_1-norm of V.

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C : \{0,1\}^n \to \mathbb{Q}$, $\mu, k \in \mathbb{Q}$, and V. Let $T(x)$ be the circuit as defined above, and $m = |V|$. We define the following random variables:

- Y is the indicator variable of $T(x)$, i.e., it is defined by $(\{0,1\}, n, T)$.
- For each $v \in V$, X_v is the indicator variable of $\mathsf{EQ}(C(x), v)$. Formally, let $C_v(x)$ be the circuit that outputs 1 if and only if $C(x) = v$, and outputs 0 otherwise, X_v is the random variable defined by $(\{0,1\}, n, C_v)$.

Let $\eta^{-1} \in \mathsf{Log}$ be a precision parameter to be determined later, and for $b \in \{0,1\}$, let $C_v^{(b)}(x) \in \{0,1\}$ be the circuit that outputs 1 if $C(x) = b$. By the definition of approximate expectation, we have

$$\sum_{v \in V} v \cdot \mathbb{E}_\eta[X_v] = \sum_{v \in V} \sum_{b \in \{0,1\}} v \cdot b \cdot \mathsf{P}_\eta(C_v^{(b)}) = \sum_{v \in V} v \cdot \mathsf{P}_\eta(C_v^{(1)})$$

By the definition of $C_v(x)$ and $C_v^{(1)}(x)$, we know that $C_v^{(1)}(x) = C_v(x)$, and therefore by the Global Consistency of approximate counting, we have that

$$|\mathsf{P}_\eta(C_v^{(1)}) - \mathsf{P}_\eta(C_v)| \le 3 \cdot \eta.$$

Subsequently, we can see from the *Precision Consistency of Expectation* that

$$\mu \geq \mathbb{E}_\delta[X] \geq \mathbb{E}_\eta[X] - (\delta + 2\eta) \cdot \|V\| \tag{3.17}$$
$$= \sum_{v \in V} v \cdot \mathsf{P}_\eta(C_v) - (\delta + 2\eta) \cdot \|V\| \tag{3.18}$$
$$\geq \sum_{v \in V} v \cdot \mathbb{E}_\eta[X_v] - (\delta + 5\eta) \cdot \|V\| \tag{3.19}$$
$$\geq k\mu \sum_{v \in V, v \geq k\mu} \mathbb{E}_\eta[X_v] - (\delta + 5\eta) \cdot \|V\|, \tag{3.20}$$

where the last inequality uses that k, μ, and V are all nonnegative. This implies that

$$\sum_{v \in V, v \geq k \cdot \mu} \mathbb{E}_\eta[X_v] \leq k^{-1} \cdot (1 + (\delta + 5\eta) \cdot \mu^{-1} \cdot \|V\|). \tag{3.21}$$

Below we also rely on the following inequality, which follows from Proposition 3.12:

$$|\mathbb{E}_\eta[Y] - \mathsf{P}_\eta(T)| \leq 3 \cdot \eta. \tag{3.22}$$

It is clear from the definition of $T(x)$ and $C_v(x)$ that

$$T(x) = \bigvee_{v \in V, v \geq k \cdot \mu} C_v(x).$$

Therefore, by the *Union Bound*, we can conclude that

$$\begin{aligned}
\mathsf{P}_\eta(T) &\leq \mathbb{E}_\eta[Y] + 3 \cdot \eta && \text{(Equation (3.22))} \\
&\leq \sum_{v \in V, v \geq k \cdot \mu} \mathbb{E}_\eta[X_v] + 3 \cdot \eta \cdot (m + 1) && \text{(Union Bound)} \\
&\leq k^{-1} \cdot (1 + (\delta + 5\eta) \cdot \mu^{-1} \cdot \|V\|) + 3 \cdot \eta \cdot (|V| + 1), && \text{(Equation (3.21))}
\end{aligned}$$

Finally, we take $\eta = \min\{\beta/(50(|V|+1)), \beta k/(50(|V|+1))\}$ and apply the Precision Consistency Axiom, so

$$\mathsf{P}_\delta(T) \leq \delta + \mathsf{P}_\eta(T) + 2\eta \leq \delta + k^{-1} \cdot (1 + \delta \cdot \mu^{-1} \cdot \|V\|) + \beta \cdot (\mu^{-1}\|V\| + 1).$$

This completes the proof. □

3.3.3 Variance and Chebyshev's Inequality

Next, we develop the basic theory of (approximate) variance and prove a form of Chebyshev's Inequality.

Definition and Basic Properties. Let X be a random variable defined by the tuple (V, n, C), where $n \in \mathsf{Log}$, V is an explicit set, and $C : \{0,1\}^n \to \mathbb{Q}$. We can define a random variable X^2 by the tuple (V^2, n, C^2), where $V^2 \triangleq \{v^2 \mid v \in V\}$ and $C^2(x) \triangleq (C(x))^2$. Similarly, we can define a random variable $X - \mu$ for any $\mu \in \mathbb{Q}$ by the tuple $(V - \mu, n, C_{-\mu})$, where $V - \mu \triangleq \{v - \mu \mid v \in V\}$ and $C_{-\mu}(x) \triangleq C(x) - \mu$. We can then define:

Definition 3.22 (Approximate Variance)**.** Let X be a random variable defined by the tuple (V, n, C), $\delta^{-1} \in \mathsf{Log}$. The approximate variance of X with precision parameter δ, denoted by $\mathsf{Var}_\delta[X]$, is defined as

$$\mathsf{Var}_\delta[X] \triangleq \mathbb{E}_\delta[(X - \mu)^2]$$

where $\mu \triangleq \mathbb{E}_\delta[X] \in \mathbb{Q}$.

As an example, we prove an analogy of the equality $\mathsf{Var}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$ for approximate variance by directly formalizing the standard proof in APX_1.

Proposition 3.23. APX_1 *proves the following statement. Let* X *be a random variable defined by the tuple* (V, n, C)*, where* $V \subseteq \mathbb{Q}$ *is an explicit set,* $n \in \mathsf{Log}$*, and* $C : \{0,1\}^n \to \mathbb{Q}$ *satisfies* $\forall x \in \{0,1\}^n\ C(x) \in V$*. Let* $\mu \triangleq \mathbb{E}_\delta[X] \in \mathbb{Q}$*. Then for any* $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$*,*

$$|\mathsf{Var}_\delta[X] - (\mathbb{E}_\delta[X^2] - \mu^2)| \le (2\delta + \beta) \cdot (1 + |\mu|) \cdot \|\hat{V}\|,$$

where $\hat{V} = V \cup (V - \mu)^2 \cup V^2 \cup \{1\}$*.*

Proof. We argue in APX_1. Fix $V \subseteq \mathbb{Q}$, $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, and $C : \{0,1\}^n \to \mathbb{Q}$. Let $\hat{V} = V \cup (V-\mu)^2 \cup V^2 \cup \{1\}$ be an explicit set, and $\mu \triangleq \mathbb{E}_\delta[X]$. We define random variables $\hat{Y}$, $\hat{X}$, $\hat{X}^2$ over $\hat{V}$ as follows:

- $\hat{Y}$ is the random variable that outputs $(X - \mu)^2$. Formally, let $Y(x) \triangleq (C(x) - \mu)^2$ be a circuit. We can prove that $Y(x) \in \hat{V}$ from the assumption $C(x) \in V$ for any $x \in \{0,1\}^n$. We then define $\hat{Y}$ by the tuple $(\hat{V}, n, Y)$.
- $\hat{X}$ be the random variable that outputs X. Formally, it is defined by the tuple $(\hat{V}, n, C)$.
- $\hat{X}^2$ be the random variable that outputs X^2. Formally, let $C^2(x) \triangleq (C(x))^2$ be a circuit. We can prove that $C^2(x) \in \hat{V}$ from the assumption $C(x) \in V$ for any $x \in \{0,1\}^n$. We then define $\hat{X}^2$ by the tuple $(\hat{V}, n, C^2)$.

Let $\eta^{-1} \in \mathsf{Log}$ be determined later. By the definition of the circuits Y, C, C^2, it is clear that

$$Y(x) = \mu^2 + 1 \cdot C^2(x) + (-2\mu) \cdot C(x).$$

By the *Linearity of Expectation*, we can see that

$$\left|\mathbb{E}_\eta[\hat{Y}] - (\mu^2 + \mathbb{E}_\eta[\hat{X}^2] - 2\mu \cdot \mathbb{E}_\eta[\hat{X}])\right| \le 6\eta \cdot \|\hat{V}\| \cdot (|\mu| + 1). \tag{3.23}$$

By the *Consistency in Support Extension*, we can also conclude that

$$\left|\mathbb{E}_\eta[\hat{Y}] - \mathbb{E}_\eta[(X-\mu)^2]\right|, \left|\mathbb{E}_\eta[\hat{X}^2] - \mathbb{E}_\eta[X^2]\right|, \left|\mathbb{E}_\eta[\hat{X}] - \mathbb{E}_\eta[X]\right| \le 3\eta \cdot \|\hat{V}\|. \tag{3.24}$$

By the triangle inequality and the *Precision Consistency of Expectation*, we have

$$\begin{aligned}
& |\mathsf{Var}_\delta[X] - (\mathbb{E}_\delta[X^2] - \mu^2)| \\
= \ & |\mathbb{E}_\delta[(X-\mu)^2] - (\mathbb{E}_\delta[X^2] - \mu^2)| \\
\le \ & |\mathbb{E}_\eta[(X-\mu)^2] - (\mathbb{E}_\eta[X^2] - \mu^2)| + 2 \cdot (\delta + 2\eta) \cdot \|\hat{V}\| && \text{(Proposition 3.13)} \\
\le \ & |\mathbb{E}_\eta[\hat{Y}] - \mathbb{E}_\eta[\hat{X}^2] + \mu^2| + (2\delta + 10\eta) \cdot \|\hat{V}\| && \text{(Equation (3.24))} \\
\le \ & |\mu^2 - (2\mu \cdot \mathbb{E}_\eta[\hat{X}] - \mu^2)| + (2\delta + 10\eta) \cdot \|\hat{V}\| + 6\eta \cdot \|\hat{V}\| \cdot (|\mu| + 1) && \text{(Equation (3.23))} \\
\le \ & |\mu^2 - (2\mu \cdot \mathbb{E}_\eta[X] - \mu^2)| + (2\delta + 10\eta) \cdot \|\hat{V}\| + 6\eta \cdot \|\hat{V}\| \cdot (|\mu| + 1) + 6\eta \cdot |\mu| \cdot \|\hat{V}\| && \text{(Equation (3.24))} \\
\le \ & |\mu^2 - (2\mu \cdot \mathbb{E}_\delta[X] - \mu^2)| + (2\delta + 10\eta) \cdot \|\hat{V}\| + 6\eta \cdot \|\hat{V}\| \cdot (|\mu| + 1) \\
& \quad + 6\eta \cdot |\mu| \cdot \|\hat{V}\| + 2|\mu| \cdot (\delta + 2\eta) \cdot \|\hat{V}\| && \text{(Proposition 3.13)} \\
\le \ & (2\delta + 10\eta) \cdot \|\hat{V}\| + 6\eta \cdot \|\hat{V}\| \cdot (|\mu| + 1) + 6\eta \cdot |\mu| \cdot \|\hat{V}\| + 2|\mu| \cdot (\delta + 2\eta) \cdot \|\hat{V}\|. && (\mu \triangleq \mathbb{E}_\delta[X])
\end{aligned}$$

The theorem then follows by taking $\eta = \beta/40$. ◻

Chebyshev's Inequality. We now prove a form of Chebyshev's inequality that provides a tail bound for random variables with known (approximate) variance. Formally:

Theorem 3.24 (Chebyshev's Inequality)**.** APX_1 *proves the following statement. Let X be a random variable defined by the tuple (V, n, C), $\delta^{-1} \in \mathsf{Log}$, where $V \subseteq \mathbb{Q}$ is an explicit set, $n \in \mathsf{Log}$, and $C : \{0,1\}^n \to \mathbb{Q}$ is a circuit such that $\forall x \in \{0,1\}^n\ C(x) \in V$. Let $\mu \triangleq \mathbb{E}_\delta[X] \in \mathbb{Q}$, $\sigma^2 \triangleq \mathsf{Var}_\delta[X]$, and $T(x)$ be the circuit that outputs 1 if $(C(x)-\mu)^2 \geq k \cdot \sigma^2$, and outputs 0 otherwise.*

Then for any $\beta^{-1} \in \mathsf{Log}$ and $k \in \mathbb{Q}$, where $k > 0$, we have that

$$\mathsf{P}_\delta(T) \leq \delta + k^{-1} \cdot (1 + \delta \cdot \sigma^{-2} \cdot \|\hat{V}\|) + \beta \cdot (\sigma^{-2} \cdot \|\hat{V}\| + 1),$$

where $\hat{V} = (V-\mu)^2 = \{(v-\mu)^2 \mid v \in V\}$.

Proof. We argue in APX_1. Fix $V, n, \delta^{-1}, \beta^{-1}, C : \{0,1\}^n \to \mathbb{Q}, \mu \triangleq \mathbb{E}_\delta[X], \sigma^2 \triangleq \mathsf{Var}_\delta[X], \beta^{-1}, k$. Let Y be the random variable defined by the tuple $(\hat{V}, n, S)$, where $S(x) \triangleq (C(x)-\mu)^2$. By the definition of approximate variance, we know that

$$\sigma^2 = \mathsf{Var}_\delta[X] = \mathbb{E}_\delta[(X-\mu)^2] = \mathbb{E}_\delta[Y].$$

By applying *Markov's Inequality* to the random variable Y, we can see that

$$\mathsf{P}_\delta(T) \leq \delta + k^{-1} \cdot (1 + \delta \cdot \sigma^{-2} \cdot \|\hat{V}\|) + \beta \cdot (\sigma^{-2} \cdot \|\hat{V}\| + 1).$$

This completes the proof. □

3.3.4 Pairwise Independence and Variance

Now we develop the notion of (almost) pairwise independence, and prove a form of the equality $\mathsf{Var}[X_1 + \cdots + X_m] = \mathsf{Var}[X_1] + \cdots + \mathsf{Var}[X_m]$ for pairwise independent random variables $X_1, \ldots, X_m$. This combined with Chebyshev's inequality serve as a standard technique to reduce the error probability of randomized algorithms.

Definition of (Almost) Independence. We start with the definition of (almost) independence of random variables. Let X_1, X_2 be random variables over V_1, V_2, respectively. Recall that the covariance of X and Y, denoted by $\mathsf{Cov}(X, Y)$, is defined as the quantity $\mathbb{E}[X \cdot Y] - \mathbb{E}[X] \cdot \mathbb{E}[Y]$, where $X \cdot Y$ is a random variable over $V_1V_2 \triangleq \{v_1 \cdot v_2 \mid v_1 \in V_1, v_2 \in V_2\}$. Formally:

Definition 3.25 (Covariance)**.** Let X_1 and X_2 be the random variables defined by the tuples (V_1, n, C_1) and (V_2, n, C_2), respectively, where $V_1, V_2 \subseteq \mathbb{Q}$ are explicit sets, $n \in \mathsf{Log}$, and $C_1, C_2 : \{0,1\}^n \to \mathbb{Q}$ are circuits. Let $\delta^{-1} \in \mathsf{Log}$, $V_1V_2 \triangleq \{v_1 \cdot v_2 \mid v_1 \in V_1, v_2 \in V_2\}$, $Y(x)$ be the circuit computing $C_1(x) \cdot C_2(x)$, and Y be the random variable defined by the tuple (V_1V_2, n, Y). The δ-approximate covariance of X_1 and X_2, denoted by $\mathsf{Cov}_\delta(X_1, X_2)$, is defined as

$$\mathsf{Cov}_\delta(X_1, X_2) \triangleq |\mathbb{E}_\delta[Y] - \mathbb{E}_\delta[X_1] \cdot \mathbb{E}_\delta[X_2]|.$$

Definition 3.26 (Almost Independence)**.** Let $\delta^{-1} \in \mathsf{Log}$, $\varepsilon \in \mathbb{Q}$. Random variables X_1 and X_2 are said to be ε-almost δ-approximately independent if $\mathsf{Cov}_\delta(X_1, X_2) \leq \varepsilon$.

We can then define the pairwise independence of a sequence of random variables.

Definition 3.27 (Pairwise Independence)**.** Let $n, m \in \mathsf{Log}$, $C_1, \ldots, C_m : \{0,1\}^n \to \mathbb{Q}$ be circuits, and $V \subseteq \mathbb{Q}$ be an explicit set such that $\forall i \in [m]\ \forall x \in \{0,1\}^n\ C_i(x) \in V$. Let $X_1, \ldots, X_m$ be random variables, where for each $i \in [m]$, X_i is defined by the tuple (V, n, C_i), and let $\delta^{-1} \in \mathsf{Log}$ and $\varepsilon \in \mathbb{Q}$. The sequence $X_1, \ldots, X_m$ of random variables is said to be ε-almost δ-approximately pairwise independent if for every pair (i, j) with $i, j \in [m]$ and $i \neq j$, X_i and X_j are ε-almost δ-approximately independent.

We may drop the parameter ε and simply say δ-approximately independent if $\varepsilon = 0$. Note that since the approximate expectation of random variables may incur an error, random variables X_1 and X_2 may not be perfectly independent even if $\varepsilon = 0$.

Sum of Pairwise Independent Variables. Now we are ready to prove the following result: Suppose that $X_1, \ldots, X_m$ are almost pairwise independent, and $Y = X_1 + \cdots + X_m$. Then the variance of Y is close to the sum of the variances of $X_1, X_2, \ldots, X_m$. Formally:

Theorem 3.28. *The following statement is provable in* APX_1. *Let* $n, m \in \mathsf{Log}$, $C_1, \ldots, C_m : \{0,1\}^n \to \mathbb{Q}$ *be circuits, and* $V \subseteq \mathbb{Q}$ *be an explicit set such that the following holds:*

- $\forall i \in [m]\ \forall x \in \{0,1\}^n\ C_i(x) \in V$;
- $\forall x \in \{0,1\}^n\ C_1(x) + C_2(x) + \cdots + C_m(x) \in V$.

Let $X_1, \ldots, X_m$ *be random variables, where for each* $i \in [m]$, X_i *is defined by the tuple* (V, n, C_i). *Let* Y *be the random variable defined by the tuple* (V, n, S), *where* $S(x)$ *is the circuit computing* $C_1(x) + C_2(x) + \cdots + C_m(x)$. *Let* $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$ *and* $\varepsilon \in \mathbb{Q}$. *Suppose that* $X_1, \ldots, X_m$ *are* ε*-almost* δ*-approximately pairwise independent. Then*

$$|\mathsf{Var}_\delta[Y] - (\mathsf{Var}_\delta[X_1] + \cdots + \mathsf{Var}_\delta[X_m])| \le (\varepsilon + 3\delta \cdot \|V\|^2) \cdot m^2 + \beta \cdot (\|V\| + 1)^3,$$

where $\|V\| = \sum_{v \in V} |v|$ *is the* ℓ_1*-norm of* V.

Proof. We argue in APX_1. Fix n, m, circuits $C_1, \ldots, C_m \in \{0,1\}^n \to \mathbb{Q}$, $V, \delta^{-1}, \varepsilon^{-1}, \beta^{-1}$. Let $\eta^{-1} \in \mathsf{Log}$ be a precision parameter to be determined later, $\mu \triangleq \mathbb{E}_\eta[Y]$, and $\mu_i \triangleq \mathbb{E}_\eta[X_i]$ for $i \in [m]$.

Overview of the proof. Recall that by Proposition 3.23, we have that

$$\left|\mathsf{Var}_\eta[Y] - (\mathbb{E}_\eta[Y^2] - \mu^2)\right| \le 3\eta \cdot (1 + |\mu|) \cdot \|\hat{V}_Y\|, \tag{3.25}$$

where $\hat{V}_Y = V \cup (V - \mu)^2 \cup V^2 \cup \{1\}$. Similarly, for each $i \in [m]$, we have that

$$\left|\mathsf{Var}_\eta[X_i] - (\mathbb{E}_\eta[X_i^2] - \mu_i^2)\right| \le 3\eta \cdot (1 + |\mu_i|) \cdot \|\hat{V}_i\|, \tag{3.26}$$

where $\hat{V}_i = V \cup (V - \mu_i)^2 \cup V^2 \cup \{1\}$. Therefore, it suffices to bound

$$\Delta \triangleq |\mathbb{E}_\eta[Y^2] - (\mathbb{E}_\eta[X_1^2] + \cdots + \mathbb{E}_\eta[X_m^2]) - \mu^2 + (\mu_1^2 + \ldots \mu_m^2)| \tag{3.27}$$

and combine it with Equation (3.25) and (3.26). At a high level, our plan is to apply the *Linearity of Expectation* to prove that $\mathbb{E}_\eta[Y^2]$ is close to

$$\sum_{i=1}^m \mathbb{E}_\eta[X_i^2] + \sum_{i,j \in [m], i \ne j} \mathbb{E}_\eta[X_i X_j], \tag{3.28}$$

which is subsequently close to

$$\sum_{i=1}^m \mathbb{E}_\eta[X_i^2] + \sum_{i,j \in [m], i \ne j} \mu_i \mu_j$$

by the almost pairwise independence of $X_1, \ldots, X_m$. Finally, we can apply the *Linearity of Expectation* to show that

$$\mu_1^2 + \cdots + \mu_m^2 + \sum_{i,j \in [m], i \ne j} \mu_i \mu_j = \left(\sum_{i=1}^m \mu_i\right)^2 \approx \mu^2.$$

Putting the estimates together provides the upper bound for Δ.

Step 1: Approximation of $\mathbb{E}_\eta[Y^2]$. We first show that $\mathbb{E}_\eta[Y^2]$ is close to Equation (3.28). Recall that Y^2, X^2, and X_iX_j are the random variables over V^2 defined as follows:

- Y^2 is defined by the tuple (V^2, n, S^2), where $S^2(x) \triangleq (S(x))^2$.
- For $i \in [m]$, X_i^2 is defined by the tuple (V^2, n, C_i^2), where $C_i^2(x) \triangleq (C_i(x))^2$.
- For $i, j \in [m]$ such that $i \neq j$, X_iX_j is defined by the tuple (V^2, n, C_{ij}), where $C_{ij}(x) \triangleq C_i(x) \cdot C_j(x)$.

It is clear from the definition of the terms that for any $z \in \{0,1\}^n$, we have

$$S^2(z) = \sum_{i=1}^{m} C_i^2(z) + \sum_{i,j\in[m], i\neq j} C_{ij}(z).$$

Thus by the *Linearity of Expectation*, we have

$$\left| \mathbb{E}_\eta[Y^2] - \left(\sum_{i=1}^{m} \mathbb{E}_\eta[X_i^2] + \sum_{i,j\in[m], i\neq j} \mathbb{E}_\eta[X_iX_j] \right) \right| \leq 3\eta \cdot \|V^2\| \cdot (m^2 + m + 1) \leq 9\eta \cdot m^2 \cdot \|V^2\|. \tag{3.29}$$

Step 2: Applying pairwise independence. In this step we show that $\mathbb{E}_\eta[X_iX_j]$ is close to $\mu_i\mu_j$ for any $i, j \in [m]$. Recall that $\mu_i \triangleq \mathbb{E}_\eta[X_i]$. By the *Precision Consistency of Expectation*, we have

$$|\mathbb{E}_\eta[X_iX_j] - \mathbb{E}_\delta[X_iX_j]| \leq (\delta + 2\eta) \cdot \|V^2\|$$

for any $i, j \in [m]$, $i \neq j$. Moreover, since $X_1, \ldots, X_m$ are ε-almost δ-approximately pairwise independent, we know that

$$|\mathbb{E}_\delta[X_iX_j] - \mathbb{E}_\delta[X_i] \cdot \mathbb{E}_\delta[X_j]| \leq \varepsilon.$$

For each $i \in [m]$, we have that $|\mathbb{E}_\delta[X_i] - \mathbb{E}_\eta[X_i]| \leq (\delta + 2\eta) \cdot \|V\|$, which implies that

$$\begin{aligned}
&|\mathbb{E}_\delta[X_i] \cdot \mathbb{E}_\delta[X_j] - \mathbb{E}_\eta[X_i] \cdot \mathbb{E}_\eta[X_j]| \\
\leq &|\mathbb{E}_\delta[X_i] \cdot \mathbb{E}_\delta[X_j] - \mathbb{E}_\eta[X_i] \cdot \mathbb{E}_\delta[X_j]| + |\mathbb{E}_\eta[X_i] \cdot \mathbb{E}_\delta[X_j] - \mathbb{E}_\eta[X_i] \cdot \mathbb{E}_\eta[X_j]| \\
\leq &(\delta + 2\eta) \cdot \|V\| \cdot (|\mathbb{E}_\eta[X_i]| + |\mathbb{E}_\delta[X_j]|) \leq 2(\delta + 2\eta) \cdot \|V\|^2.
\end{aligned}$$

Combining the equations above, we have that

$$|\mathbb{E}_\eta[X_iX_j] - \mu_i\mu_j| \leq \varepsilon + (\delta + 2\eta) \cdot \|V^2\| + 2(\delta + 2\eta) \cdot \|V\|^2 \leq \varepsilon + 3(\delta + 2\eta)\|V\|^2. \tag{3.30}$$

Step 3: Applying linearity of expectation. The last step is to prove that $\mu \approx \mu_1 + \cdots + \mu_m$. Recall that $\mu \triangleq \mathbb{E}_\eta[Y]$ and $\mu_i \triangleq \mathbb{E}_\eta[X_i]$, where Y is defined by the tuple (V, n, S) and X_i is defined by the tuple (V, n, C_i). It is clear from the definition of S and C_i that for any $z \in \{0,1\}^n$,

$$S(z) = C_1(z) + C_2(z) + \cdots + C_m(z).$$

Therefore, by the *Linearity of Expectation*, we have that

$$|\mu - (\mu_1 + \cdots + \mu_m)| = |\mathbb{E}_\eta[Y] - (\mathbb{E}_\eta[X_1] + \cdots + \mathbb{E}_\eta[X_m])| \leq 3\eta \cdot \|V\| \cdot (m + 1). \tag{3.31}$$

Subsequently, we can see that

$$\begin{aligned}
\left|\mu^2 - (\mu_1 + \cdots + \mu_m)^2\right| &= |\mu + (\mu_1 + \cdots + \mu_m)| \cdot |\mu - (\mu_1 + \cdots + \mu_m)| \\
&\leq (2|\mu| + 3\eta \cdot \|V\| \cdot (m + 1)) \cdot 3\eta \cdot \|V\| \cdot (m + 1) \\
&\leq 6|\mu| \cdot \eta \cdot 2m \cdot \|V\| + 9\eta^2 \cdot (2m)^2 \cdot \|V\|^2 \\
&\leq 4 \cdot (9\eta^2 + 6\eta) \cdot m^2 \cdot \|V\|^2 && \text{(since } |u| \leq \|V\|\text{)} \\
&\leq 60 \cdot \eta \cdot (m\|V\|)^2, && (3.32)
\end{aligned}$$

where the last inequality holds as we will take $\eta \leq 1$.

Wrapping things up. Combining Equation (3.29), (3.30) and (3.32), we can see that

$$\begin{aligned}
\Delta =\ & |\mathbb{E}_\eta[Y^2] - (\mathbb{E}_\eta[X_1^2] + \cdots + \mathbb{E}_\eta[X_m^2]) - \mu^2 + (\mu_1^2 + \dots \mu_m^2)| \\
\le\ & \left| \sum_{i,j\in[m],i\neq j} \mathbb{E}_\eta[X_iX_j] - \mu^2 + \sum_{i=1}^m \mu_i^2 \right| + 9\eta \cdot m^2 \cdot \|V\|^2 && \text{(Equation (3.29))} \\
\le\ & \left| \sum_{i,j\in[m],i\neq j} \mu_i\mu_j - \mu^2 + \sum_{i=1}^m \mu_i^2 \right| + 9\eta \cdot m^2 \cdot \|V\|^2 + m^2(\varepsilon + 3(\delta + 2\eta)\cdot \|V\|^2) && \text{(Equation (3.30))} \\
\le\ & \left| -\mu^2 + \left(\sum_{i=1}^m \mu_i\right)^2 \right| + (\varepsilon + 3\delta \cdot \|V\|^2)\cdot m^2 + 20 \cdot \eta \cdot (m\|V\|)^2 \\
\le\ & 60 \cdot \eta \cdot (m\|V\|)^2 + (\varepsilon + 3\delta \cdot \|V\|^2)\cdot m^2 + 20 \cdot \eta \cdot (m\|V\|)^2 && \text{(Equation (3.32))} \\
\le\ & (\varepsilon + 3\delta \cdot \|V\|^2)\cdot m^2 + 80 \cdot \eta \cdot (m\|V\|)^2.
\end{aligned}$$

Finally, we combine this with Equation (3.25) and (3.26), which gives

$$|\mathsf{Var}_\delta[Y] - (\mathsf{Var}_\delta[X_1] + \cdots + \mathsf{Var}_\delta[X_m])| \le \Delta + 3\eta \cdot (1 + |\mu|) \cdot \|\hat{V}_Y\| + 3\eta \cdot \sum_{i=1}^m (1 + |\mu_i|) \cdot \|\hat{V}_i\|. \tag{3.33}$$

Note that

$$\begin{aligned}
|\mu| \le \|\hat{V}_Y\| \le\ & \|V\| + \|(V-\mu)^2\| + \|V^2\| + 1 \\
\le\ & \|V\| + \|V\|^2 + 1 + \sum_{v\in V} (v-\mu)^2 \\
\le\ & \|V\| + \|V\|^2 + 1 + \|V\|^2 + |V| \cdot \mu^2 + 2\mu \cdot \|V\|^2 \\
\le\ & 2 \cdot \|V\|^3 + (|V| + 2) \cdot \|V\|^2 + \|V\| + 1 = 8 \cdot |V| \cdot (\|V\| + 1)^3.
\end{aligned}$$

Similarly, we have that $|\mu_i| \le \|\hat{V}_i\| \le 8 \cdot |V| \cdot (\|V\| + 1)^3$.

Let $\eta \triangleq \min\{\beta/(2000 \cdot m^2 \cdot |V|), 1/10\} \le 1$. By combining Equation (3.33) and the upper bound above, we have that

$$\begin{aligned}
& |\mathsf{Var}_\delta[Y] - (\mathsf{Var}_\delta[X_1] + \cdots + \mathsf{Var}_\delta[X_m])| \\
\le\ & (\varepsilon + 3\delta \cdot \|V\|^2)\cdot m^2 + 80 \cdot \eta \cdot (m\|V\|)^2 + 1000 \cdot m \cdot \eta \cdot |V| \cdot (\|V\| + 1)^3 \\
\le\ & (\varepsilon + 3\delta \cdot \|V\|^2)\cdot m^2 + \beta \cdot (\|V\| + 1)^3.
\end{aligned}$$

This completes the proof. □

3.4 Independence, Error Reduction, and Concentration Bounds

We now consider the provability in APX_1 of concentration bounds for independent and identically distributed (i.i.d.) random variables, which are important tools in combinatorics, probability, and the analysis of randomized algorithms.

3.4.1 Explicit Independence

Before stating and proving the concentration bounds, we formally define the way we formulate independent and identically distributed random variables, and prove a form of the multiplication principle for approximate counting.

Formalization of i.i.d. RVs. We will only consider random variables that are "explicitly" i.i.d., in the sense that they are defined by the same sampling algorithm using disjoint parts of the random seed. This suffices for our applications and greatly simplifies the calculation of parameters for approximate counting. We first formally define explicit independence of random variables as follows:

Definition 3.29 (Explicit Independence)**.** Let X and X' be random variables defined by (V, n, C) and (V, n, C'), respectively. We say that X and X' are *explicitly independent* if C and C' read disjoint bits of the n-bit seed; that is, there is a partition $\pi_1 \cup \pi_2$ of $[n]$ such that for any seed $x \in \{0,1\}^n$, C only reads x_{π_1} and C' only reads x_{π_2}, where x_π denotes the bits of x with indices in π.

Similarly, we define explicitly i.i.d. random variables as follows:

Definition 3.30 (Explicitly i.i.d. RVs)**.** Let $n, m \in \mathsf{Log}$, $C : \{0,1\}^n \to \mathbb{Q}$ be a circuit, $V \subseteq \mathbb{Q}$ be an explicit set such that $\forall x \in \{0,1\}^n\ C(x) \in V$. The explicitly i.i.d. random variables $X_1, \ldots, X_m$ defined by the tuple (V, n, C) are obtained as follows.

Let $C_i(\cdot) : \{0,1\}^{nm} \to V$ be the circuit such that for any $\overline{x} = x_1 \circ x_2 \circ \cdots \circ x_m \in \{0,1\}^n$, where $x_i \in \{0,1\}^n$ for each $i \in [m]$, $C_i(\overline{x}) \triangleq C(x_i)$. For each $i \in [m]$, the random variable X_i is defined by the tuple (V, nm, C_i).

3.4.2 Multiplication Principle

We will need a form of multiplication principle: For any explicitly independent random variables X and Y, we have $\mathbb{E}[XY] \approx \mathbb{E}[X] \cdot \mathbb{E}[Y]$, or equivalently, $\mathsf{Cov}(X, Y)$ is small. In other words, explicitly independent random variables are approximately independent. Formally:

Theorem 3.31 (Multiplication Principle)**.** APX_1 *proves the following statement. Let $n, \delta^{-1} \in \mathsf{Log}$, V_1, V_2 be explicit sets, $C_1, C_2 \in \{0,1\}^n \to \mathbb{Q}$ be circuits. Suppose that the random variables X_1 and X_2, defined by the tuples (V_1, n, C_1) and (V_2, n, C_2), respectively, are explicitly independent. Then for any $\beta^{-1} \in \mathsf{Log}$,*

$$\mathsf{Cov}_\delta(X, Y) = |\mathbb{E}_\delta[X_1 X_2] - \mathbb{E}_\delta[X_1] \cdot \mathbb{E}_\delta[X_2]| \leq (2\delta + \beta) \cdot \|\hat{V}\| + (4\delta + \beta) \cdot \|\hat{V}\|^2,$$

where $\hat{V} \triangleq V_1 \cup V_2 \cup V_1 V_2$, and $V_1 V_2 \triangleq \{v_1 v_2 \mid v_1 \in V_1, v_2 \in V_2\}$.

Proof. We argue in APX_1. Fix $n, \delta^{-1} \in \mathsf{Log}$, V_1, V_2, C_1, C_2, and $\beta^{-1} \in \mathsf{Log}$. Let $\eta^{-1} \in \mathsf{Log}$ be determined later and $\hat{V} \triangleq V_1 \cup V_2 \cup V_1 V_2$ be an explicit set.

Overview of the proof. At a high level, the proof goes as follows. Let $\pi_1 \cup \pi_2$ be a partition of $[n]$ such that for any seed $x \in \{0,1\}^n$, X_1 only reads x_{π_1} and X_2 only reads x_{π_2}. Suppose, towards a contradiction, that $|\mathbb{E}[X_1 X_2] - \mathbb{E}[X_1] \cdot \mathbb{E}[X_2]|$ is large. By the *Averaging Argument for Expectation*, we may find an assignment ρ_1 to the part x_{π_1} of the seed such that

$$|\mathbb{E}[X_1 X_2|_{\rho_1}] - \mathbb{E}[X_1|_{\rho_1}] \cdot \mathbb{E}[X_2]|$$

is large. More formally, we are applying *Averaging Argument for Expectation* by treating $X_1 X_2$ and X_1 as random variables with coefficients 1 and $-\mathbb{E}[X_2]$, respectively. Note that since X_2 does not read the part x_{π_1} of the seed, we know that $\mathbb{E}[X_2]$ is close to $\mathbb{E}[X_2|_{\rho_1}]$, and subsequently

$$|\mathbb{E}[X_1 X_2|_{\rho_1}] - \mathbb{E}[X_1|_{\rho_1}] \cdot \mathbb{E}[X_2|_{\rho_1}]|$$

is also large.

Next, we apply the *Averaging Argument for Expectation* again by treating $X_1 X_2|_{\rho_1}$ and $X_2|_{\rho_1}$ as random variables with coefficients 1 and $-\mathbb{E}[X_1|_{\rho_1}]$, respectively, using the seed x_{π_2}. This gives an assignment ρ_2 to x_{π_2} such that

$$|\mathbb{E}[X_1 X_2|_{\rho_1}|_{\rho_2}] - \mathbb{E}[X_1|_{\rho_1}] \cdot \mathbb{E}[X_2|_{\rho_1}|_{\rho_2}]| \, .$$

is large. Note that, again, since X_1 does not read the part x_{π_2} of the seed, we know that $\mathbb{E}[X_1|_{\rho_1}]$ is close to $\mathbb{E}[X_1|_{\rho_1}|_{\rho_2}]$, and subsequently

$$|\mathbb{E}[X_1X_2|_{\rho_1}|_{\rho_2}] - \mathbb{E}[X_1|_{\rho_1}|_{\rho_2}] \cdot \mathbb{E}[X_2|_{\rho_1}|_{\rho_2}]|\,.$$

is also large. However, this is impossible as $X_1X_2|_{\rho_1}|_{\rho_2}$, $X_1|_{\rho_1}|_{\rho_2}$, and $X_2|_{\rho_1}|_{\rho_2}$ are random variables with seed length 0 and are supposed to satisfy $X_1X_2|_{\rho_1}|_{\rho_2} = X_1|_{\rho_1}|_{\rho_2} \cdot X_2|_{\rho_1}|_{\rho_2}$ by definition.

We now prove the theorem in detail. Note that we will implement the proof idea above in backward direction for simplicity of calculation.

Step 1: Averaging argument after fixing ρ_1. Let $\hat{Y}, \hat{X}_1, \hat{X}_2$ be the random variables over $\hat{V} \triangleq V_1 \cup V_2 \cup V_1V_2$ obtained from X_1X_2, X_1, X_2 via support extension. Let $\pi_1 \cup \pi_2$ be a partition of $[n]$ such that for any seed $x \in \{0,1\}^n$, C_1 only reads x_{π_1} and C_2 only reads x_{π_2}. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. It is clear that for any fixed assignments ρ_1, ρ_2 to x_{π_1}, x_{π_2}, respectively,

$$\mathbb{E}_\eta[\hat{Y}|_{\rho_1}|_{\rho_2}] = \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}|_{\rho_2}] \cdot \mathbb{E}_\eta[\hat{X}_2|_{\rho_1}|_{\rho_2}] \tag{3.34}$$

by the definition of the random variables.

Note that $\hat{X}_1|_{\rho_1}$ is a random variable that does not read the its seed x_{ρ_2}. Therefore, for any fixed ρ_2, ρ_2', $|\mathbb{E}_\eta[\hat{X}_1|_{\rho_1}|_{\rho_2'}] - \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}|_{\rho_2}]| = 0$. By the *Averaging Argument for Expectation*, we have that for any ρ_2,

$$|\mathbb{E}_\eta[\hat{X}_1|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}|_{\rho_2}]| \le 3\eta \cdot \|\hat{V}\|, \tag{3.35}$$

and subsequently by Equation (3.34),

$$|\mathbb{E}_\eta[\hat{Y}|_{\rho_1}|_{\rho_2}] - \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}] \cdot \mathbb{E}_\eta[\hat{X}_2|_{\rho_1}|_{\rho_2}]| \le 3\eta \cdot \|\hat{V}\| \cdot \mathbb{E}_\eta[\hat{X}_2|_{\rho_1}|_{\rho_2}] \le 3\eta \cdot \|\hat{V}\|^2. \tag{3.36}$$

Therefore, by the *Averaging Argument for Expectation*, we have that for any fixed assignment ρ_1 to x_{π_1},

$$|\mathbb{E}_\eta[\hat{Y}|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}] \cdot \mathbb{E}_\eta[\hat{X}_2|_{\rho_1}]| \le 3\eta \cdot \|\hat{V}\| \cdot (1 + |\mathbb{E}_\eta[\hat{X}_1|_{\rho_1}]|) \le 3\eta \cdot \|\hat{V}\| \cdot (1 + \|\hat{V}\|). \tag{3.37}$$

Note that here we treat $\hat{Y}|_{\rho_1}$, $\hat{X}_2|_{\rho_1}$ as random variables, and $\mathbb{E}_\eta[\hat{X}_1|_{\rho_1}]$ as a coefficient.

Step 2: Averaging argument again. Similarly to Equation (3.35), we will first show that $\mathbb{E}_\eta[\hat{X}_2|_{\rho_1}]$ is close to $\mathbb{E}_\eta[\hat{X}_2]$ for any ρ_1. We can see that for any assignments ρ_1, ρ_1' to x_{π_1} and ρ_2 to x_{π_2}, $\mathbb{E}_\eta[\hat{X}_2|_{\rho_1}|_{\rho_2}] = \mathbb{E}_\eta[\hat{X}_2|_{\rho_1'}|_{\rho_2}]$ as $\hat{X}_2$ does not read x_{π_1}. Therefore, by the *Averaging Argument for Expectation*, for every ρ_1, ρ_1',

$$|\mathbb{E}_\eta[\hat{X}_2|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_2|_{\rho_1'}]| \le 6\eta \cdot \|\hat{V}\|\cdot$$

Again, by the *Averaging Argument for Expectation* applied to $\hat{X}_2$, we get ρ_1' such that

$$|\mathbb{E}_\eta[\hat{X}_2|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_2]| \le 3\eta \cdot \|\hat{V}\| + |\mathbb{E}_\eta[\hat{X}_2|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_2|_{\rho_1'}]| \le 9\eta \cdot \|V\|.$$

Combining this with Equation (3.37), we have

$$\begin{aligned}
&|\mathbb{E}_\eta[\hat{Y}|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}] \cdot \mathbb{E}_\eta[\hat{X}_2]| \le |\mathbb{E}_\eta[\hat{Y}|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}] \cdot \mathbb{E}_\eta[\hat{X}_2|_{\rho_1}]| + 9\eta \cdot \|\hat{V}\|^2 \\
&\quad \le 12\eta \cdot \|\hat{V}\|^2 + 3\eta \cdot \|\hat{V}\|.
\end{aligned} \tag{3.38}$$

By applying the *Averaging Argument for Expectation* on Equation (3.38) with random variables $\hat{Y}$ and $\hat{X}_1$, we have that

$$\begin{aligned}
&|\mathbb{E}_\eta[\hat{Y}] - \mathbb{E}_\eta[\hat{X}_1] \cdot \mathbb{E}_\eta[\hat{X}_2]| \le |\mathbb{E}_\eta[\hat{Y}|_{\rho_1}] - \mathbb{E}_\eta[\hat{X}_1|_{\rho_1}] \cdot \mathbb{E}_\eta[\hat{X}_2]| + 3\eta \cdot \|V\| \cdot (1 + \|V\|) \\
&\quad \le 15\eta \cdot \|\hat{V}\|^2 + 6\eta \cdot \|\hat{V}\|.
\end{aligned} \tag{3.39}$$

Wrapping things up. Note that by the *Consistency in Support Extension*, we have that

$$|\mathbb{E}_\eta[\hat{Y}] - \mathbb{E}_\eta[X_1 X_2]|, |\mathbb{E}_\eta[\hat{X}_1] - \mathbb{E}_\eta[X_1]|, |\mathbb{E}_\eta[\hat{X}_2] - \mathbb{E}_\eta[X_2]| \le 3\eta \cdot \|\hat{V}\|.$$

By the *Precision Consistency of Expectation*, we have

$$|\mathbb{E}_\delta[X_1 X_2] - \mathbb{E}_\eta[X_1 X_2]|, |\mathbb{E}_\delta[X_1] - \mathbb{E}_\eta[X_1]|, |\mathbb{E}_\delta[X_2] - \mathbb{E}_\eta[X_2]| \le (2\delta + \eta) \cdot \|\hat{V}\|.$$

We can therefore deduce from Equation (3.39) that

$$\begin{aligned}
& |\mathbb{E}_\delta[X_1 X_2] - \mathbb{E}_\delta[X_1] \cdot \mathbb{E}_\delta[X_2]| \\
\le\ & |\mathbb{E}_\eta[X_1 X_2] - \mathbb{E}_\eta[X_1] \cdot \mathbb{E}_\eta[X_2]| + (2\delta + \eta) \cdot \|\hat{V}\| + (4\delta + 2\eta) \cdot \|V\|^2 \\
\le\ & |\mathbb{E}_\eta[\hat{Y}] - \mathbb{E}_\eta[\hat{X}_1] \cdot \mathbb{E}_\eta[\hat{X}_2]| + (2\delta + \eta) \cdot \|\hat{V}\| + (4\delta + 2\eta) \cdot \|V\|^2 + 3\eta \cdot \|\hat{V}\| + 6\eta \cdot \|\hat{V}\|^2 \\
\le\ & |\mathbb{E}_\eta[\hat{Y}] - \mathbb{E}_\eta[\hat{X}_1] \cdot \mathbb{E}_\eta[\hat{X}_2]| + (2\delta + 4\eta) \cdot \|\hat{V}\| + (4\delta + 8\eta) \cdot \|\hat{V}^2\| \\
\le\ & (2\delta + 10\eta) \cdot \|\hat{V}\| + (4\delta + 23\eta) \cdot \|\hat{V}\|^2.
\end{aligned}$$

This completes the proof by setting $\eta = \beta/100$. ☐

Subsequently, we can obtain the following more convenient form of the multiplication principle for explicitly independent Bernoulli random variables:

Corollary 3.32 (Multiplication Principle for Bernoulli RVs)**.** APX_1 *proves the following statement. Let* $n, m, \delta^{-1} \in \mathsf{Log}$ *and* $X_1, \ldots, X_m$ *be explicitly independent random variables over* $\{0,1\}$ *with seed length* n*. Then*

$$\left|\mathbb{E}_\delta[X_1 X_2 \ldots X_m] - \prod_{j \in [i]} \mathbb{E}_\delta[X_i]\right| \le 8\delta \cdot m,$$

where the random variable $X_1 X_2 \ldots X_m$ *is defined in the natural way.*

Proof. We argue in APX_1. Fix $n, m, \delta^{-1} \in \mathsf{Log}$. We prove by induction on i such that for every $i \in [m]$, we have

$$\left|\mathbb{E}_\delta[X_1 X_2 \ldots X_i] - \prod_{j \in [i]} \mathbb{E}_\delta[X_i]\right| \le 8\delta \cdot i. \tag{3.40}$$

Note that this employs induction on a quantifier-free formula, which is available in APX_1. The base case $i = 1$ is straightforward. Suppose that Equation (3.40) holds. By the *Multiplication Principle*,

$$\begin{aligned}
&|\mathbb{E}_\delta[X_1 X_2 \ldots X_i X_{i+1}] - \mathbb{E}_\delta[X_1 X_2 \ldots X_i] \cdot \mathbb{E}_\delta[X_{i+1}]| \\
&\quad \le 3\delta \cdot \|V\| + 5\delta \cdot \|V\|^2 \le 8\delta,
\end{aligned} \tag{3.41}$$

and subsequently

$$\begin{aligned}
& \left|\mathbb{E}_\delta[X_1 X_2 \ldots X_i X_{i+1}] - \prod_{j \in [i+1]} \mathbb{E}_\delta[X_i]\right| \\
\le\ & \left|\mathbb{E}_\delta[X_1 X_2 \ldots X_i] \cdot \mathbb{E}_\delta[X_{i+1}] - \prod_{j \in [i+1]} \mathbb{E}_\delta[X_i]\right| + 8\delta && \text{(Equation (3.41))} \\
\le\ & \left|\mathbb{E}_\delta[X_1 X_2 \ldots X_i] - \prod_{j \in [i]} \mathbb{E}_\delta[X_i]\right| \cdot |\mathbb{E}_\delta[X_{i+1}]| + 8\delta \\
\le\ & 8\delta \cdot i \cdot \|V\| + 8\delta \le 8\delta \cdot (i+1). && \text{(Induction Hypothesis)}
\end{aligned}$$

This completes the proof. ☐

3.4.3 Error Reduction for One-Sided Error Statements

The multiplication principle allows us to prove the correctness of error reduction via repetition for one-sided error algorithms. Specifically, for any circuit C that accepts a $(1-\varepsilon)$-fraction of its inputs, the circuit $C^{\vee k}(x_1,\dots,x_k)=\bigvee_{i\in[k]} C(x_i)$ accepts all but an ε^k-fraction its inputs.

This is formalized as the following theorem:

Theorem 3.33 (One-sided error reduction lemma)**.** *Let $C^{\vee k}:\{0,1\}^{nk}\to\{0,1\}$ be the circuit defined as $C^{\vee k}(x_1,\dots,x_k)\triangleq\bigvee_{i\in[k]} C(x_i)$ for any circuit $C:\{0,1\}^n\to\{0,1\}$. The following statement is provable in APX_1. For any $n,k,\delta^{-1},\gamma^{-1},\beta^{-1}\in\mathsf{Log}$ and circuit $C:\{0,1\}^n\to\{0,1\}$, if $\mathsf{P}_\delta(\neg C)\le\varepsilon$ and $\gamma\ge(\delta+\beta+\varepsilon)^k+\delta+\beta$, then $\mathsf{P}_\delta(\neg C^{\vee k})\le\gamma$.*

Proof. We argue in APX_1. Fix $n,\delta^{-1},\gamma^{-1},\beta^{-1}\in\mathsf{Log}$ and a circuit $C:\{0,1\}^n\to\{0,1\}$. Let $V\triangleq\{0,1\}$, and let $\eta^{-1}\in\mathsf{Log}$ be a parameter to be determined later. Let X_i be the random variable that takes an nk-bit seed $(x_1,\dots,x_k)\in(\{0,1\}^n)^k$ and outputs $\neg C(x_i)$ for $i\in[k]$. It is clear that $X_1,\dots,X_m$ are explicit independent random variables. Let Y be the random variable that takes an nk-bit seed $(x_1,\dots,x_k)\in(\{0,1\}^n)^k$ and outputs $\neg C^{\vee k}(x_1,\dots,x_k)$.

RVs and approximate counting. It is clear that Y is the indicator random variable for the circuit $\neg C^{\vee k}$, and thus by Proposition 3.12, $|\mathbb{E}_\eta[Y]-\mathsf{P}_\eta(\neg C^{\vee k})|\le 3\eta$.

We will prove that for every $i\in[k]$, $|\mathbb{E}_\eta[X_i]-\mathsf{P}_\eta(\neg C)|\le 6\eta$. Fix any $i\in[k]$ and let $\pi_i\cup\overline{\pi_i}=[nk]$, where π_i denotes the set of indices corresponding to the i-th n-bit block that X_i reads, and $\overline{\pi_i}$ denotes the other $(k-1)n$ indices. For every assignment ρ to $\overline{\pi_i}$, we can see that $X_i|_\rho$ is the indicator random variable for C, and thus by Proposition 3.12,

$$|\mathbb{E}_\eta[X_i|_\rho]-\mathsf{P}_\eta(\neg C)|\le 3\eta.$$

Subsequently, we can apply the *Averaging Argument for Expectation* to prove that

$$|\mathbb{E}_\eta[X_i]-\mathsf{P}_\eta(\neg C)|\le 6\eta. \tag{3.42}$$

Note that by the assumption, we have that $\mathsf{P}_\delta(\neg C)\le\varepsilon$. By the Precision Consistency Axiom for approximate counting, we have $\mathsf{P}_\eta(C)\le\delta+2\eta+\varepsilon$, and thus

$$\mathbb{E}_\eta[X_i]\le\mathsf{P}_\eta(C)+6\eta\le\delta+8\eta+\varepsilon \tag{3.43}$$

for every $i\in[k]$.

Wrapping things up. We first prove that $|\mathbb{E}_\eta[Y]-\mathbb{E}_\eta[X_1X_2\dots X_k]|\le 6\eta$. Note that Y and $X_1X_2\dots X_k$ are both random variables taking nk-bit seeds, and for every assignment ρ to the seeds, we have $\mathbb{E}_\eta[Y|_\rho]=\mathbb{E}_\eta[X_1X_2\dots X_k|_\rho]$. By the *Averaging Argument for Expectation*, we have

$$|\mathbb{E}_\eta[Y]-\mathbb{E}_\eta[X_1X_2\dots X_k]|\le 3\eta\cdot\|V\|\cdot 2\le 6\eta. \tag{3.44}$$

Additionally, by the Precision Consistency Axiom of approximate counting, we have

$$\mathsf{P}_\delta(\neg C^{\vee k})\le\mathsf{P}_\eta(\neg C^{\vee k})+\delta+2\eta.$$

Let $\eta\triangleq\beta/(20k)$. Recall that it suffices to prove $\mathbb{E}_\eta[Y]\le\gamma$. It follows that

$$\begin{aligned}
\mathsf{P}_\delta(\neg C^{\vee k}) &\le \mathsf{P}_\eta(\neg C^{\vee k})+\delta+2\eta && (\text{Precision Consistency Axiom})\\
&\le \mathbb{E}_\eta[Y]+\delta+5\eta\\
&\le \mathbb{E}_\eta[X_1X_2\dots X_k]+\delta+12\eta && (\text{Equation (3.44)})\\
&\le \prod_{i\in[k]}\mathbb{E}_\eta[X_i]+\delta+8\eta k+12\eta && (\textit{Multiplication Principle for Bernoulli RVs})\\
&\le \prod_{i\in[k]}(\delta+8\eta+\varepsilon)+\delta+8\eta k+12\eta && (\text{Equation (3.43)})\\
&\le (\delta+\beta+\varepsilon)^k+\delta+\beta\le\gamma,
\end{aligned}$$

which completes the proof. ☐

3.4.4 Chernoff Bound for $O(\log n)$ Random Variables

We consider a form of the Chernoff bound where the number of random variables $m \in \mathsf{LogLog}$. In a nutshell, we formalize a combinatorial proof using Binomial coefficients due to Chvátal (see, e.g., [Mul18, Section 3.2]).

Definition 3.34 (Binomial coefficient, in PV_1)**.** Let $n, m \in \mathsf{Log}$ and $m \le n$. The binomial coefficient $\binom{n}{m}$ is defined recursively as:

$$\binom{n}{0} \triangleq 1,\ \binom{n}{m} \triangleq \binom{n-1}{m-1} + \binom{n-1}{m}. \tag{3.45}$$

For $m > n$, we let $\binom{n}{m} \triangleq 0$. Note that the function computing $(1^n, 1^m) \mapsto \binom{n}{m}$ can be defined by a PV function that runs in $\mathsf{poly}(n)$ time, such that Equation (3.45) is provable in PV_1.

Lemma 3.35 (Binomial theorem)**.** *The following statement is provable in* PV_1*. For every* $n \in \mathsf{Log}$ *and* $x, y \in \mathbb{Q} \setminus \{0\}$*,*

$$(x+y)^n = \sum_{i=0}^{n} \binom{n}{i} x^i y^{n-i}.$$

Proof Sketch. Fix $n \in \mathsf{Log}$ and $x, y \in \mathbb{Q}$. We prove by induction on $n \in \mathsf{Log}$. In the base case, the equation trivially holds for $n = 0$ as both sides are 1. The induction step follows from Equation (3.45). ☐

Now we are ready to prove the Chernoff bound when $n \in \mathsf{LogLog}$.

Theorem 3.36 (Chernoff bound I, LogLog form)**.** *The following sentence is provable in* APX_1*. Let* $V = \{0,1\}$*,* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$*, and* $m \in \mathsf{LogLog}$*. Let* $X_1, \dots, X_m$ *be a sequence of explicit i.i.d. random variables over* V *defined by a tuple* (V, n, C) *and taking an* nm*-bit seed* $(z_1, \dots, z_m) \in (\{0,1\}^n)^m$*. Let* $p \triangleq \mathbb{E}_\delta[X_i] \in \mathbb{Q}$ *for any* $i \in [m]$*, and* $Y_{\ge k}$ *be the indicator variable of* $X_1 + \dots + X_m \ge k$ *that takes an* nm*-bit seed. Then for* $t \in \mathbb{Q}$*,* $0 \le t \le 1$*, and* $k \ge (1+t)pm$*,*

$$\mathbb{E}_\delta[Y_{\ge k}] \le \left(e^{-t^2 p/4} + (4\delta + \beta) \cdot 2^{-pt(1+t)}\right)^m + \delta + \beta.$$

Proof. We argue in APX_1. Fix $V = \{0,1\}$, $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $m \in \mathsf{LogLog}$, and $X_1, \dots, X_m$. Let $p \triangleq \mathbb{E}_\delta[X_i]$, t, k, and $Y_{\ge k}$ be defined as above. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. By the *Precision Consistency of Expectation*,

$$|\mathbb{E}_\eta[X_i] - p| = |\mathbb{E}_\delta[X_i] - \mathbb{E}_\eta[X_i]| \le \delta + 2\eta. \tag{3.46}$$

Probability of each subset. Let $\alpha \subseteq [m]$ encode a subset of variables. We define the random variable Y_α over $\{0,1\}$ as follows: Given any seed $(z_1, \dots, z_m) \in (\{0,1\}^n)^m$, $Y_\alpha = 1$ if and only if for every $i \in \alpha$, $X_i = 1$. In the first step, we show that

$$\mathbb{E}_\eta[Y_\alpha] \le (p + \delta + O(\eta m))^{|\alpha|} \tag{3.47}$$

for each $\alpha \subseteq [m]$. Note that as $m \in \mathsf{LogLog}$, we can define all random variables Y_α for $\alpha \subseteq [m]$ by an explicit list of circuits. This will be useful later in the proof.

Fix any $\alpha = \{i_1, \dots, i_t\} \subseteq [m]$. As $X_1, \dots, X_m$ are explicitly independent random variables, by the *Multiplication Principle for Bernoulli RVs*, we have that

$$\left|\mathbb{E}_\eta\left[\prod_{j\in[t]} X_{i_j}\right] - \prod_{j\in[t]} \mathbb{E}_\eta[X_{i_j}]\right| \le 8\eta m.$$

Subsequently, we have

$$\mathbb{E}_\eta\left[\prod_{j\in[t]} X_{i_j}\right] \le 8\eta m + (p+\delta+2\eta)^{|\alpha|}.$$

Note that for every assignment ρ to the random seed, $\mathbb{E}_\eta[Y_\alpha|_\rho] = \mathbb{E}_\eta[\prod_{j\in[t]} X_{i_j}|_\rho]$. By the *Averaging Argument for Expectation*, we have

$$\mathbb{E}_\eta[Y_\alpha] \le \mathbb{E}_\eta\left[\prod_{j\in[t]} X_{i_j}\right] + 6\eta \le 14\eta m + (p+\delta+2\eta)^{|\alpha|}. \tag{3.48}$$

Let $\overline{Y}_\alpha$ be the random variable defined over $\{0,1\}$ as follows: Given any seed $(z_1,\dots,z_m) \in (\{0,1\}^n)^m$, $Y_\alpha = 1$ if and only if for every $i \in [m]\setminus\alpha$, $X_i = 0$. Similar to the proof above, we have that

$$\mathbb{E}_\eta\left[\overline{Y}_\alpha\right] \le (1-p+\delta+10\eta m)^{m-|\alpha|} + 20\eta m. \tag{3.49}$$

Combining all subsets. Let $\hat{Y}_{\ge k}$ be the following random variable over $\{0,1\}$:

$$\hat{Y}_{\ge k} \triangleq \sum_{\alpha\subseteq[m],|\alpha|\ge k} Y_\alpha \overline{Y}_\alpha.$$

By applying the *Multiplication Principle* and *Linearity of Expectation*, we have

$$\begin{aligned}
&\mathbb{E}_\eta[\hat{Y}_{\ge k}] \\
&\le 3\eta\cdot 2^m + \sum_{\alpha\subseteq[m],|\alpha|\ge k} \mathbb{E}_\eta[Y_\alpha\overline{Y}_\alpha] && (\textit{Union Bound}) \\
&\le 3\eta\cdot 2^m + 8\eta\cdot 2^m + \sum_{\alpha\subseteq[m],|\alpha|\ge k} \mathbb{E}_\eta[Y_\alpha]\cdot\mathbb{E}[\overline{Y}_\alpha] && (\textit{Multiplication Principle}) \\
&\le 11\eta\cdot 2^m + \sum_{\alpha\subseteq[m],|\alpha|\ge k} \Big((p+\delta+2\eta m)^{|\alpha|} + 14\eta m\Big)\Big((1-p+\delta+10\eta m)^{m-|\alpha|} + 20\eta m\Big) \\
& && (\text{Equations (3.48) and (3.49)}) \\
&\le 90\eta m\cdot 2^m + \sum_{j\ge k}\binom{m}{j}(p+\delta+2\eta m)^j(1-p+\delta+10\eta m)^{m-j}.
\end{aligned}$$

Note that the binomial number in the last line is efficiently computable (even using a brute-force counting algorithm) as $m, j \in \mathsf{LogLog}$.

Recall that $Y_{\ge k}$ is the random variable indicating that $X_1+\dots+X_m \ge k$. Note that for every assignment ρ to the random seed, we have that $\mathbb{E}_\eta[Y_{\ge k}|_\rho] \le \mathbb{E}_\eta[\hat{Y}_{\ge k}|_\rho]$. Therefore, by the *Averaging Argument for Expectation*, we have

$$\mathbb{E}_\eta[Y_{\ge k}] \le \mathbb{E}_\eta[\hat{Y}_{\ge k}] + 6\eta \le \sum_{j\ge k}\binom{m}{j}(p+\delta+2\eta m)^j(1-p+\delta+10\eta m)^{m-j} + 100\eta m\cdot 2^m. \tag{3.50}$$

Binomial coefficient inequalities. It remains to prove an upper bound for Equation (3.50). Note that as $m \in \mathsf{LogLog}$, we can easy formalize the standard proof in [Mul18, Section 3.2], where all equalities about binomial coefficients can be proved in PV.

Let $\tau = e^{\lambda} \geq 1$ be a parameter to be determined later. In more detail, we can perform the following calculation for any $\varepsilon^{-1} \in \mathsf{Log}$:

$$\begin{aligned}
&\sum_{j\geq k}\binom{m}{j}(p+\varepsilon)^j(1-p+\varepsilon)^{m-j} \\
\leq&\sum_{j\geq k}\binom{m}{j}(p+\varepsilon)^j(1-p+\varepsilon)^{m-j}\tau^{j-k} + \sum_{0\leq j<k}(p+\varepsilon)^j(1-p+\varepsilon)^{m-j}\tau^{j-k} \\
=&\tau^{-k}\sum_{j=0}^{m}\binom{m}{j}\left((p+\varepsilon)\tau\right)^j(1-p+\varepsilon)^{m-j} \\
=&\tau^{-k}\big(1+p(\tau-1)+\varepsilon(\tau+1)\big)^m && \text{(Binomial Theorem)} \\
\leq&\left(\frac{1+p(e^{\lambda}-1)+\varepsilon(e^{\lambda}+1)}{e^{\lambda p(1+t)}}\right)^m.
\end{aligned}$$

Note that for $\lambda \in (0,1)$, we have $1+\lambda \leq e^{\lambda} \leq 1+\lambda+3\lambda^2/4$, and this is provable in PV. Then

$$\ln\big(1+p(e^{\lambda}-1)\big) \leq p(e^{\lambda}-1) \leq p\lambda + 3p\lambda^2/4.$$

This implies that $1+p(e^{\lambda}-1) \leq \exp(p\lambda+3p\lambda^2/4)$. We set $\varepsilon \triangleq \delta + 30\eta m$, $\eta \leq \beta/(120m)$, then $\varepsilon \leq \delta + \beta/4$, and $\lambda \triangleq t$. Then we have

$$\left(\frac{1+p(e^{\lambda}-1)+\varepsilon(e^{\lambda}+1)}{e^{\lambda p(1+t)}}\right)^m \leq \left(e^{-t^2p/4} + 4(\delta+\beta/4)\cdot e^{-pt(1+t)}\right)^m.$$

Finally, we can obtain that

$$\begin{aligned}
\mathbb{E}_{\delta}[Y_{\geq k}] &\leq \mathbb{E}_{\eta}[Y_{\geq k}] + \delta + 2\eta && \text{(Precision Consistency of Expectation)} \\
&\leq \left(e^{-t^2p/4} + 4(\delta+\beta/4)\cdot e^{-pt(1+t)}\right)^m + 200\eta m\cdot 2^m + \delta + 2\eta \\
&\leq \left(e^{-t^2p/4} + (4\delta+\beta)\cdot e^{-pt(1+t)}\right)^m + \delta + \beta,
\end{aligned}$$

where the last inequality follows if we set $\eta \triangleq \beta/(1000m\cdot 2^m)$. ☐

Using essentially the same proof with $\tau = e^{-\lambda} < 1$, we can obtain a Chernoff bound for the other side of the tail probability:

Theorem 3.37 (Chernoff bound II, LogLog form)**.** *The following sentence is provable in APX_1. Let $V = \{0,1\}$, $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, and $m \in \mathsf{LogLog}$. Let $X_1, \ldots, X_m$ be a sequence of explicit i.i.d. random variables over V that takes an nm-bit seed $(z_1, \ldots, z_m) \in (\{0,1\}^n)^m$. Let $p \triangleq \mathbb{E}_{\delta}[X_i] \in \mathbb{Q}$, and $Y_{\leq k}$ be the indicator variable of $X_1 + \cdots + X_m \leq k$ that takes an nm-bit seed. Then for $t \in \mathbb{Q}$, $0 \leq t \leq 1$, and $k \leq (1-t)pm$,*

$$\mathbb{E}_{\delta}[Y_{\leq k}] \leq \left(e^{-t^2p/4} + (4\delta+\beta)\cdot 2^{-pt(1-t)}\right)^m + \delta + \beta.$$

4 Theoretical Computer Science in APX_1

In this section, we formalize in APX_1 several fundamental results from algorithms, complexity theory, and related areas.

4.1 Yao's Distinguisher-to-Predictor Transformation

Theorem 4.1 (Yao's transformation)**.** APX_1 *proves the following statement. Let* $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $G : \{0,1\}^m \to \{0,1\}^n$ *be a multi-output circuit, and* $C \in B_n$ *be a circuit such that*

$$|\mathsf{P}_\delta(C \circ G) - \mathsf{P}_\delta(C)| \geq 2\delta + \varepsilon \tag{4.1}$$

for some $\varepsilon \in (0,1) \cap \mathbb{Q}$*, where* $C \circ G$ *is the* m*-input circuit defined as* $(C \circ G)(u) \triangleq C(G(u))$*.*

Then there is an index $i \in [n]$ *and a circuit* $P : \{0,1\}^{i-1} \to \{0,1\}$ *such that the following holds: Let* $T(u) \in B_m$ *be the circuit such that* $T(u) = 1$ *if* $P(G(u)_{<i}) = G(u)_i$ *(i.e.,* P *successfully predicts the* i*-th bit of* $G(u)$*), then*

$$\left|\mathsf{P}_\delta(T) - \frac{1}{2}\right| \geq \frac{\varepsilon}{4n} - (\delta + \beta).$$

Proof. We formalize the standard proof of Yao's lemma in APX_1 (see, e.g., [AB09, Chapter 9]). Fix $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $G : \{0,1\}^m \to \{0,1\}^n$, and circuit $C \in B_n$. For every index $i \in \{0, 1, \ldots, n\}$, we define the circuit $C_i : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}$ as follows:

- C_i parses its input as $(u, x) \in \{0,1\}^m \times \{0,1\}^n$.
- Let $z \triangleq G(u)_{\leq i} \circ x_{>i}$, i.e., the string where the first i bits agree with the first i bits of $G(u)$, and the remaining bits agree with the last $n - i$ bits of x. The circuit $C_i(u, x)$ then outputs $C(z)$.

Let $X_0, X_1, \ldots, X_n$ be the random variables over $\{0,1\}$ with seed length $m + n$, where X_i is the indicator variable of $C_i(j, x) = 1$. That is, X_i is defined by the tuple $(\{0,1\}, m + n, C_i)$.

Step 1: Gap between $\mathbb{E}_\eta[X_0]$ and $\mathbb{E}_\eta[X_n]$. We first argue that $|\mathbb{E}_\eta[X_0] - \mathbb{E}_\eta[X_n]|$ is large. Note that by Equation (4.1) and the Precision Consistency Axiom, we have

$$|\mathsf{P}_\eta(C \circ G) - \mathsf{P}_\eta(C)| \geq \varepsilon - 4\eta. \tag{4.2}$$

Note that X_0 is the random variable over $\{0,1\}$ with seed length $m + n$ that, on the seed $(u, x) \in \{0,1\}^m \times \{0,1\}^n$, ignores the first part of the seed and outputs $C(x)$. For any assignment $u \in \{0,1\}^m$ to the first part of the seed, it can be verified that $|\mathbb{E}_\eta[X_0|_u] - \mathsf{P}_\eta(C)| \leq 3\eta$, and thus by the *Averaging Argument for Expectation*, we know that

$$|\mathbb{E}_\eta[X_0] - \mathsf{P}_\eta(C)| \leq 6\eta.$$

Similarly, we can prove that

$$|\mathbb{E}_\eta[X_n] - \mathsf{P}_\eta(C \circ G)| \leq 6\eta. \tag{4.3}$$

This is because X_n is the random variable that, on the seed $(u, x) \in \{0,1\}^m \times \{0,1\}^n$, ignores the second part of the seed and outputs $C \circ G(u)$.

By combining Equation (4.2) and Equation (4.3), we have

$$|\mathbb{E}_\eta[X_0] - \mathbb{E}_\eta[X_n]| \geq \varepsilon - 16\eta \tag{4.4}$$

Step 2: Gap between $\mathbb{E}_\eta[X_i]$ and $\mathbb{E}_\eta[\overline{X_{i+1}}]$. As $|\mathbb{E}_\eta[X_0] - \mathbb{E}_\eta[X_n]| \geq \beta - 16\eta + \varepsilon$, we know that for some $1 \leq i \leq n$,

$$|\mathbb{E}_\eta[X_{i-1}] - \mathbb{E}_\eta[X_i]| \geq \frac{\varepsilon}{n} - \frac{16\eta}{n}. \tag{4.5}$$

In more detail, suppose towards a contradiction that this is not true. We can prove by induction on j that if $j \leq n$, $|\mathbb{E}_\eta[X_0] - \mathbb{E}_\eta[X_j]| < (j/n) \cdot (\varepsilon - 16\eta)$.

We will produce a predictor P for this index i. Recall that both X_{i-1} and X_i are random variables that parse their seeds as $(u, x) \in \{0,1\}^m \times \{0,1\}^n$, and

- X_{i-1} outputs $C(G(u)_{<i} \circ x_i \circ x_{>i})$;

- X_i outputs $C(G(u)_{<i} \circ G(u)_i \circ x_{>i})$.

Let $\overline{X_i}$ be the random variable that parses its input as $(u, x) \in \{0,1\}^m \times \{0,1\}^n$ and outputs $C(G(u)_{<i} \circ \overline{G(u)_i} \circ x_{>i})$. We will prove

$$\left|\mathbb{E}_\eta[X_i] - \mathbb{E}_\eta[\overline{X_i}]\right| > \frac{\varepsilon}{2n} - 100\eta \tag{4.6}$$

by rewriting both $\mathbb{E}_\eta[X_{i-1}]$ and $\mathbb{E}_\eta[X_i]$ in Equation (4.5).

Let X_{i-1}^0, X_{i-1}^1 be the random variables over $\{0,1\}$ that parse their seeds as $(u, x) \in \{0,1\}^m \times \{0,1\}^n$ satisfying that:

- X_{i-1}^0 outputs 1 if and only if $C(G(u)_{<i} \circ x_i \circ x_{>i}) = 1$ and $G(u)_i = x_i$.
- X_{i-1}^1 outputs 1 if and only if $C(G(u)_{<i} \circ \overline{G(u)_i} \circ x_{>i}) = 1$ and $G(u)_i \neq x_i$.

One can observe that for any assignment ρ to their seeds, $X_{i-1}|_\rho = X_{i-1}^0|_\rho + X_{i-1}^1|_\rho$, and thus by the *Linearity of Expectation*,

$$\left|\mathbb{E}_\eta[X_{i-1}] - \left(\mathbb{E}_\eta[X_{i-1}^0] + \mathbb{E}_\eta[X_{i-1}^1]\right)\right| \le 6\eta. \tag{4.7}$$

Similarly, let X_i^0, X_i^1 be the random variables over $\{0,1\}$ that parse their seeds as $(u, x) \in \{0,1\}^m \times \{0,1\}^n$ satisfying that:

- X_i^0 outputs 1 if and only if $C(G(u)_{<i} \circ x_i \circ x_{>i}) = 1$ and $G(u)_i = x_i$.
- X_i^1 outputs 1 if and only if $C(G(u)_{<i} \circ G(u)_i \circ x_{>i}) = 1$ and $G(u)_i \neq x_i$.

For any assignment ρ to their seeds, $X_i|_\rho = X_i^0|_\rho + X_i^1|_\rho$, and thus by the *Averaging Argument for Expectation*,

$$\left|\mathbb{E}_\eta[X_i] - \left(\mathbb{E}_\eta[X_i^0] + \mathbb{E}_\eta[X_i^1]\right)\right| \le 6\eta. \tag{4.8}$$

One can observe that X_i^0 and X_{i-1}^0 are exactly the same random variable. Moreover, we argue that

$$\left|\mathbb{E}_\eta[X_i] - 2 \cdot \mathbb{E}_\eta[X_i^1]\right| \le 20\eta. \tag{4.9}$$

(As a sanity check, $\mathbb{E}[X_i] = 2 \cdot \mathbb{E}[X_i^1]$ in exact expectation.) Assume for contradiction that this does not hold. By the *Averaging Argument for Expectation*, there is an assignment ρ to all but x_i such that

$$\left|\mathbb{E}_\eta[X_i|_\rho] - 2 \cdot \mathbb{E}_\eta[X_i^1|_\rho]\right| > 10\eta.$$

Note that both $X_i|_\rho$ and $X_i^1|_\rho$ have seed length 1, and we know that $\mathbb{E}[X_i|_\rho] = 2 \cdot \mathbb{E}[X_i^1|_\rho]$. This leads to a contradiction by the *Brute Force Counting Lemma*. Similarly, we can prove that

$$\left|\mathbb{E}_\eta[X_{i-1}] - 2 \cdot \mathbb{E}_\eta[X_{i-1}^1]\right| \le 10\eta. \tag{4.10}$$

By Equations (4.5) and (4.7) to (4.10), we have

$$\begin{aligned}
\left|\mathbb{E}_\eta[X_i] - \mathbb{E}_\eta[\overline{X_i}]\right| &\ge \left|\left(\mathbb{E}_\eta[X_{i-1}^0] + \mathbb{E}_\eta[X_{i-1}^1]\right) - \left(\mathbb{E}_\eta[X_i^0] + \mathbb{E}_\eta[X_i^1]\right)\right| - 12\eta && \text{(Equations (4.7) and (4.8))}\\
&\ge \left|\mathbb{E}_\eta[X_{i-1}^1] - \mathbb{E}_\eta[X_i^1]\right| - 12\eta\\
&\ge \frac{|\mathbb{E}_\eta[X_{i-1}] - \mathbb{E}_\eta[X_i]|}{2} - 60\eta && \text{(Equations (4.9) and (4.10))}\\
&> \frac{\varepsilon}{2n} - 100\eta. && \text{(Equation (4.5))}
\end{aligned}$$

Step 3: Producing the Predictor. We now prove that Equation (4.6) suffices to produce the predictor. Let $P : \{0,1\}^{i-1} \times \{0,1\}^n \to \{0,1\}$ and $T : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}$ be the following circuits: $P(v, x) \triangleq C(v \circ x_i \circ x_{>i}) \oplus x_i$, and $T(u, x)$ outputs 1 if and only if $P(G(u)_{<i}, x) = G(u)_i$.

We also define two circuits $T_0, T_1 : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}$ such that

- $T_0(u, x)$ outputs 1 if $C(G(u)_{<i} \circ G(u)_i \circ x_{>i}) = 0$ and $x_i = G(u)_i$.
- $T_1(u, x)$ outputs 1 if $C(G(u)_{<i} \circ \overline{G(u)_i} \circ x_{>i}) = 1$ and $x_i \neq G(u)_i$.

One can prove (in PV_1) that $T_0(u,x) = 1$ if and only if $T(u,x) = 1$ and $x_i = G(u)_i$, while $T_1(u,x) = 1$ if and only if $T(u,x) = 1$ and $x_i \neq G(u)_i$. Therefore for every $(u,x) \in \{0,1\}^m \times \{0,1\}^n$, $T(u,x) = T_0(u,x) + T_1(u,x)$. By considering their indicator variables and applying the *Linearity of Expectation*, one can prove that

$$|\mathsf{P}_\eta(T) - (\mathsf{P}_\eta(T_0) + \mathsf{P}_\eta(T_1))| \leq 50\eta. \tag{4.11}$$

Next, we argue that

$$|\mathbb{E}_\eta[X_i] - (1 - 2 \cdot \mathsf{P}_\eta(T_0))| \leq 20\eta. \tag{4.12}$$

Suppose, towards a contradiction, that Equation (4.12) does not hold. Let I_{T_0} be the indicator random variable of T_0, and we have $|\mathbb{E}_\eta[I_{T_0}] - \mathsf{P}_\eta(T_0)| \leq 3\eta$ (see Proposition 3.12). Both X_i and I_{T_0} have seed $(u,x) \in \{0,1\}^m \times \{0,1\}^n$. By the *Averaging Argument for Expectation*, there is an assignment ρ to all variables but x_i (i.e. the i-th bit of x) such that

$$|\mathbb{E}_\eta[X_i|_\rho] - (1 - 2 \cdot \mathbb{E}_\eta[I_{T_0}|_\rho])| > 10\eta.$$

Note that both $X_i|_\rho$ and $I_{T_0}|_\rho$ have seed length 1, and we know by the definition that[17]

$$|\mathbb{E}[X_i|_\rho] - (1 - 2 \cdot \mathbb{E}[I_{T_0}|_\rho])| = 0.$$

By the *Brute Force Counting Lemma*, we conclude a contradiction that thus proves Equation (4.12). Similarly, we can prove that

$$\left|\mathbb{E}_\eta[\overline{X_i}] - 2 \cdot \mathsf{P}_\eta(T_1)\right| \leq 20\eta. \tag{4.13}$$

By combining Equations (4.6) and (4.11) to (4.13), we have that

$$\begin{aligned}
\left|\mathsf{P}_\eta(T) - \frac{1}{2}\right| &\geq \left|\mathsf{P}_\eta(T_0) + \mathsf{P}_\eta(T_1) - \frac{1}{2}\right| - 50\eta && \text{(Equation (4.11))} \\
&\geq \left|\frac{1 - \mathbb{E}_\eta[X_i]}{2} + \frac{\mathbb{E}_\eta[\overline{X_i}]}{2} - \frac{1}{2}\right| - 80\eta && \text{(Equations (4.12) and (4.13))} \\
&\geq \frac{1}{2} \cdot \left|\mathbb{E}_\eta[X_i] - \mathbb{E}_\eta[\overline{X_i}]\right| - 80\eta \\
&\geq \frac{\varepsilon}{4n} - 180\eta. && \text{(Equation (4.6))}
\end{aligned}$$

Recall that T takes an input $(u,x) \in \{0,1\}^m \times \{0,1\}^n$. Let T_x be the circuit obtained by fixing the second part of its input to be $x \in \{0,1\}^n$. By considering its indicator variable (see Proposition 3.12) and applying the *Averaging Argument for Expectation*, there exists a string $x \in \{0,1\}^n$ such that

$$\left|\mathsf{P}_\eta[T_x] - \frac{1}{2}\right| \geq \frac{\varepsilon}{4n} - 200\eta.$$

By the definition, one can see that T_x evaluating to 1 means the predictor $P_x(v) \triangleq C(v \circ x \circ x_{>i}) \oplus x_i$ correctly predicts the i-th bit of $G(u)$. This concludes the proof by applying the *Global Consistency* of approximate counting, the Precision Consistency Axiom, and setting $\eta \triangleq \beta/400$. □

4.2 Schwartz-Zippel Lemma

Before stating the Schwartz-Zippel lemma, we need to clarify the formalization of finite fields and polynomials. A finite field $\mathbb{F}$ is said to be feasible if $|\mathbb{F}| \in \mathsf{Log}$, e.g., $\mathbb{F} = \mathbb{F}_p$ for some prime number $p \in \mathsf{Log}$. It is verified in [Jeř05, Section 4.3.3] that for any feasible field, the field elements can be encoded such that (1)

[17] Indeed, since after fixing ρ the only randomness is x_i and $X_i \restriction_\rho$ is independent of x_i, we have $I_{T_0} \restriction_\rho = \mathbf{1}\{x_i = G(u)_i\}\, \mathbf{1}\{X_i \restriction_\rho = 0\} = \mathbf{1}\{x_i = G(u)_i\}\,(1 - X_i \restriction_\rho)$. Consequently, $\mathbb{E}[I_{T_0} \restriction_\rho] = \frac{1}{2} \cdot (1 - X_i \restriction_\rho) = \frac{1}{2} \cdot \Big(1 - \mathbb{E}[X_i \restriction_\rho]\Big)$, which implies that $\mathbb{E}[X_i \restriction_\rho] = 1 - 2 \cdot \mathbb{E}[I_{T_0} \restriction_\rho]$.

the field operations can be implemented by PV function symbols, and (2) field axioms can be proved in PV. For simplicity, we identify an element in $\mathbb{F}$ and its encoding as a string.

Fix a feasible field $\mathbb{F}$. A degree-d univariate polynomial $p \in \mathbb{F}[x]$ can be defined by a list of coefficients $c_0, c_1, \ldots, c_d \in \mathbb{F}$ such that $p(x) \triangleq c_0 + c_1 x + \cdots + c_d x^d$. A polynomial is said to be nonzero if any of $c_0, c_1, \ldots, c_d$ is nonzero.

Proposition 4.2 (Implicit in [Jeř05, Lemma 4.3.6]). *It is provable in* PV *that any nonzero degree-d polynomial $p \in \mathbb{F}[x]$ has at most d roots.*

Proposition 4.3 (Implicit in [Jeř05]). *It is provable in* PV *that for any $d < |\mathbb{F}|$, any distinct $x_1, \ldots, x_d \in \mathbb{F}$, and $y_1, \ldots, y_d \in \mathbb{F}$, there is a degree-$d$ polynomial $p \in \mathbb{F}[x]$ such that $p(x_i) = y_i$ for every $i \in [d]$.*

Let $m, d \in \mathsf{Log}$. We say that a circuit C *computes an m-variate function in* $\mathbb{F}$, denoted by $C : \mathbb{F}^m \to \mathbb{F}$, if for every $x_1, \ldots, x_m \in \mathbb{F}$, $C(x_1, \ldots, x_m) \in \mathbb{F}$. Depending on the encoding of field elements, some circuit may not compute an m-variate function in $\mathbb{F}$ as it outputs a string that does not encode any field element. We say that $C : \mathbb{F}^m \to \mathbb{F}$ has *individual degree at most* d if for every $i \in [m]$ and any assignment ρ to all but the i-th variable, there is a polynomial $p_{i,\rho} \in \mathbb{F}[x]$ of degree at most d such that $p_{i,\rho}(x) = C|_\rho(x)$ for $x \in \mathbb{F}$.

Theorem 4.4 (Schwartz-Zippel Lemma). APX_1 *proves the following statement. Let $\mathbb{F}$ be a feasible field such that each field element is encoded by a string of length $b \in \mathsf{LogLog}$. Let $m, d \in \mathsf{Log}$, $d < |\mathbb{F}|$, and $C : \mathbb{F}^m \to \mathbb{F}$ be a circuit of individual degree at most d. Let $T_C : (\{0,1\}^b)^m \to \{0,1\}$ be the circuit that given $(x_1, \ldots, x_m) \in \{0,1\}^b$, it accepts if $x_i \in \mathbb{F}$ for each $i \in [m]$, and $C(x_1, \ldots, x_m) = 0$.*

Suppose that for some $\vec{z} \in \mathbb{F}^m$, $C(\vec{z}) \neq 0$. Then for every $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $\mathsf{P}_\delta(T_C) \leq md/|\mathbb{F}| + \delta + \beta$.

Proof. The key idea is to formalize the proof of Atserias and Tzameret [AT25] in APX_1. We argue in APX_1. Fix any feasible field $\mathbb{F}$, $b \in \mathsf{LogLog}$, $m, d \in \mathsf{Log}$, $C : \mathbb{F}^m \to \mathbb{F}$, $\vec{z} = (z_1, \ldots, z_m) \in \mathbb{F}_m$, and $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, as in the statement.

Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. For every $i \in \{0, 1, \ldots, m\}$, we define $T_C^i : (\{0,1\}^b)^m \to \{0,1\}$ as follows: Given $(x_1, \ldots, x_m) \in \{0,1\}^b$, it accepts if $x_i \in \mathbb{F}$ for each $i \in [m]$ and $C(x_1, \ldots, x_i, z_{i+1}, \ldots, z_m) = 0$. It is clear that $T_C^0 \equiv \mathsf{Null}$, and $T_C^m \equiv T_C$.

Let $X_0, X_1, \ldots, X_m$ be the indicator random variables for $T_C^0, T_C^1, \ldots, T_C^m$. We will prove that for every $i \geq 1$,

$$\mathbb{E}_\eta[X_i] - \mathbb{E}_\eta[X_{i-1}] \leq d/|\mathbb{F}| + 10\eta. \tag{4.14}$$

Assume for contradiction that it is not the case. By *Averaging Argument for Expectation*, there is an assignment ρ to all but the i-th part x_i of the seed such that

$$\mathbb{E}_\eta[X_i|_\rho] - \mathbb{E}_\eta[X_{i-1}|_\rho] > d/|\mathbb{F}| + 10\eta - 6\eta > 0. \tag{4.15}$$

Note that $X_{i-1}|_\rho$ is the indicator random variable of $T_C^{i-1}|_\rho$, and by the definition, $T_C^{i-1}|_\rho$ is a constant circuit that does not read the seed. Therefore it falls into one of the two cases:

- Suppose that $T_C^{i-1}|_\rho \equiv \mathsf{True}$. We also know that the circuit $\mathsf{EQ}(T_C^{i-1}|_\rho, 1)$ that given $x \in \{0,1\}^b$, outputs 1 if and only if $T_C^{i-1}(x) = 1$ is also functionally equivalent to True. One can prove by the definition that
$$\mathbb{E}_\eta[X_{i-1}|_\rho] = \mathsf{P}_\eta(\mathsf{EQ}(T_C^{i-1}|_\rho, 1)) = 1,$$
where the second inequality follows from the Boundary Axiom. This leads to a contradiction to Equation (4.15).
- Otherwise, $T_C^{i-1}|_\rho \equiv \mathsf{Null}$. Let $\rho = (y_1, \ldots, y_{i-1}, *, y_{i+1}, \ldots, y_m)$. Recall that as the individual degree of C is at most d, there is a polynomial $p \in \mathbb{F}[x]$ such that $p \equiv C(y_1, \ldots, y_{i-1}, \cdot, z_{i+1}, \ldots, z_m)$. It follows from $T_C^{i-1}|_\rho \equiv \mathsf{Null}$ and Equation (4.15) that
$$p(z_i) = C(y_1, \ldots, y_{i-1}, z_i, z_{i+1}, \ldots, z_m) \neq 0,$$

and thus p is a nonzero polynomial. By Proposition 4.2, it has at most d roots. Moreover, as the input length of $T_C^{i-1}|_\rho$ is $b \in \mathsf{LogLog}$, we can prove by Proposition 3.12 and *Brute Force Counting Lemma* that

$$\mathbb{E}_\eta[X_i|_\rho] \leq \mathsf{P}_\eta(T_C^i) + 2\eta \leq d/|\mathbb{F}| + 4\eta.$$

This leads to a contradiction to Equation (4.15).

Finally, as $T_C^0 \equiv \mathsf{Null}$, we know that $\mathbb{E}_\eta[X_0] \leq 10\eta$. By induction on i (using the n-Induction Axiom) and Equation (4.14), we can prove that

$$\mathbb{E}_\eta[X_m] \leq md/|\mathbb{F}| + 10\eta \cdot (m+1). \tag{4.16}$$

Subsequently, we have

$$\begin{aligned}
\mathsf{P}_\delta(T_C) &\leq \mathsf{P}_\eta(T_C) + \delta + 2\eta && \text{(Precision Consistency Axiom)}\\
&\leq \mathsf{P}_\eta(T_C^m) + \delta + 5\eta && \text{(\textit{Global Consistency})}\\
&\leq \mathbb{E}_\eta(X_m) + \delta + 8\eta && \text{(Proposition 3.12)}\\
&\leq md/|\mathbb{F}| + \delta + 10\eta \cdot (m+1) + 8\eta\\
&\leq md/|\mathbb{F}| + \delta + \beta,
\end{aligned}$$

where the last inequality follows by taking $\eta \triangleq \beta/(10m+20)$. □

4.3 Linear Hashing

The linear hash function $x \mapsto Ax \bmod 2$ is one of the simplest constructions of hash functions. The following theorem formalizes that linear hashing is an (almost) universal hash function.

Theorem 4.5 (Universality of linear hashing)**.** APX_1 *proves the following statement. Let* $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$*. For every* $x, y \in \{0,1\}^n$*, let* $T_{x,y} : \{0,1\}^{nm} \to \{0,1\}$ *be the circuit that parses its input as a Boolean matrix* $A \in \{0,1\}^{m\times n}$ *and outputs* 1 *if and only if* $Ax \equiv Ay \pmod 2$*. Then for all distinct* $x, y \in \{0,1\}^n$*,* $\mathsf{P}_\delta(T_{x,y}) \leq \delta + \beta + (1/2+\beta)^m$*.*

Proof. We argue in APX_1. Fix $n, m, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $x, y \in \{0,1\}^n$, and let $T_{x,y}$ be the circuit defined above. Suppose that $x \neq y$. We will prove that $\mathsf{P}_\delta(T_{x,y}) \leq \delta + \beta + (1/2+\beta)^m$.

We first upper bound the probability when $m = 1$. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later, and $C : \{0,1\}^n \to \{0,1\}$ be the circuit that given $c \in \{0,1\}^n$, it outputs $\langle c, x-y\rangle \bmod 2$. We will prove that $\mathsf{P}_\eta(\neg C) \leq 1/2 + 6\eta$.

Suppose, towards a contradiction, that $\mathsf{P}_\eta(\neg C) > 1/2 + 6\eta$. As $x \neq y$, there is an index $i \in [n]$ such that $x_i \neq y_i$. Fix the index i. Let X be the indicator random variable of $\neg C$; by Proposition 3.12, we know that $\mathbb{E}_\eta[X] > 1/2 + 6\eta - 3\eta = 1/2 + 3\eta$. Subsequently, by *Averaging Argument for Expectation*, there is an assignment ρ to all but the i-th bit of the random seed such that $\mathbb{E}_\eta[X|_\rho] > 1/2$. However, as the seed length of $X|_\rho$ is 1 and the probability is 1/2, this violates the *Brute Force Counting Lemma*.

Let $C^{\vee m} : \{0,1\}^{nm} \to \{0,1\}$ denote the circuit $C^{\vee m}(x_1, \dots, x_m) \triangleq \bigvee_{i\in[m]} C(x_i)$. By *One-sided Error Reduction Lemma*, we have

$$\mathsf{P}_\eta(\neg C^{\vee m}) \leq (1/2 + 8\eta)^m + 2\eta.$$

It can be observed that $\neg C^{\vee m}$ is functionally equivalent to $T_{x,y}$, and thus by the *Global Consistency* of approximate counting,

$$\mathsf{P}_\eta(T_{x,y}) \leq \mathsf{P}_\eta(\neg C^{\vee m}) + 3\eta \leq (1/2 + 8\eta)^m + 5\eta.$$

By the Precision Consistency Axiom, we have

$$\mathsf{P}_\delta(T_{x,y}) \leq \mathsf{P}_\eta(T_{x,y}) + \delta + 2\eta \leq (1/2 + 8\eta)^m + \delta + 7\eta.$$

This completes the proof when we take $\eta \triangleq \beta/20$. □

4.4 Lower Bounds for Parity Against AC^0 Circuits

The first result of this section is a formalization in APX_1 of an average-case lower bound for the Parity function $\oplus_n$ against AC^0. Our proof is based on a technique due to Furst, Saxe, and Sipser [FSS84]. Previous formalizations of the lower bound[18] due to Müller and Pich [MP20] (following [FSS84]) and Krajíček [Kra95, Theorem 15.2.3] (following Razborov's proof of the switching lemma [Raz95]) require the theory $\mathsf{APC}_1 = \mathsf{PV}_1 + \mathsf{dWPHP}(\mathsf{PV})$.[19]

Theorem 1.6 (Average-Case AC^0 Lower Bound for $\oplus_n$ in APX_1)**.** *For all constants $k, d \geq 1$, there exists a constant $n_0 \geq 1$ such that APX_1 proves the following statement. Let $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $n > n_0$, and $C : \{0,1\}^n \to \{0,1\}$ be an AC^0_d circuit of size at most n^k. Let $T_C : \{0,1\}^n \to \{0,1\}$ be the circuit that, given $x \in \{0,1\}^n$, outputs 1 if and only if $C(x) = \oplus_n(x)$. Then*

$$\mathsf{P}_\delta(T_C) \leq \frac{1}{2} + \frac{1}{n^k} + \delta + \beta. \tag{1.1}$$

The main technical challenge is to avoid "encoding-based counting argument" that rely on $\mathsf{dWPHP}(\mathsf{PV})$, which is not available in APX_1. The encoding-based counting argument is used in both [MP20] and Razborov's [Raz95] proof of the switching lemma. This was partially addressed by Agrawal et al. [AAI+01] (see also [Agr01])[20], which presented a deterministic polynomial-time algorithm that outputs a suitable restriction given by the switching lemma. As one of our contributions, we show that the correctness of the algorithm in Agrawal et al. [AAI+01] can be established in PV_1. This, together with tools developed in Section 3), allow us to formalize the *average-case* lower bound in APX_1.

Interestingly, using the same technique, we further show that the *worst-case* lower bound $\oplus_n \notin \mathsf{AC}^0$ can be formalized in PV_1. This resolves an open problem from [MP20].

Theorem 1.7 (Worst-Case AC^0 Lower Bound for $\oplus_n$ in PV_1)**.** *For all constants $k, d \geq 1$, there exists a constant $n_0 \geq 1$ such that PV_1 proves the following statement. For every $n \in \mathsf{Log}$, $n > n_0$, and AC^0_d circuit $C : \{0,1\}^n \to \{0,1\}$ of size at most n^k, there exists a string $x \in \{0,1\}^n$ such that $C(x) \neq \oplus_n(x)$.*

Notation. We let n denote the number of input variables. A k-CNF is a propositional formula of the form $C_1 \wedge C_2 \wedge \cdots \wedge C_m$, where each clause C_i is a disjunction of at most k literals, i.e., variables or their negations. Similarly, a k-DNF is a disjunction of terms, where each term is a conjunction of at most k literals. We say that a formula is a k-NF if it is either a k-CNF or a k-DNF. A clause[21] C of a k-NF can be described by its type (i.e. $\wedge$ or $\vee$) and two subsets $S_C^+, S_C^- \subseteq [n]$ of size $|S_C^+| + |S_C^-| \leq k$, where S_C^+ denotes the ID of variables in the clause, and S_C^- denotes the ID of negations of variables in the clause. Let $S_C \triangleq S_C^+ \cup S_C^-$.

Note that we may assume without loss of generality that $S_C^+ \cap S_C^- = \varnothing$, as otherwise the clause will be either always 0 or always 1.

For simplicity, we assume without loss of generality that AC^0 circuits satisfy the following properties:

- The circuit is *layered*, i.e., gates in each layer are fed by gates only in previous layer.
- All negation gates are pushed to be directly above input variables. Equivalently, there is no negation gate inside the circuit, i.e., gates in the first layer can be fed by literals, i.e., input variables or their negations.

An AC^0 circuit satisfying these properties is called a *well-formed* circuit. Note that an arbitrary AC^0 circuit can be transformed into a well-formed circuit with only a polynomial size overhead, and the correctness of the transformation can be proved in PV_1. We start directly from well-formed circuits to simplify calculations.

[18] Both results consider only worst-case lower bounds, but it can be verified that an average-case lower bound follows from a similar argument when formalized appropriately (in the style of Jeřábek [Jeř05, Jeř07a]).

[19] Note that both results automatically give a formalization of the worst-case lower bound in $\mathsf{PV}_1 + \mathsf{rWPHP}(\mathsf{PV})$, as APC_1 is $\forall\Sigma_1^b$-conservative over $\mathsf{PV}_1 + \mathsf{rWPHP}(\mathsf{PV})$ [Jeř04, Jeř07a] and the worst-case lower bound can be formalized as a $\forall\Sigma_1^b$-sentence (see, e.g., [MP20, Theorem 1.1]).

[20] As mentioned in [AAI+01], the result was known to Ajtai and Wigderson (unpublished).

[21] In the context of k-NFs, we use "clause" to refer to a subformula, regardless of the connective.

4.4.1 Deterministic Selection of Subset for Restriction

We start by stating two lemmas that formalize the core combinatorial property used in the proof of Furst, Saxe, and Sipser [FSS84]. Similar functions are used implicitly in [AAI+01]. Here we use potential functions in order to implement a derandomization via the method of conditional expectations. The approach provides both a feasible algorithm and a feasible proof.

Lemma 4.6 (Potential Function for Small Sets)**.** *The following sentence is provable in* PV_1 *for every constant* $c \geq 1$*. Let* $n, t \in \mathsf{Log}$ *with* $t \leq n$*,* $s, m \in \mathsf{LogLog}$*,* $s \leq m$*,* $p \triangleq t/n \in \mathbb{Q}$*, and* $S_1, \ldots, S_m \subseteq [n]$ *be disjoint sets of size at most* c *such that* $pc < 1$*. There is a circuit* $\Phi_{\mathsf{s}} : \{*, \circ\}^{\leq n} \to \mathbb{Q} \cap [0, 1]$ *such that the following holds.*

- (Initial Condition)*.* $\Phi_{\mathsf{s}}(\varepsilon) \leq m^s (pc)^{m-s}$*.*
- (Recursion Condition)*. For every* $x \in \{*, \circ\}^i$*,* $i < n$*, we have*

$$\Phi_{\mathsf{s}}(x) = p \cdot \Phi_{\mathsf{s}}(x*) + (1-p) \cdot \Phi_{\mathsf{s}}(x\circ).$$

- (Termination Condition)*. For every* $x \in \{*, \circ\}^n$*,* $\Phi_{\mathsf{s}}(x) \in \{0, 1\}$*. Moreover, let* $T_x \triangleq \{i \in [n] \mid x_i = \circ\}$*. Then* $\Phi_{\mathsf{s}}(x) = 0$ *if and only if there are at least* s *subsets* $S \in \{S_1, \ldots, S_m\}$ *such that* $S \subseteq T_x$*.*

Proof Sketch. We argue in PV_1. Let $n, t, s \in \mathsf{Log}$, $m \in \mathsf{LogLog}$, $p \triangleq t/n$, and $S_1, \ldots, S_m \subseteq [n]$. The circuit Φ_{s} is defined as follows: Given $x \in \{*, \circ\}^i$ for some $0 \leq i \leq n$, let $T_x \triangleq \{j \leq i \mid x_j = \circ\}$. It outputs

$$\Phi_{\mathsf{s}}(x) \triangleq \sum_{\alpha \subseteq [m], |\alpha| < s} \phi(x, \alpha),$$

where

$$\phi(x, \alpha) \triangleq \begin{cases} 0 & \exists j \in \alpha \ \exists k \in [i] \ (k \in S_j \wedge x_k = *) \\ \prod_{j \in [\alpha]} (1-p)^{|S_j \setminus T_x|} \prod_{j' \in [m] \setminus \alpha} \psi(x, j') & \text{otherwise} \end{cases},$$

$$\psi(x, j') \triangleq \begin{cases} 1 & \exists k \in [i] \ (k \in S_{j'} \wedge x_k = *) \\ 1 - (1-p)^{|S_{j'} \setminus T_x|} & \text{otherwise} \end{cases}.$$

For instructive purposes, we mention the combinatorial interpretation of the functions (which is not a part of the PV_1 proof): Given any $x \in \{*, \circ\}^i$, we randomly assign $x_{i+1}, \ldots, x_n$ independently to $*$ with probability p and to $\circ$ with probability $1-p$. Let $T'_x \triangleq \{i \in [n] \mid x_i = \circ\}$. Then

- $\psi(x, j')$ is the probability that $S_{j'} \nsubseteq T'_x$.
- $\phi(x, \alpha)$ is the probability that $S_j \subseteq T'_x$ if and only if $j \in \alpha$.
- $\Phi_{\mathsf{s}}(x)$ is the probability that at most $s-1$ subsets $S \in \{S_1, \ldots, S_m\}$ satisfy $S \subseteq T_x$.

We come back to the PV_1 proof. The recursion condition and termination condition can be verified by a tedious but straightforward calculation, which we omit here. To prove the initial condition, notice that

$$\begin{aligned} \psi(\varepsilon, j') &= 1 - (1-p)^{|S'_j|} \leq 1 - (1-p)^c \leq pc \\ \phi(\varepsilon, \alpha) &\leq \prod_{j' \in [m] \setminus \alpha} \psi(\varepsilon, j') \leq (pc)^{m-|\alpha|} \\ \Phi_{\mathsf{s}}(\varepsilon) &\leq \sum_{0 \leq j < s} \binom{m}{j} (pc)^{m-j} \leq m^s (pc)^{m-s} \end{aligned}$$

The argument can be implemented in PV_1 using that $s, m \in \mathsf{LogLog}$ and c is constant. This completes the proof. ☐

Lemma 4.7 (Potential for General Set Systems)**.** *For every choice of constants $k, c \geq 1$, there are constants $b, n_0 \geq 1$ such that the following sentence is provable in PV_1. Let $n, t, m \in \mathsf{Log}$, $n > n_0$, $p \triangleq t/n \in \mathbb{Q}$, and $S_1, S_2, \ldots, S_m \subseteq [n]$ be nonempty subsets of size at most c. Then there is a circuit $\Phi : \{*, \circ\}^{\leq n} \to \mathbb{Q} \cap [0,1]$ such that the following holds.*

- (Initial Condition). *If $t \leq \sqrt{n}$, then $\Phi(\varepsilon) \leq n^{-k}$.*
- (Recursion Condition). *For every $x \in \{*, \circ\}^i$, $i < n$, we have*

$$\Phi(x) = p \cdot \Phi(x*) + (1-p) \cdot \Phi(x\circ).$$

- (Termination Condition). *For every $x \in \{*, \circ\}^n$, $\Phi(x) \in \{0,1\}$. Moreover, for $T_x \triangleq \{i \in [n] \mid x_i = \circ\}$, if $\Phi(x) = 0$, then one of the following conditions holds:*
  1. *$|S_1 \cup S_2 \cup \cdots \cup S_m \setminus T_x| \leq b$.*
  2. *There are disjoint nonempty sets $V_1, \ldots, V_\ell \subseteq [n]$ that are subsets of ℓ distinct sets among $S_1, \ldots, S_m$, such that $V_i \subseteq T_x$ for every $i \in [\ell]$, where $\ell \geq k \ln n$.*

  Moreover, the disjoint sets $V_1, \ldots, V_\ell$ can be obtained by a PV function given $S_1, \ldots, S_m$ and $x \in \{*, \circ\}^n$.

Proof. Fix any constants $k, c \geq 1$ and let $b, n_0 \geq 1$ be constants to be determined later. We argue in PV_1. Fix $n, t, m \in \mathsf{Log}$, $p \triangleq t/n$, and $S_1, \ldots, S_m \subseteq [n]$.

Disjoint Set Decomposition. Consider the iterative algorithm: Let $\mathcal{U}_0 \leftarrow \{S_1, \ldots, S_m\}$. In the i-th step, where $i \geq 1$, we choose a maximal set $\mathcal{V}_i \subseteq \mathcal{U}_{i-1}$ such that the sets in $\mathcal{V}_i$ are disjoint, and compute $\mathcal{U}_i$ as follows:

- Let $\mathcal{U}_i \leftarrow \varnothing$. For every $S \in \mathcal{U}_{i-1}$, we include $S \setminus (\bigcup_{S \in \mathcal{V}_i} S)$ in $\mathcal{U}_i$ if this set is nonempty.

Note that each set remaining in $\mathcal{U}_i$ must be a subset of some $S_1, \ldots, S_m$. Moreover, it must be of size at most $c - i$, as the maximal set $\mathcal{V}_i$ intersects with each set in $\mathcal{U}_{i-1}$ (otherwise it is not maximal). The algorithm terminates if no set is added to $\mathcal{U}_i$ during an iteration.

Let $d \leq c$ and $\mathcal{V}_1, \ldots, \mathcal{V}_d$ be the sets obtained by the algorithm. One can prove that the union of the sets in $\mathcal{V}_1, \ldots, \mathcal{V}_d$ covers $S_1 \cup \cdots \cup S_m$. Furthermore, it can be verified that for each $i \in [d]$ and $V_1, \ldots, V_\ell \in \mathcal{V}_i$, there are distinct $i_1, \ldots, i_\ell \in [m]$ such that $V_1 \subseteq S_{i_1}, \ldots, V_\ell \subseteq S_{i_\ell}$.

Wide Case. We first consider the case that for some $i \in [d]$, $\mathcal{V}_i$ contains at least $\ell \triangleq 2k \ln n$ sets. Fix i to be the smallest number satisfying this. Let $V_1, \ldots, V_\ell$ be the first ℓ sets in $\mathcal{V}_i$. Let $\Phi_{\mathsf{s}}(x)$ be the potential function in Lemma 4.6 for $V_1, \ldots, V_\ell$ and $s \triangleq \ell/2$ (note that $\ell \in \mathsf{LogLog}$). We define $\Phi(x) \triangleq \Phi_{\mathsf{s}}(x)$.

It then suffices to verify that the three required properties hold.

- (*Initial Condition*). By Lemma 4.6, we know that

$$\Phi(\varepsilon) \leq \ell^{\ell/2} \cdot (pc)^{\ell/2} \leq \left(\frac{bkc \ln n}{\sqrt{n}}\right)^{\ell/2} \leq n^{-k},$$

  where we assume a choice of $b \geq 2$ and a sufficiently large n, which can be ensured by setting n_0 as a large constant.
- (*Recursion Condition*). It follows from the recursion condition in Lemma 4.6.
- (*Termination Condition*). For $x \in \{*, \circ\}^n, \Phi(x) \in \{0,1\}$ by Lemma 4.6. Let $T_x \triangleq \{i \in [n] \mid x_i = \circ\}$. Suppose that $\Phi(x) = 0$. We know that there are at least $\ell/2 \geq k \ln n$ sets V among $V_1, \ldots, V_\ell$ such that $V \subseteq T_x$. This satisfies the second termination condition. Moreover, the $k \ln n$ sets can be obtained by a PV function given $S_1, \ldots, S_m$ and $x \in \{*, \circ\}^n$, using the algorithm that constructs $\mathcal{V}_1, \ldots, \mathcal{V}_d$.

Narrow Case. Now we consider the case that $\mathcal{V}_i \le 2k \ln n$ sets for every $i \in [d]$. Note that the union of the sets in $\mathcal{V}_1, \ldots, \mathcal{V}_d$ covers $S_1 \cup \cdots \cup S_m$. We know that

$$|S_1 \cup \cdots \cup S_m| = \left| \bigcup_{i \in [d]} \bigcup_{V \in \mathcal{V}_i} V \right| \le d \cdot (2k \ln n) \cdot c \le 2kc^2 \ln n.$$

Recall that $b \ge 1$ is a constant to be determined. Let $S \triangleq S_1 \cup \cdots \cup S_m$. We define the potential function $\Phi(x)$ as follows. Given $x \in \{*, \circ\}^i$ for $i < n$, let $q^* \triangleq |\{j \in [i] \cap S \mid x_j = *\}|$, $q^\circ \triangleq |\{j \in [i] \cap S \mid x_j = \circ\}|$, and $q^\diamond \triangleq |S| - q^* - q^\circ$. Then

$$\Phi(x) \triangleq \sum_{j=\max\{b+1, q^*\}}^{|S|-q^\circ} \binom{q^\diamond}{j - q^*} p^{j-q^*} (1-p)^{q^\diamond - j + q^*}. \tag{4.17}$$

For instructive purposes, we mention that the combinatorial interpretation of the function is as follows. We randomly assign $x_{i+1}, \ldots, x_n$ independently to $*$ with probability p and to $\circ$ with probability $1-p$. Then $\Phi(x)$ is the probability that the number of indices $j \in [n] \cap S$ such that $x_j = *$ is at least $b+1$. The combinatorial interpretation is *not* a part of the PV_1 proof.

Note that $\Phi(x) \ge 0$ as each term is non-negative, and $\Phi(x) \le 1$ by the *Binomial Theorem*. It suffices to verify the three required properties.

- (*Initial Condition*). Note that

$$\Phi(\varepsilon) \le \sum_{j=b+1}^{|S|} \binom{|S|}{j} p^j \le \sum_{j=b+1}^{|S|} |S|^j p^j \le \sum_{j=b+1}^{|S|} \left(\frac{2kc^2 \ln n}{\sqrt{n}} \right)^j \le n^{-k},$$

  where the last inequality holds if we set $b \ge 10k$ and n_0 to be sufficiently large.
- (*Recursion Condition*). Fix any $x \in \{*, \circ\}^i$ for $i < n$, we consider whether $i+1 \in S$. If not, we have that $\Phi(x) = \Phi(x*) = \Phi(x\circ)$ and the equation holds. Otherwise, let $q^* \triangleq |\{j \in [i] \cap S \mid x_j = *\}$, $q^\circ \triangleq \{j \in [i] \cap S \mid x_j = \circ\}$, and $q^\diamond \triangleq |S| - q^* - q^\circ$. We can see that

$$\Phi(x*) \triangleq \sum_{j=\max\{b+1, q^*+1\}}^{|S|-q^\circ} \binom{q^\diamond - 1}{j - q^* - 1} p^{j-q^*-1} (1-p)^{q^\diamond - j + q^*};$$

$$\Phi(x\circ) \triangleq \sum_{j=\max\{b+1, q^*\}}^{|S|-q^\circ-1} \binom{q^\diamond - 1}{j - q^*} p^{j-q^*} (1-p)^{q^\diamond - 1 - j + q^*}.$$

  It follows from Equation (3.45) that $\Phi(x) = p \cdot \Phi(x*) + (1-p) \cdot \Phi(x\circ)$.
- (*Termination Condition*). For $x \in \{*, \circ\}^n$, let $q^*, q^\circ, q^\diamond$ be defined as above. We have $q^\diamond = 0$ and $q^* + q^\circ = |S|$. In that case, one can observe that if $q^* > b$, then $\Phi(x) = 1$, and otherwise $\Phi(x) = 0$. Moreover, we know by the definition of q^* that if $\Phi(x) = 0$, for $T_x \triangleq \{i \in [n] \mid x_i = \circ\}$, we have $|S \setminus T_x| = q^* \le b$. This satisfies the first termination condition.

This completes the proof. □

We are now ready to state the *Subset Selection Lemma*.

Lemma 4.8 (Subset Selection Lemma)**.** *For every $k, c \ge 1$, there are constants $b, n_0 \ge 1$ such that the following sentence is provable in PV_1. Let $n, t, \ell \in \mathsf{Log}$, $n > n_0$, $\ell \le n^k$, and $F_1, F_2, \ldots, F_\ell$ be c-NFs over n input variables. If $t \le \sqrt{n}$, there exists a subset $T \subseteq [n]$ of size at most $n - t$ such that for every $i \in [\ell]$, at least one of the following conditions hold.*

- *If we fix the j-th variable for every $j \in T$, F_i is fed by at most b literals.*

- *There are $m' \geq k \ln n$ disjoint non-empty clauses $C'_1, \ldots, C'_{m'}$ such that (1) each C'_j is a sub-clause of a different clause in F_i; (2) $S_{C'_j} \subseteq T$.*

Moreover, the subset T and the clauses $C'_1, \ldots, C'_m$ for each i are computed by a PV *function given $F_1, \ldots, F_\ell$ and $1^n, 1^t, 1^\ell$.*

Proof. Fix any $k, c \geq 1$ and let $b, n_0 \geq 1$ be determined later. We argue in PV_1. Fix $n, t \in \mathsf{Log}$, $\ell \leq n^k$, $p \triangleq t/n$, and c-NFs $F_1, \ldots, F_\ell$ over n variables.

Fix any $i \in [\ell]$. Suppose that F_i has m_i clauses, and let C_{ij} be the j-th clause of F_i. Let $S_{ij} \triangleq S_{C_{ij}}$ be the subset of variables which or whose negation feeds C_{ij}. For each $i \in [\ell]$, consider the sets $S_{i1}, \ldots, S_{im_i}$; by Lemma 4.7 (using $3k$ instead of k), there are constants b', n'_0 and a potential function $\Phi_i : \{*, \circ\}^{\leq n} \to \mathbb{Q} \cap [0, 1]$ such that the following conditions hold.

- (*Initial Condition*). $\Phi_i(\varepsilon) \leq n^{-3k}$;
- (*Recursion Condition*). $\Phi_i(x) = p \cdot \Phi_i(x*) + (1-p) \cdot \Phi_i(x\circ)$.
- (*Termination Condition*). For $x \in \{*, \circ\}^n$, $\Phi_i(x) \in \{0, 1\}$. Moreover, let $T_x \triangleq \{i \in [n] \mid x_i = \circ\}$. If $\Phi_i(x) = 0$, then one of the following two conditions holds:
  1. $|S_{i1} \cup \cdots \cup S_{im_i} \setminus T_x| \leq b$. This effectively means that if we fix j-th variable for every $j \in T_x$, then F_i is fed by at most b literals.
  2. There are disjoint nonempty sets $V_1, \ldots, V_{\ell_i} \subseteq [n]$ that are subsets of ℓ_i distinct sets among $S_{i1}, \ldots, S_{im_i}$, such that $V_i \subseteq T_x$ for every $i \in [\ell]$, where $\ell_i \geq 3k \ln n \geq k \ln n$. This means that there are $m' = \ell_i \geq k \ln n$ disjoint clauses $C'_1, \ldots, C'_{m'}$, each of which is a sub-clause of a clause in F_i, such that $S_{C'_j} \subseteq T$.

  Therefore, if $\Phi_i(x) = 0$, then the clause satisfies the required property if we choose $T \triangleq T_x$.

Consider the potential function $\Phi(x)$ as follows. Given any $x \in \{*, \circ\}^i$, $i \leq n$, let q° be the number of $\circ$'s in x. We define

$$\Phi(x) \triangleq q^\circ + (1-p)(n-i) + n \cdot \sum_{i \in [\ell]} \Phi_i(x).$$

We can show that $\Phi(x)$ satisfies the following conditions:

- (*Initial Condition*). $\Phi(\varepsilon) \leq (1-p)n + n \cdot \ell \cdot n^{-3k} < (1-p)n + 1$.
- (*Recursion Condition*). $\Phi(x) = p \cdot \Phi(x*) + (1-p) \cdot \Phi(x\circ)$.
- (*Termination Condition*). For $x \in \{*, \circ\}^n$, $\Phi(x)$ is an integer. Moreover, $\Phi(x)$ is at most the sum of (1) the number of $*$'s in x and (2) n times the number of c-NFs F_i that violates the required properties if we choose $T = T_x = \{i \in [n] \mid x_i = \circ\}$. In particular, if $\Phi(x) \leq (1-p)n$, $T = T_x$ is a desired subset.

It remains to construct $x \in \{*, \circ\}^n$ such that $\Phi(x) \leq (1-p)n$. Indeed, it can be obtained by the greedy algorithm that, starting from $x \leftarrow \varepsilon$, appends either $*$ or $\circ$ to x to minimize $\Phi(x)$. By induction on i, we can prove that the string $x \in \{*, \circ\}^{\leq i}$ after the i-th round of the algorithm satisfies that $\Phi(x) \leq \Phi(\varepsilon) < (1-p)n + 1$. This is available in PV_1 as the property can be verified by a straightforward polynomial-time algorithm. Finally, the string $x \in \{*, \circ\}^n$ obtained after n rounds satisfies that $\Phi(x) \leq (1-p)n$, as it must be an integer smaller than $(1-p)n + 1$. This completes the proof. □

4.4.2 Average-Case Lower Bound in APX_1

We say that a partial assignment ρ *trivializes* an NF F if F is a constant function after applying ρ. Note that for every NF F that contains a non-constant clause C, there is an assignment to variables in C that trivializes F.

Lemma 4.9 (Random Restriction Lemma). *For every $k, c, b \in \mathbb{N}$, there exists an $n_0 \geq 1$ such that the following is provable in* APX_1*. Let $n, t, \ell \in$* Log*, $n > n_0$. Let $F_1, \ldots, F_\ell$ be c-NFs over n input variables, and $T \subseteq [n]$ be a subset of size $n - t$ such that for every $i \in [\ell]$, at least one of the following conditions hold.*

- (Narrow). *If we simultaneously fix all variables whose indices are in T, the c-NF F_i will be fed by at most b literals.*
- (Wide). *There are $m'_i \geq k \ln n$ (explicitly given) non-empty disjoint clauses $C'_{i,1}, \ldots, C'_{i,m'_i}$ such that (1) each $C'_{i,j}$ is a sub-clause of a different clause in F_i; (2) $S_{C'_{i,j}} \subseteq T$ for every $j \in [m'_i]$.*

Let Y be the random variable over $\{0,1\}$ that takes a seed x of length n, parses it as an assignment ρ to variables in T (i.e. it fixes the i-th variable to x_i for every $i \in T$), and outputs 1 if and only if at least one of $F_1, \ldots, F_\ell$ is neither trivialized nor depends on at most b literals after applying ρ. Then for every $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$

$$\mathbb{E}_\delta[Y] \leq \ell \cdot (1 - 2^{-c} + \beta)^{k \ln n} + \delta + \beta.$$

Proof. We argue in APX_1. Fix $k, c, b \in \mathbb{N}$ and let $n_0 \geq 1$ be a constant to be determined. Fix $n, t, \ell \in \mathsf{Log}$, $F_1, \ldots, F_\ell$, and $T \subseteq [n]$. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later.

For simplicity, we assume that $F_1, \ldots, F_{\ell'}$ are the gates that satisfy the second condition, and let $m'_i \geq k \cdot \ln n$ be the number of disjoint clauses $C'_{i,1}, \ldots, C'_{i,m'_i}$. We will define random variables over $\{0,1\}$ with n bit seeds as follows.

- For every $i \in [\ell']$ and $j \in [m'_i]$, X_{ij} is defined as the following random variable: Let $x \in \{0,1\}^n$ be the seed. Then $X_{ij} = 1$ if and only if fixing the r-th variable to x_r for every $r \in T$ does not trivialize $C'_{i,j}$.
- For every $i \in [\ell']$, let $X_i = \prod_{j=1}^{m'_i} X_{ij}$.
- Notice that $Y \leq \bigvee_{i \in [\ell']} X_i$.

Note that each X_{ij} reads at most $c \in \mathbb{N}$ bits of its seed. Thus by the *Brute Force Counting Lemma*, we know that $\mathbb{E}_\eta[X_{ij}] \leq 1 - 2^{-c} + 2\eta$. By the *Multiplication Principle*, we have

$$\mathbb{E}_\eta[X_{i1} X_{i2} \ldots X_{im'_i}] \leq (1 - 2^{-c} + 2\eta)^{m'_i} + 8\eta \cdot m'_i \leq (1 - 2^{-c} + 2\eta)^{k \cdot \ln n} + 8\eta \cdot |F_i| \leq (1 - 2^{-c} + \beta)^{k \ln n} + 8\eta \cdot |F_i|,$$

where the last inequality holds when $\eta \leq \beta/2$. By the *Union Bound*,

$$\mathbb{E}_\eta[Y] \leq \sum_{i=1}^{\ell'} \mathbb{E}_\eta[X_i] + 3\eta \cdot \ell' \leq \ell \cdot (1 - 2^{-c} + \beta)^{k \ln n} + 8\eta \cdot \ell \cdot (2^c n^c) + 3\eta \cdot \ell.$$

The lemma then follows from the *Precision Consistency of Expectation* by setting $\eta \triangleq \beta/(20 \cdot \ell \cdot 2^c n^c)$. □

Now we are ready to prove the average-case lower bound for $\oplus_n$ against AC^0.

Theorem 1.6 (Average-Case AC^0 Lower Bound for $\oplus_n$ in APX_1)**.** *For all constants $k, d \geq 1$, there exists a constant $n_0 \geq 1$ such that APX_1 proves the following statement. Let $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $n > n_0$, and $C : \{0,1\}^n \to \{0,1\}$ be an AC^0_d circuit of size at most n^k. Let $T_C : \{0,1\}^n \to \{0,1\}$ be the circuit that, given $x \in \{0,1\}^n$, outputs 1 if and only if $C(x) = \oplus_n(x)$. Then*

$$\mathsf{P}_\delta(T_C) \leq \frac{1}{2} + \frac{1}{n^k} + \delta + \beta. \tag{1.1}$$

Proof. We prove by induction on d in the meta-theory that the statement holds for every k. The constant $n_0^{k,d} \geq 1$ will be determined later in the proof. In both the base case and induction case, we argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and the circuit $C : \{0,1\}^n \to \{0,1\}$. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later.

Base Case. Suppose that $d = 1$. Towards a contradiction, assume that Equation (1.1) does not hold. Then by the Precision Consistency Axiom, we have

$$\mathsf{P}_\eta(T_C) > \frac{1}{2} + \frac{1}{n^k} + (\beta - 2\eta). \tag{4.18}$$

Note that $\mathsf{Fix}_b(T_C)$ is the circuit $T_{\mathsf{Fix}_b(C)\oplus b}$.

We first show that for $D : \{0,1\}^n \to \{0,1\}$, if D depends on at most $n-1$ of its input bits, then $\mathsf{P}_\eta(T_D) \le 1/2 + O(\eta)$. Suppose that D does not depend on the i-th input bit. Let X_D be the indicator random variable of T_D and ρ be any assignment to all but the i-th input bit of D. By the *Brute Force Counting Lemma*, we have that $\mathbb{E}_\eta[X_D|_\rho] \le 1/2 + 2\eta$. Subsequently, by the *Averaging Argument for Expectation*, we can conclude that

$$\mathbb{E}_\eta[X_D] \le \frac{1}{2} + 2\eta + 3\eta \le \frac{1}{2} + 5\eta. \tag{4.19}$$

This implies that $\mathsf{P}_\eta(T_D) \le 1/2 + 8\eta$ by Proposition 3.12.

Now we assume that C depends on all of its input bits and is of depth at most $d = 1$. Consider the following iterative P-oracle algorithm. Let $C_0 \triangleq C$ and $s_0 \triangleq 0$. The algorithm maintains the invariant that after the i-th round, C_i depends on all of its input bits. Given that the invariant holds after the $(i-1)$-th round, there exists $b_i \in \{0,1\}$ such that

- $\mathsf{Fix}_{b_i}(C_{i-1})$ is a constant circuit;
- $\mathsf{Fix}_{1-b_i}(C_{i-1})$ depends on all of its input bits.

Fix that $b_i \in \{0,1\}$. The algorithm then defines $C_i \triangleq \mathsf{Fix}_{1-b_i}(C_{i-1}) \oplus (1-b_i)$ and $s_i \triangleq s_{i-1} \oplus (1-b_i)$ in the i-th round, and the invariant is maintained.

We will prove by induction on $i \le n$ that

$$\mathsf{P}_\eta(T_{C_i \oplus s_i}) > \frac{1}{2} + \frac{2^i}{n^k} + (\beta - 8(i+1)\cdot\eta).$$

(Note that $n \in \mathsf{Log}$ and $i \le n$, so the induction hypothesis can be expressed by an open formula in APX_1.) The base case follows from Equation (4.18). Suppose that the inequality holds for $i < n$. Notice that

$$\begin{aligned}
\mathsf{P}_\eta(T_{C_i\oplus s_i}) &\le \frac{\mathsf{P}_\eta(\mathsf{Fix}_{b_i}(T_{C_i}) \oplus s_i) + \mathsf{P}_\eta(\mathsf{Fix}_{1-b_i}(T_{C_i}) \oplus s_i)}{2} && \text{(Precision Consistency Axiom)} \\
&= \frac{\mathsf{P}_\eta(T_{\mathsf{Fix}_{b_i}(C_i)\oplus s_i \oplus b_i}) + \mathsf{P}_\eta(T_{\mathsf{Fix}_{1-b_i}(C_i)\oplus s_i \oplus (1-b_i)})}{2} \\
&= \frac{\mathsf{P}_\eta(T_{\mathsf{Fix}_{b_i}(C_i)\oplus s_i \oplus b_i}) + \mathsf{P}_\eta(T_{C_{i+1}\oplus s_{i+1}})}{2} \\
&\le \frac{1}{2}\left(\frac{1}{2} + 8\eta + \mathsf{P}_\eta(T_{C_{i+1}\oplus s_{i+1}})\right).
\end{aligned}$$

(The last inequality follows as $\mathsf{Fix}_{b_i}(C_i)$ is a constant circuit and thus does not depend on all of its input bits.) It then follows by the induction hypothesis that

$$\mathsf{P}_\eta(T_{C_{i+1}\oplus s_{i+1}}) > \frac{1}{2} + \frac{2^{i+1}}{n^k} + (\beta - 8(i+2)\cdot\eta).$$

We set $n_0 \in \mathbb{N}$ to be sufficiently large such that $2^{n-10}/n^k \ge 1$ for every $n > n_0$. Therefore, we have that $\mathsf{P}_\eta(T_{C_{n-10}}) > 3/2 + (\beta - 8(n+1)\cdot\eta)$, where C_{n-10} has input length exactly 10 and is an AC^0 circuit of depth 1. This is provably impossible if we set $\eta \triangleq \beta/(20n)$ by the *Brute Force Counting Lemma*.

Induction Case. Suppose that the theorem holds for $d \in \mathbb{N}$. Our goal is to prove the theorem for $d+1$. Let $n_0^{k,d}$ be the constant $n_0 \ge 1$ corresponding to the theorem for d and k. Fix any $k \ge 1$ and let $n_0^{k,d+1}$ be a constant to be determined. Towards a contradiction, assume that Equation (1.1) does not hold. Then by the Precision Consistency Axiom, we have

$$\mathsf{P}_\eta(T_C) > \frac{1}{2} + \frac{1}{n^k} + (\beta - 2\eta). \tag{4.20}$$

At a high level, we will apply random restrictions twice to convert C to an AC^0_d circuit that computes the parity function w.h.p. on a smaller input length; after that, we can apply the induction hypothesis to conclude the proof.

Restriction 1. Let $G_1, G_2, \ldots, G_\ell$ be the gates in the first layer, i.e., directly fed by literals. We may view them as 1-NFs, as each literal can be viewed as a clause with one literal. Let $t \triangleq \sqrt{n}$. By the *Subset Selection Lemma*, there exists a subset $T_1 \subseteq [n]$ of size at most $n - t$ such that for every gate G_i, one of the conditions hold.

- If we fix all variables in T_1, G_i will be fed by at most b_1 variables, where $b_1 \in \mathbb{N}$ is a constant.
- At least $100k \cdot \ln n$ literals of G_i are using variables in T_1.

Fix the subset $T_1 \subseteq [n]$. We assume that $|T_1| = n - t$; if not, we add $n - t - |T_1|$ arbitrary elements to it.

We define random variables over $\{0,1\}$ with n bit seeds as follows.

- Let Y be the random variable in the *Random Restriction Lemma*. That is, given $x \in \{0,1\}^n$, it parses x as a partial assignment ρ to variables in T_1, and outputs 1 if and only if at least one of the gates is neither trivialized nor depends on at most b literals after applying ρ.
- Let Y_T be the indicator random variable of T_C.

By the *Random Restriction Lemma* and using $c = 1$, we have that

$$\mathbb{E}_\eta[Y] \le \ell \cdot (1/2 + \eta)^{100k \ln n} + 2\eta \le n^{-8k},$$

where the last inequality holds when η is sufficiently small and n is sufficiently large (by setting $n_0^{k,d+1} \in \mathbb{N}$). By the *Averaging Argument for Expectation*, there exists an assignment ρ_1 to variables in T_1 such that

$$\mathbb{E}_\eta[Y_T|_{\rho_1}] - \mathbb{E}_\eta[Y|_{\rho_1}] \ge \frac{1}{2} + \frac{1}{n^k} + (\beta - 30\eta) - \frac{1}{n^{8k}} > 5\eta, \tag{4.21}$$

where the last inequality holds if η is sufficiently small.

Fix the assignment ρ_1. Note that $Y|_{\rho_1} \in \{0,1\}$ as Y only reads its input variables in T_1. Therefore, we must have $Y|_{\rho_1} = 0$. In this case, all gates are either trivialized or fed by at most b_1 variables after applying ρ_1, and thus can be replaced by a gate of fan-in at most b_1.

Let $\sigma_1 \triangleq \oplus_{n-t}(\rho_1)$, $n_1 \triangleq t$, and $C_1 : \{0,1\}^{n_1} \to \{0,1\}$ be the circuit obtained from C by applying the assignment ρ_1, replacing each gate in the first layer with an equivalent gate of fan-in at most b_1, and XORing the output of the circuit with the bit σ_1. Note that $C|_{\rho_1}(x) = C_1(x) \oplus \sigma_1$ for every $x \in \{0,1\}^{n_1}$. Moreover, C_1 is of size at most $n^k \le n_1^{2k}$.

Let $T_{C_1} : \{0,1\}^{n_1} \to \{0,1\}$ be the circuit that, given x, it outputs 1 if and only if $C_1(x) = \oplus_{n_1}(x)$. It turns out that $Y_T|_{\rho_1}$ is the indicator random variable of T_{C_1}; to see this, notice that for every assignment x to all variables but T_1, $C(x \cup \rho_1) = \oplus_n(x \cup \rho_1)$ if and only if $C|_{\rho_1}(x) = (\oplus_{n_1}(x)) \oplus \sigma_1$, where $C|_{\rho_1}(x) \oplus \sigma_1 = C_1(x)$. Therefore, by Proposition 3.12, we have that

$$\mathsf{P}_\eta(T_{C_1}) \ge \mathbb{E}_\eta[Y_T|_\rho] - 3\eta \ge \frac{1}{2} + \frac{1}{n^k} + (\beta - 33\eta) - \frac{1}{n^{8k}} \ge \frac{1}{2} + \frac{1}{n_1^{4k}} + (\beta - 33\eta),$$

where the last inequality holds if n is sufficiently large (by setting $n_0^{k,d+1} \in \mathbb{N}$).

Restriction 2. As mentioned above, each gate in the first layer of C_1 has fan-in at most b_1, and thus the gates in the second layer of C_1 computes b_1-NFs. Let $F_1, F_2, \ldots, F_{\ell_1}$ be the b_1-NFs in the second layer of C_1. Let $t_1 \triangleq \sqrt{n_1}$. By the *Subset Selection Lemma* with appropriate choice of parameters, there exists a subset $T_2 \subseteq [n]$ of size at most $n_1 - t_1$ such that for every $i \in [\ell_1]$, one of the conditions hold.

- If we fix all variables in T_2, F_i will depend on at most b_2 variables, where $b_2 \in \mathbb{N}$ is a constant.
- There are $m_1' \ge 100k \cdot 4^{b_1} \cdot \ln n$ disjoint sub-clauses of F_i that only use literals from variables in T_2.

Let $n_2 \triangleq t_1$ and fix the set $T_2 \subseteq [n]$. We assume that $|T_2| = n_1 - t_1$; if not, we add $n_1 - t_1 - |T_2|$ arbitrary elements to it. We define random variables over $\{0,1\}$ with n bit seeds as follows.

- Let Y' be the random variable in the *Random Restriction Lemma*. That is, given $x \in \{0,1\}^n$, it parses x as a partial assignment ρ to variables in T_2, and outputs 1 if and only if each of the b-NFs is either trivialized or depends on at most b_2 literals after applying ρ.

- Let Y'_T be the indicator random variable of T_{C_1}.

By the *Random Restriction Lemma*, we have that

$$\mathbb{E}_\eta[Y'] \leq \ell_1 \cdot (1 - 2^{-b_1} + \eta)^{100k \cdot 4^{b_1} \cdot \ln n} + 2\eta \leq n_1^{-8k},$$

where the last inequality holds when η is sufficiently small and n is sufficiently large (by setting $n_0^{k,d+1} \in \mathbb{N}$). By the *Averaging Argument for Expectation*, there exists an assignment ρ_2 to variables in T_2 such that

$$\mathbb{E}_\eta[Y'_T|_{\rho_2}] - \mathbb{E}_\eta[Y'|_{\rho_2}] \geq \frac{1}{2} + \frac{1}{n_1^{4k}} + (\beta - 36\eta) - \frac{1}{n_1^{8k}} > 5\eta, \tag{4.22}$$

where the last inequality holds if η is sufficiently small.

Fix the assignment ρ_2. Note that $Y'|_{\rho_2} \in \{0,1\}$ as it only reads its input variables in T_2. Therefore, we must have $Y'_{\rho_2} = 0$. In such case, all b_1-NFs (i.e. gates in the second layer of C_1) are either trivialized or fed by at most b_2 variables. In such case, we can transform C_1 into an equivalent circuit of depth at most d as follows. Suppose that $d \geq 2$ (the case for $d = 1$ is left as an exercise). For each gate G in the second layer, if it is not trivialized, we remove G and consider each gate G' in the third layer originally fed by G:

- If G' is an AND gate, we rewrite G as an equivalent CNF of size at most $b_2 \cdot 2^{b_2}$ and connect all clauses of it to G'.
- If G' is an OR gate, we rewrite G as an equivalent DNF of size at most $b_2 \cdot 2^{b_2}$ and connect all clauses of it to G'.

In either case, the circuit remains functionally equivalent.

Let $\sigma_2 \triangleq \oplus_{n_1 - t_1}(\rho_2)$ and $C_2 : \{0,1\}^{n_2} \to \{0,1\}$ be the depth-d circuit that computes $C_1|_{\rho_2}(x) \oplus \sigma_2$. The size of C_2 blows up by a linear factor, which is at most $O(n^k) \leq n_2^{6k}$, when n is sufficiently large (by setting $n_0^{k,d+1} \in \mathbb{N}$). Let $T_{C_2} : \{0,1\}^{n_2} \to \{0,1\}$ be the circuit that, given x, it outputs 1 if and only if $C_2(x) = \oplus_{n_2}(x_2)$. As before, $Y'_T|_\rho$ is the indicator random variable of T_{C_2}. Therefore, we have

$$\begin{aligned}
\mathsf{P}_\eta(T_{C_2}) &\geq \mathbb{E}_\eta[Y'_T|_\rho] - 3\eta && \text{(Proposition 3.12)}\\
&\geq \frac{1}{2} + \frac{1}{n_1^{4k}} + (\beta - 39\eta) - \frac{1}{n_1^{8k}} && \text{(Equation (4.22))}\\
&\geq \frac{1}{2} + \frac{1}{n_2^{6k}} + (\beta - 39\eta)\\
&> \frac{1}{2} + \frac{1}{n_2^{6k}} + 2\eta, && (4.23)
\end{aligned}$$

where the last two lines hold when n is sufficiently large (by setting $n_0^{k,d+1} \in \mathbb{N}$) and η is sufficiently small.

Now we arrive at a contradiction: $C_2 : \{0,1\}^{n_2} \to \{0,1\}$ is a depth-d circuit of size at most n_2^{6k}, and it computes parity with advantage $1/n_2^{6k}$. This violates Equation (1.1). The theorem then follows from the induction hypothesis for depth d and size n_2^{6k} if we set $\eta^{-1} \in \mathsf{Log}$ and $n_0^{k,d+1} \in \mathbb{N}$ appropriately based on $b_1, b_2, n_0^{6k,d}$ and the requirements of inequalities used in the proofs. □

4.4.3 Worst-Case Lower Bound in PV_1

First, we derandomize the *Random Restriction Lemma* via an explicit implementation of the method of conditional expectations in PV_1.

Lemma 4.10 (Derandomized Restriction Lemma). *For every $k, c, b \in \mathbb{N}$, there exists an $n_0 \geq 1$ such that the following is provable in PV_1. Let $n, t, \ell \in \mathsf{Log}$, $n > n_0$. Let $F_1, \ldots, F_\ell$ be c-NFs over n input variables, and $T \subseteq [n]$ be a subset of size $n - t$ such that for every $i \in [\ell]$, at least one of the following conditions hold.*

- (Narrow). *If we fix the j-th variable for every $j \in T$, F_i is fed by at most b literals.*

- (Wide). *There are $m'_i \geq k \ln n$ (explicitly given) disjoint clauses $C'_{i,1}, \ldots, C'_{i,m'_i}$ such that (1) each $C'_{i,j}$ is a sub-clause of a different clause in F_i; (2) $S_{C'_{i,j}} \subseteq T$ for every $j \in [m'_i]$.*

Suppose that $\ell \cdot (1 - 2^{-c})^{k \ln n} < 1$. Then there exists an assignment ρ to the variables in T such that each of $F_1, \ldots, F_\ell$ is either trivialized or depends on at most b literals after applying ρ.

Proof. We argue in PV_1. Let $k, c, b \in \mathbb{N}$ and $n_0 \geq 1$ be a constant to be determined later. Fix $n, t, \ell \in \mathsf{Log}$, c-NFs $F_1, \ldots, F_\ell$, and $T \subseteq [n]$. We say that a c-NF is good after applying a restriction ρ to the variables in T if it is either trivialized or depends on at most b literals. As the c-NFs satisfying the first bullet are good regardless of the assignment ρ, we assume without loss of generality that all c-NFs satisfy the second bullet. We will construct an assignment ρ such that all such c-NFs are trivialized.

For simplicity of presentation, we assume that $T = \{1, 2, \ldots, n-t\}$. Fix any $i \in [\ell]$, and let $C'_{i,1}, \ldots, C'_{i,m'_i}$ be the disjoint sub-clauses such that $S_{C'_{i,j}} \subseteq T$. For a partial assignment $x \in \{0,1\}^{\leq n-t}$ to the first $|x|$ variables, we say that:

- $C'_{i,j}$ is positively determined if it is an AND gate and all literals of it are fixed to 1, or it is an OR gate and all literals of it are fixed to 0.
- $C'_{i,j}$ is negatively determined if it is an AND gate and one of its literals is fixed to 0, or it is an OR gate and one of its literals is fixed to 1.
- $C'_{i,j}$ is d-far from positively determined if it is not negatively determined, and there are exactly d of its literals that remain unfixed.

We define $\phi_{ij}, \Phi_i, \Phi : \{0,1\}^{\leq n-t} \to \{0,1\}$ as follows. Given any $x \in \{0,1\}^{\leq n-t}$ parsed as a partial assignment to the first $|x|$ literals,

$$\Phi(x) \triangleq \sum_{i=1}^{\ell} \Phi_i(x), \quad \Phi_i(x) \triangleq \prod_{j=1}^{m'_i} (1 - \phi_{ij}(x)) \tag{4.24}$$

$$\phi_{ij}(x) \triangleq \begin{cases} 0 & C'_{ij} \text{ is negatively determined} \\ 2^{-d} & C'_{ij} \text{ is } d\text{-far from positively determined} \end{cases} \tag{4.25}$$

For instructive purposes, we mention that the combinatorial interpretation of $\Phi(x)$ is the expected number of c-NFs that are not trivialized if we extend x to an assignment to variables in T by fixing each unfixed bit uniformly at random. Note that this is not a part of the PV_1 proof. Instead, we prove that:

- (*Initial Condition*). Note that $\phi_{ij}(\varepsilon) \geq 2^{-c}$ and thus

$$\Phi_i(\varepsilon) \leq (1 - 2^{-c})^{m'_i} \leq (1 - 2^{-c})^{k \ln n} < \frac{1}{\ell}, \quad \Phi(\varepsilon) < 1.$$

- (*Recursion Condition*). For every $x \in \{0,1\}^{< n-t}$, we can prove that $\Phi_i(x) = (\Phi_i(x0) + \Phi_i(x1))/2$. To see this, notice that:
  - When the $(|x|+1)$-th variable does not appear in $C_{i,1}, \ldots, C_{i,m'_i}$, $\Phi_i(x) = \Phi(x0) = \Phi(x1)$.
  - Otherwise, it appears in exactly one of $C_{i,1}, \ldots, C_{i,m'_i}$ as the clauses are disjoint. Assume for simplicity that it appears in $C_{i,1}$ and it is an OR gate. Then

$$\Phi_i(x) = (1 - \phi_{i1}(x)) \cdot \prod_{j=2}^{m'_i} (1 - \phi_{ij}(x)),$$

$$\Phi_i(x0) = (1 - 2 \cdot \phi_{i1}(x)) \prod_{j=2}^{m'_i} (1 - \phi_{ij}(x)), \quad \Phi_i(x1) = \prod_{j=2}^{m'_i} (1 - \phi_{ij}(x)).$$

  Thus $\Phi_i(x) = (\Phi_i(x0) + \Phi_i(x1))/2$. Subsequently, $\Phi(x) = (\Phi(x0) + \Phi(x1))/2$.

- (*Termination Condition*). $\Phi_i(x) \in \{0,1\}$ for $x \in \{0,1\}^{n-t}$. Moreover, if $\Phi(x) = 0$, the partial assignment x will trivialize all c-NFs.

The lemma then follows from a greedy algorithm as in the *Subset Selection Lemma*. $\square$

Theorem 1.7 (Worst-Case AC^0 Lower Bound for $\oplus_n$ in PV_1)**.** *For all constants $k, d \geq 1$, there exists a constant $n_0 \geq 1$ such that PV_1 proves the following statement. For every $n \in \mathsf{Log}$, $n > n_0$, and AC^0_d circuit $C : \{0,1\}^n \to \{0,1\}$ of size at most n^k, there exists a string $x \in \{0,1\}^n$ such that $C(x) \neq \oplus_n(x)$.*

Proof Sketch. The proof closely follows the proof of *Average-Case AC^0 Lower Bound for $\oplus_n$*, so we will only sketch the argument. We prove it by induction on d in the meta-theory, and the constant $n_0 = n_0^{k,d}$ depends on both k and d. The case when $d = 1$ is easy and left as an exercise.

For $d \geq 2$, we assume towards a contradiction that C computes $\oplus_n$. We first apply the *Subset Selection Lemma* to find a subset T of size $n - \sqrt{n}$ by viewing the gates in the first layer as 1-NFs, and then apply the *Derandomized Restriction Lemma* to find an assignment ρ to variables in T such that the gates in the first layer are either trivialized or of fan-in at most $b = O(1)$ after applying ρ. Let $n_1 = \sqrt{n}$. We can construct (from C and ρ) a circuit C_1 that computes $\oplus_{n_1}$ on the unfixed bits such that all gates in the first layer are of fan-in b.

We then apply the *Subset Selection Lemma* again to find a subset T_1 of size $n_1 - \sqrt{n_1}$ by viewing the gates in the second layer as b-NFs, and then apply the *Derandomized Restriction Lemma* to find an assignment ρ_1 to variables in T_1 such that the gates in the second layer are either trivialized or of fan-in at most $b_1 = O(1)$ after applying ρ_1. Let $n_2 = \sqrt{n}_1$. We can then construct (from C_1 and ρ_1) a circuit C_2 of depth at most $d - 1$ that computes $\oplus_{n_2}$ on the unfixed bits. The size of the circuit is at most $n^{k+1} \leq n_2^{6k}$. This leads to a contradiction to the induction hypothesis by setting $n_0^{k,d}$ to be sufficiently large based on $n_0^{6k,d-1}$. $\square$

4.5 Blum-Luby-Rubinfeld Linearity Testing

We now formalize the linearity testing algorithm due to Blum, Luby, and Rubinfeld [BLR93]. Recall that a function $g : \{0,1\}^n \to \{0,1\}$ is said to be linear if $g(x \oplus y) = g(x) \oplus g(y)$, where $\oplus$ denotes bit-wise XOR; equivalently, $g(x) = \langle x, z \rangle \bmod 2$ for some $z \in \{0,1\}^n$. Let $g : \{0,1\}^n \to \{0,1\}$ be a function. Blum, Luby, and Rubinfeld [BLR93] proved that for any sufficiently small constant $\varepsilon > 0$:

- (*Linearity Testing*): If g is ε-far from any linear function, then the BLR linearity testing algorithm fails with probability at least $\Omega(\varepsilon)$. Conversely, if g is ε-close to a linear function, the BLR linearity testing algorithm fails with probability at most $O(\varepsilon)$.
- (*Self Correction*): The key idea behind linearity testing is a random self correctness algorithm: If g is ε-close to a linear function $\hat{g}$, then the function $f(x,r) \triangleq g(x \oplus r) \oplus g(r)$ is a *randomized* algorithm that computes $\hat{g}$ with error $O(\varepsilon)$, where x is the input and r is the random seed.

Linearity testing is the key component of the exponential length PCP theorem $\mathsf{NP} \subseteq \mathsf{PCP}[\mathsf{poly}, 1]$, which is further used to reduce the number of queries in the proof of the PCP theorem $\mathsf{NP} = \mathsf{PCP}[\log n, 1]$ (see, e.g., [Har04]).

We first state the main theorems, namely the completeness and soundness of the BLR linearity testing. The completeness states that a function that is close to a linear function is likely to be accepted. Formally:

Theorem 4.11 (Completeness of BLR linearity testing)**.** *APX_1 proves the following. Let $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C : \{0,1\}^n \to \{0,1\}$ be a circuit, and $z \in \{0,1\}^n$ be a string. Let $\varepsilon \in \mathbb{Q}$ such that $\varepsilon < 1/2$. Define the following circuits:*

- *Let $T_C(x)$ be the circuit that outputs 1 if and only if $C(x) \neq \langle x, z \rangle \bmod 2$.*
- *Let $T_{C,\mathsf{BLR}}(x,y) : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be the circuit that outputs 1 when $C(x) \oplus C(y) \neq C(x \oplus y)$.*

Suppose that $\mathsf{P}_\delta(T_C) \leq \varepsilon$. Then $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \leq 3\varepsilon + 4\delta + \beta$.

The soundness states that if a function is likely to be accepted by the BLR linearity testing algorithm, then it is close to a linear function.

Theorem 4.12 (Soundness of BLR linearity testing)**.** APX_1 *proves the following statement. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ *and* $C : \{0,1\}^n \to \{0,1\}$ *be a circuit. Let* $\varepsilon \in \mathbb{Q}$*. Assume that* $\varepsilon, \delta, \beta < 0.01$*. We define the following circuits:*

- *For every* $z \in \{0,1\}^n$*, let* $T_{C,z}(x)$ *be the circuit that outputs* 1 *if and only if* $C(x) \neq \langle x, z\rangle \bmod 2$*.*
- *Let* $T_{C,\mathsf{BLR}}(x, y)$ *be the circuit that outputs* 1 *if and only if* $C(x) \oplus C(y) \neq C(x \oplus y)$*.*

Suppose that $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \le \varepsilon$*. Then there exists a string* $z \in \{0,1\}^n$ *such that* $\mathsf{P}_\delta(T_{C,z}) \le 5\varepsilon + 6\delta + \beta$*.*

We formalize the combinatorial proof [BLR93] via majority correction (see [BCH+96] for an alternate proof). Note that the same proof is also formalized by Pich [Pic15a] in APC_1 to prove the exponential PCP theorem $\mathsf{NP} \subseteq \mathsf{PCP}[\mathsf{poly}, 1]$, and our main contribution is to show that it can be formalized in the (possibly weaker) theory APX_1.[22]

4.5.1 Two Useful Lemmas

Before formalizing the BLR linearity testing algorithm, we prove two useful lemmas. The first lemma shows that the acceptance probability of a circuit does not change significantly if the input is XORed with a *fixed* string. Formally:

Lemma 4.13 (Re-randomization)**.** APX_1 *proves the following statement. For every* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$*, circuit* $T : \{0,1\}^n \to \{0,1\}$*, and* $x \in \{0,1\}^n$*, let* $T_x^{\oplus} : \{0,1\}^n \to \{0,1\}$ *be the circuit defined as* $T_x^{\oplus}(r) \triangleq T(x \oplus r)$*. Then* $|\mathsf{P}_\delta(T) - \mathsf{P}_\delta(T_x^{\oplus})| \le 2\delta + \beta$*.*

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $x \in \{0,1\}^n$, the circuit T and $T_x^{\oplus}$. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Suppose, towards a contradiction, that $|\mathsf{P}_\delta(T) - \mathsf{P}_\delta(T_x^{\oplus})| > 2\delta + \beta$. Then by the Precision Consistency Axiom, we have that

$$|\mathsf{P}_\eta(T) - \mathsf{P}_\eta(T_x^{\oplus})| > \beta - 4\eta.$$

Recall that for a circuit C, $C^{k,z}$ denotes the circuit obtained by fixing the last k input bits of C to be z. We will design an n-round iterative $\mathsf{PV}(\mathsf{P})$ algorithm that, in the i-th round, outputs a string $z_i \in \{0,1\}^i$ such that

$$|\mathsf{P}_\eta(T^{i,z_i}) - \mathsf{P}_\eta(T_x^{\oplus,i,z_i'})| > \beta - 10 \cdot (i+1) \cdot \eta. \tag{4.26}$$

where $z_i' \triangleq z_i \oplus x_{>n-i}$. The algorithm initializes by setting $z_0 \triangleq \varepsilon$. In the i-th round, it works as follows:

- Recall that by the invariant that Equation (4.26) holds in the $(i-1)$-th round, we have

$$|\mathsf{P}_\eta(T^{i-1,z_{i-1}}) - \mathsf{P}_\eta(T_x^{\oplus,i-1,z_{i-1}'})| > \beta - 10 \cdot i \cdot \eta.$$

- By the Local Consistency Axiom, we know that

$$\begin{aligned} &|\mathsf{P}_\eta(T^{i-1,z_{i-1}}) - \mathsf{P}_\eta(T_x^{\oplus,i-1,z_{i-1}'})| \\ \le\ & \frac{1}{2}\sum_{b\in\{0,1\}} \left|\mathsf{P}_\eta(\mathsf{Fix}_b(T^{i-1,z_{i-1}})) - \mathsf{P}_\eta(\mathsf{Fix}_{b\oplus x_{n-i+1}}(T_x^{\oplus,i-1,z_{i-1}'}))\right| + 3\eta. \end{aligned}$$

  Subsequently, there is a constant $b \in \{0,1\}$ such that

$$\left|\mathsf{P}_\eta(\mathsf{Fix}_b(T^{i-1,z_{i-1}})) - \mathsf{P}_\eta(\mathsf{Fix}_{b\oplus x_{n-i+1}}(T_x^{\oplus,i-1,z_{i-1}'}))\right| > \beta - 10 \cdot i \cdot \eta - 3\eta. \tag{4.27}$$

  The algorithm finds such $b \in \{0,1\}$ by querying the P-oracle, and outputs $z_i \triangleq b \circ z_{i-1}$.

[22] Our formalization is slightly different: We formalize linear functions $x \mapsto \langle x, z\rangle \bmod 2$ by explicitly giving z, while Pich [Pic14] formalizes linear functions f using the sentence that for every x, y, $f(x \oplus y) = f(x) \oplus f(y)$; nevertheless, the difference in formalization does not matter in most cases.

To see that the algorithm is correct, notice that the circuit $\mathsf{Fix}_b(T^{i-1,z_{i-1}})$ is functionally equivalent to T^{i,z_i}, and $\mathsf{Fix}_{b\oplus x_{n-i+1}}(T_x^{\oplus i-1,z'_{i-1}})$ is functionally equivalent to $T_x^{\oplus,i,z_i}$. Therefore, by Equation (4.27) and the *Global Consistency*, we have

$$|\mathsf{P}_\eta(T^{i,z_i}) - \mathsf{P}_\eta(T_x^{\oplus,i,z'_i})| \ge \left|\mathsf{P}_\eta(\mathsf{Fix}_b(T^{i-1,z_{i-1}})) - \mathsf{P}_\eta(\mathsf{Fix}_{b\oplus x_{n-i+1}}(T_x^{\oplus,i-1,z'_{i-1}}))\right| - 6\eta \ge \beta - 10\cdot(i+1)\cdot\eta.$$

The correctness of the algorithm can thus be proved by induction on a $\mathsf{PV}(\mathsf{P})$ term, which is available by Theorem 2.10.

Finally, in the n-th round, the algorithm outputs a string $z_n \in \{0,1\}^n$ such that

$$|\mathsf{P}_\eta(T^{n,z_n}) - \mathsf{P}_\eta(T_x^{\oplus,n,z'_n})| > \beta - 10\cdot(n+1)\cdot\eta,$$

where $z'_n \triangleq z_n \oplus x$. Note that both circuits above have input length 0 and, by the definition, must output the same value. This violates the Boundary Axiom by setting $\eta \triangleq \beta/(100(n+1))$. $\square$

The second lemma is as follows. Let X_1 and X_2 be two explicitly i.i.d. RVs over $\{0,1\}$. If $\Pr[X_1 = X_2]$ is larger than $1/2$, then $\mathbb{E}[X_i]$ must be biased. Formally:

Lemma 4.14. APX_1 *proves the following statement. Let $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $V = \{0,1\}$, and X_1, X_2 be explicitly i.i.d. RVs over V defined by the circuit $C : \{0,1\}^n \to \{0,1\}$. Let Y_c be the indicator random variable of $X_1 = X_2$, where X_1 and X_2 takes disjoint random seeds. Then for $i \in \{1,2\}$,*

$$\left|\mathbb{E}_\delta[X_i] - \frac{1}{2}\right| \ge \sqrt{\frac{\mathbb{E}_\delta(Y_c)}{2} - \frac{1}{4} - 5\delta - \beta}.$$

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, circuits C, T, and random variables X_1, X_2, Y_c. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Note that as X_1 and X_2 are both the indicator random variable of C, we can prove by Proposition 3.12 that $|\mathbb{E}_\eta[X_1] - \mathsf{P}_\eta(C)|, |\mathbb{E}_\eta[X_2] - \mathsf{P}_\eta(C)| \le 3\eta$, and subsequently

$$|\mathbb{E}_\eta[X_1] - \mathbb{E}_\eta[X_2]| \le 6\eta. \tag{4.28}$$

Let Y_0, Y_1 be random variables over $\{0,1\}$ such that Y_i takes $(x_1, x_2) \in \{0,1\}^n \times \{0,1\}^n$ and output 1 if and only if $C(x_1) = C(x_2) = i$. It is easy to see that for every assignment $\rho = (x_1, x_2) \in \{0,1\}^{2n}$ to the seed, $Y_c|_\rho = Y_0|_\rho + Y_1|_\rho$. Therefore, by the *Averaging Argument for Expectation*,

$$|\mathbb{E}_\eta[Y_c] - \mathbb{E}_\eta[Y_0] - \mathbb{E}_\eta[Y_1]| \le 6\eta. \tag{4.29}$$

Let $\overline{X}_1, \overline{X}_2$ be the random variables defined by $1 - X_1$ and $1 - X_2$, respectively. Using *Complementary Counting*,

$$\left|\mathbb{E}_\eta[\overline{X}_i] + \mathbb{E}_\eta[X_i] - 1\right| \le 6\eta \tag{4.30}$$

for $i \in \{1,2\}$. We can further observe that for every assignment ρ to the random seed, $Y_0|_\rho = \overline{X}_1\overline{X}_2|_\rho$ and $Y_1|_\rho = X_1X_2|_\rho$, and subsequently by the *Averaging Argument for Expectation*,

$$\left|\mathbb{E}_\eta[Y_0] - \mathbb{E}_\eta[\overline{X}_1\overline{X}_2]\right| \le 6\eta, \quad |\mathbb{E}_\eta[Y_1] - \mathbb{E}_\eta[X_1X_2]| \le 6\eta.$$

Subsequently, by the *Multiplication Principle*,

$$\left|\mathbb{E}_\eta[Y_0] - \mathbb{E}_\eta[\overline{X}_1]\cdot\mathbb{E}_\eta[\overline{X}_2]\right| \le 14\eta, \quad |\mathbb{E}_\eta[Y_1] - \mathbb{E}_\eta[X_1]\cdot\mathbb{E}_\eta[X_2]| \le 14\eta. \tag{4.31}$$

Fix any $i \in \{1,2\}$ and let $p \triangleq \mathbb{E}_\delta[X_i]$, $q \triangleq \mathbb{E}_\delta[Y_C]$. For simplicity, we assume that $0 < \mathbb{E}_\eta[X_1] < 1$ and $0 < \mathbb{E}_\eta[X_2] < 1$. We can perform the following calculation:

$$\begin{aligned}
\mathbb{E}_\delta[Y_C] &\le \mathbb{E}_\eta[Y_C] + (\delta + 2\eta) && (\textit{Precision Consistency of Expectation})\\
&\le \mathbb{E}_\eta[Y_0] + \mathbb{E}_\eta[Y_1] + (\delta + 8\eta) && (\text{Equation (4.29)})\\
&\le \mathbb{E}_\eta[\overline{X}_1] \cdot \mathbb{E}_\eta[\overline{X}_2] + \mathbb{E}_\eta[X_1] \cdot \mathbb{E}_\eta[X_2] + (\delta + 36\eta) && (\text{Equation (4.31)})\\
&\le (1 - \mathbb{E}_\eta[X_1] + 6\eta)(1 - \mathbb{E}_\eta[X_2] + 6\eta) + \mathbb{E}_\eta[X_1] \cdot \mathbb{E}_\eta[X_2] + (\delta + 36\eta) && (\text{Equation (4.30)})\\
&\le (1 - \mathbb{E}_\eta[X_i] + 6\eta)(1 + \mathbb{E}_\eta[X_i] + 12\eta) + \mathbb{E}_\eta[X_i] \cdot (\mathbb{E}_\eta[X_i] - 6\eta) + (\delta + 36\eta) && (\text{Equation (4.28)})\\
&\le (1 - p + \delta + 14\eta)^2 + (p + \delta + 14\eta)^2 + (\delta + 36\eta) && (\textit{Precision Consistency of Expectation})\\
&\le (1-p)^2 + (\delta + 14\eta)^2 + 2(1-p)(\delta + 14\eta) + p^2 + (\delta + 14\eta)^2 + 2p(\delta + 14\eta) + (\delta + 36\eta)\\
&\le (1-p)^2 + p^2 + 5\delta + 78\eta\\
&\le 1 - 2(p - p^2) + 5\delta + \beta,
\end{aligned}$$

where the last inequality holds if we set $\eta \triangleq \beta/100$. Thus we have $p - p^2 \le (1-q)/2 + 5\delta + \beta$, and subsequently

$$\left|\frac{1}{2} - p\right| = \sqrt{\left(\frac{1}{2} - p\right)^2} = \sqrt{\frac{1}{4} - (p - p^2)} \ge \sqrt{\frac{1}{4} - \left(\frac{1-q}{2} + 5\delta + \beta\right)} \ge \sqrt{\frac{q}{2} - \frac{1}{4} - 5\delta - \beta}.$$

This completes the proof. □

4.5.2 Completeness of BLR Linearity Testing

We first formalize the completeness of the linearity testing algorithm. That is, if a circuit $C : \{0,1\}^n \to \{0,1\}$ computes a function that is indeed close to a linear function $x \mapsto \langle z, x\rangle \bmod 2$, then the self-correction algorithm works. Formally:

Lemma 4.15 (Completeness of BLR self-correction)**.** APX_1 *proves the following statement. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$*,* $C : \{0,1\}^n \to \{0,1\}$ *be a circuit, and* $z \in \{0,1\}^n$ *be a string. Let* $\varepsilon \in \mathbb{Q}$ *such that* $\varepsilon < 1/2$*. Define the following circuits:*

- *Let* $T_C(x)$ *be the circuit that outputs* 1 *if and only if* $C(x) \neq \langle x, z\rangle \bmod 2$*.*
- *Let* $D(x,r) : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *be the circuit that outputs* $C(x \oplus r) \oplus C(r)$*.*
- *For* $x \in \{0,1\}^n$*, let* $T_{D,x}(r)$ *be the circuit that outputs* 1 *if and only if* $D(x,r) \neq \langle x, z\rangle \bmod 2$*.*

Suppose that $\mathsf{P}_\delta(T_C) \le \varepsilon$*. Then for every* $x \in \{0,1\}^n$*,* $\mathsf{P}_\delta(T_{D,x}) \le 2(\delta + \varepsilon) + \beta$*.*

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, the circuit C, $z \in \{0,1\}^n$, and $\varepsilon \in \mathbb{Q}$. Let $T_C, D, T_{D,x}$ be the circuit as defined above, and $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Suppose that $\mathsf{P}_\delta(T_C) \le \varepsilon$, we know by the Precision Consistency Axiom that

$$\mathsf{P}_\eta(T_C) \le \varepsilon + \delta + 2\eta. \tag{4.32}$$

Fix any $x \in \{0,1\}^n$. Let X_1, X_2 be random variables over $\{0,1\}$ that takes a seed $r \in \{0,1\}^n$, where $X_1 = 1$ if $C(r) \neq \langle r, z\rangle \bmod 2$, and $X_2 = 1$ if $C(x \oplus r) \neq \langle x \oplus r, z\rangle \bmod 2$. It is clear that X_1 is the indicator random variable of T_C, and thus by Equation (4.32) and Proposition 3.12, $\mathbb{E}_\eta[X_1] \le \varepsilon + \delta + 5\eta$. By Proposition 3.12 and the *Re-randomization Lemma*, we can further show that

$$\mathbb{E}_\eta[X_2] \le \mathsf{P}_\eta(T_C) + 6\eta \le \varepsilon + \delta + 8\eta.$$

Let $T_\vee(r)$ be the circuit that outputs 1 if and only if $C(r) \neq \langle r, z\rangle \bmod 2$ or $C(x \oplus r) \neq \langle x \oplus r, z\rangle \bmod 2$, and Y be the indicator random variable of $T_\vee$. By Proposition 3.12 and the *Union Bound*, we have

$$\mathsf{P}_\eta(T_\vee) \le \mathbb{E}_\eta[Y] + 3\eta \le \mathbb{E}_\eta[X_1] + \mathbb{E}_\eta[X_2] + 6\eta \le 2\varepsilon + 2\delta + 19\eta. \tag{4.33}$$

Finally, we observe that if $T_{D,x}(r) = 1$, then $T_\vee(r) = 1$. To see this, assume that $T_\vee(r) = 0$, we have

$$D(x,r) = C(x \oplus r) \oplus C(r) = \langle x \oplus r, z\rangle + \langle r, z\rangle \bmod 2 = \langle x, z\rangle \bmod 2,$$

which implies that $T_{D,x}(r) = 0$. Therefore, we have that

$$\begin{aligned} \mathsf{P}_\delta(T_{D,x}) &\le \mathsf{P}_\eta(T_{D,x}) + \delta + 2\eta && (\text{Precision Consistency Axiom}) \\ &\le \mathsf{P}_\eta(T_\vee) + \delta + 5\eta && (\textit{Monotonicity of Approximate Counting}) \\ &\le 2\varepsilon + 2\delta + 24\eta. && (\text{Equation (4.33)}) \end{aligned}$$

This completes the proof by setting $\eta \triangleq \beta/30$. ☐

It can be observed that this immediately gives the completeness of the BLR identity testing algorithm. Namely, if C is close to a linear function, then it passes the linearity testing with high probability.

Theorem 4.11 (Completeness of BLR linearity testing)**.** APX_1 *proves the following. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C : \{0,1\}^n \to \{0,1\}$ *be a circuit, and* $z \in \{0,1\}^n$ *be a string. Let* $\varepsilon \in \mathbb{Q}$ *such that* $\varepsilon < 1/2$. *Define the following circuits:*

- *Let* $T_C(x)$ *be the circuit that outputs* 1 *if and only if* $C(x) \neq \langle x, z\rangle \bmod 2$.
- *Let* $T_{C,\mathsf{BLR}}(x,y) : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *be the circuit that outputs* 1 *when* $C(x) \oplus C(y) \neq C(x \oplus y)$.

Suppose that $\mathsf{P}_\delta(T_C) \le \varepsilon$. *Then* $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \le 3\varepsilon + 4\delta + \beta$.

Proof. We argue in APX. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $C : \{0,1\}^n \to \{0,1\}$, $z \in \{0,1\}^n$, $\varepsilon \in \mathbb{Q}$, and T_C, $T_{C,\mathsf{BLR}}$ be the circuits as described above. Let $T'_{C,\mathsf{BLR}} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be the circuit that given (x,y), outputs 1 when $C(x \oplus y) \oplus C(y) \neq \langle x, z\rangle \bmod 2$. Let $I_C, I_{C,\mathsf{BLR}}, I'_{C,\mathsf{BLR}}$ be the indicator random variables of $T_C, T_{C,\mathsf{BLR}}, T'_{C,\mathsf{BLR}}$, respectively.

Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Note that by the *Completeness of BLR Self Correction* and Proposition 3.12, we can prove that for any assignment ρ to be first part of the seed of $T'_{C,\mathsf{BLR}}$, we have $\mathbb{E}_\eta[I_{C,\mathsf{BLR}}|_\rho] \le 2(\delta + \varepsilon) + 4\eta$. Subsequently, by the *Averaging Argument for Expectation*, we have

$$\mathbb{E}_\eta[I_{C,\mathsf{BLR}}] \le 2(\delta + \varepsilon) + 4\eta + 6\eta \le 2(\delta + \varepsilon) + 10\eta. \tag{4.34}$$

By Proposition 3.12 and Precision Consistency Axiom, we also have $\mathbb{E}_\eta[I_C] \le \varepsilon + \delta + 5\eta$.

It can be observed that if $T_{C,\mathsf{BLR}}(x,y) = 1$, then either $T_C(x,y) = 1$ or $T'_{C,\mathsf{BLR}}(x,y) = 1$. Therefore, by the *Union Bound*, we can prove that

$$\mathbb{E}_\eta[I_{C,\mathsf{BLR}}] \le \mathbb{E}_\eta[I_C] + \mathbb{E}_\eta[I'_{C,\mathsf{BLR}}] + 6\eta \le 3(\delta + \varepsilon) + 15\eta.$$

Subsequently, by Proposition 3.12, $\mathsf{P}_\eta(T_{C,\mathsf{BLR}}) \le 3(\delta + \varepsilon) + \delta + 18\eta$. The result then follows from the Precision Consistency Axiom by setting $\eta \triangleq \beta/30$. ☐

4.5.3 Correctness of Majority Correction

We move on to prove the soundness of the BLR linearity testing. As a first step, we prove that if C passes the linearity testing, then the BLR self correction algorithm is *single-valued*. Formally:

Lemma 4.16 (Single-valuedness of BLR correction)**.** APX_1 *proves the following statement. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ *and* $C : \{0,1\}^n \to \{0,1\}$ *be a circuit. Let* $\varepsilon \in \mathbb{Q}$. *Assume that* $\varepsilon, \delta, \beta \le 0.01$. *Define the following circuits:*

- *Let* $T_{C,\mathsf{BLR}}(x,y)$ *be the circuit that outputs* 1 *if and only if* $C(x) \oplus C(y) \neq C(x \oplus y)$.
- *Let* $D_{x,b}(r) : \{0,1\}^n \to \{0,1\}$ *be the circuit that outputs* 1 *if and only if* $C(x \oplus r) \oplus C(r) = b$.

Suppose that $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \le \varepsilon$. *For every* $x \in \{0,1\}^n$, $\mathsf{P}_\delta(D_{x,b}) \ge 1 - 4\varepsilon - (4\delta + \beta)$ *for some* $b \in \{0,1\}$.

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, the circuit C, and $\varepsilon \in \mathbb{Q}$. Let $T_{C,\mathsf{BLR}}$ and $D_{x,b}$ be the circuits as defined above, and $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Suppose that $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \le \varepsilon$, we know by the Precision Consistency Axiom that

$$\mathsf{P}_\eta(T_{C,\mathsf{BLR}}) \le \varepsilon + \delta + 2\eta. \tag{4.35}$$

Fix any $x \in \{0,1\}^n$. Let $D'_x : \{0,1\}^{2n} \to \{0,1\}$ be circuit that takes $(r_1, r_2) \in \{0,1\}^n$ are the input, and outputs 1 if $C(x \oplus r_1) \oplus C(r_1) = C(x \oplus r_2) \oplus C(r_2)$. Let Y be the indicator random variable of D'_x, and $\overline{Y}$ be the indicator random variable of $1 - D'_x$. It follows from *Averaging Argument for Expectation* that $\mathbb{E}_\eta[Y] \ge 1 - \mathbb{E}_\eta[\overline{Y}] - 6\eta$.

Consider the following two circuits $T, T' : \{0,1\}^{2n} \to \{0,1\}$:

- $T(r_1, r_2) \triangleq 1$ if and only if $C(r_1) \oplus C(r_2) \neq C(r_1 \oplus r_2)$.
- $T'(r_1, r_2) \triangleq 1$ if and only if $C(x \oplus r_1) \oplus C(x \oplus r_2) \neq C((x \oplus r_1) \oplus (x \oplus r_2))$.

Let X, X' be the indicator random variable of T and T', respectively. It is clear that T is exactly $T_{C,\mathsf{BLR}}$, and thus by Equation (4.35), $\mathsf{P}_\eta(T) \le \varepsilon + \delta + 2\eta$. Similarly, as T'_i is obtained $T_{C,\mathsf{BLR}}$ by taking bitwise-XOR to the input string with the fixed string (x, x), by Equation (4.35) and the *Re-randomization Lemma*, $\mathsf{P}_\eta(T') \le \eta + \delta + 5\eta$.

Moreover, one can observe that $D'_x(x, y) = 0$ implies that either $T(x, y)$ or $T'(x, y)$ outputs 1: This is because if $T(x, y) = T'(x, y) = 0$, we can conclude that

$$C(x \oplus r_1) \oplus C(x \oplus r_2) \oplus C(r_1) \oplus C(r_2) = C(r_1 \oplus r_2) \oplus C((x \oplus r_1) \oplus (x \oplus r_2)) = 0,$$

which implies that $D'_x(x, y) = 1$. Subsequently, by the *Union Bound*, we have

$$\mathbb{E}_\eta[\overline{Y}] \le \mathbb{E}_\eta[X_i] + \mathbb{E}_\eta[X'_i] + 3\eta \le 2(\varepsilon + \delta) + 10\eta.$$

and thus $\mathbb{E}_\eta[Y] \ge 1 - \mathbb{E}_\eta[\overline{Y}] - 6\eta \ge 1 - 2(\varepsilon + \delta) - 16\eta$.

Let I_x, I'_x be explicitly i.i.d. RVs over $\{0,1\}$ defined by the circuit $D_x(r) \triangleq C(x \oplus r) \oplus C(r)$, and Y_c is the indicator random variable of $I_x = I'_x$. By definitions, we can see that $Y_c|_\rho = Y|_\rho$ for any assignment ρ, and thus by the *Averaging Argument for Expectation*,

$$\mathbb{E}_\eta[Y_c] \ge \mathbb{E}_\eta[Y] - 6\eta \ge 1 - 2(\varepsilon + \delta) - 22\eta.$$

Subsequently, by Lemma 4.14, we have

$$\left|\mathbb{E}_\eta[I_x] - \frac{1}{2}\right| \ge \sqrt{\frac{1 - 2(\varepsilon + \delta) - 22\eta}{2} - \frac{1}{4} - 6\eta} \ge \sqrt{\frac{1}{4} - (\varepsilon + \delta + 17\eta)} \ge \frac{1}{2} - 4(\varepsilon + \delta + 17\eta). \tag{4.36}$$

Recall that I_x is the random variable that takes $(r_1, r_2) \in \{0,1\}^{2n}$ as random seed and outputs $C(x \oplus r_1) \oplus C(r_1)$. Suppose that $\mathbb{E}_\eta[I_x] \ge 1 - 4(\varepsilon + \delta + 17\eta)$. By *Averaging Argument for Expectation*, there is an assignment ρ of the second part r_2 of the seed (which was for I'_x) such that

$$\mathbb{E}_\eta[I_x|_\rho] \ge \mathbb{E}_\eta[I_x] - 6\eta \ge 1 - 4\eta - (4\delta + 74\eta).$$

As $I_x|_\rho$ is the indicator random variable of $D_{x,1}$, it follows from Proposition 3.12 that $\mathsf{P}_\eta(D_{x,1}) \ge \mathbb{E}_\eta[I_x] - 3\eta \ge 1 - 4\varepsilon - (4\delta + 74\eta)$. It suffices if we set $\eta \le \beta/100$. The other case $\mathbb{E}_\eta[I_x] \le 4(\varepsilon + \delta + 17\eta)$ can be resolved by considering $\overline{I_x} \triangleq 1 - I_x$. ☐

Lemma 4.16 shows that the BLR self correction algorithm is single-valued assuming that the circuit C passes the linearity testing. Second, we show that the "corrected" function $g(\cdot)$ satisfies that $g(x) \oplus g(y) = g(x \oplus y)$ for *every* $x, y \in \{0,1\}^n$. Formally:

Lemma 4.17 (Linearity of BLR correction)**.** APX_1 *proves the following statement. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ *and* $C : \{0,1\}^n \to \{0,1\}$ *be a circuit. Let* $\varepsilon \in \mathbb{Q}$*. Assume that* $\varepsilon, \delta, \beta \le 0.01$*. Let* $D_{x,b}$ *be the circuit in Lemma 4.16, and* $g(x)$ *be the* P*-oracle circuit that works as follows: Given* $x \in \{0,1\}^n$*, it outputs* $b \in \{0,1\}$ *if* $\mathsf{P}_\delta(D_{x,b}) \ge 1 - 4\varepsilon - (4\delta + \beta)$*, and* $\perp$ *otherwise.*

Let $T_{C,\mathsf{BLR}}(x,y)$ *be the circuit that outputs 1 if and only if* $C(x) \oplus C(y) \neq C(x \oplus y)$*. Suppose that* $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \le \varepsilon$*. Then for every* $x_1, x_2 \in \{0,1\}^n$*,* $g(x_1) \oplus g(x_2) = g(x_1 \oplus x_2)$*.*

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, the circuit C, and $\varepsilon \in \mathbb{Q}$. Assume that $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \le \varepsilon$. Note that by Lemma 4.16 and $\varepsilon, \delta, \beta < 0.01$, $g(x) \in \{0,1\}$ for every $x \in \{0,1\}^n$. Fix $x_1, x_2 \in \{0,1\}^n$ and let $b_j \triangleq g(x_j)$ for $j \in \{1,2\}$, and $b_\oplus \triangleq g(x_1 \oplus x_2)$.

Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later, and $X_1, X_2, X_\oplus$ be the random variables over $\{0,1\}$ with seed $r \in \{0,1\}^n$ defined as follows:

- X_1 outputs 1 if and only if $b_1 = C(x_1 \oplus r) \oplus C(r)$.
- X_2 outputs 1 if and only if $b_2 = C(x_2 \oplus r) \oplus C(r)$.
- $X_\oplus$ outputs 1 if and only if $b_\oplus = C(x_1 \oplus x_2 \oplus (x_2 \oplus r)) \oplus C(x_2 \oplus r)$.

Note that as $b_j \triangleq g(x_j)$, we know by the definition of g that $\mathsf{P}_\delta(D_{x_j,b_j}) \ge 1 - 4\varepsilon - (4\delta + \beta)$. It can be observed that X_1, X_2 are the indicator random variables of D_{x_1,b_1}, D_{x_2,b_2}, thus by Proposition 3.12 and the *Precision Consistency of Expectation*,

$$\mathbb{E}_\eta[X_1], \mathbb{E}_\eta[X_2] \ge 1 - 4\varepsilon - (5\delta + \beta - 2\eta).$$

Moreover, $X_\oplus$ is the indicator variable of the circuit that outputs 1 if $b_\oplus = C(x_1 \oplus x_2 \oplus (x_2 \oplus r)) \oplus C(x_2 \oplus r)$, and the circuit is obtained from $D_{x_1 \oplus x_2, b_\oplus}$ by taking XOR to the input with a fixed string x_2. Therefore, by Proposition 3.12 and the *Re-randomization Lemma*, we have $\mathbb{E}_{0.01}[X_\oplus] \ge 1 - 4\varepsilon - (5\delta + \beta - 2\eta)$.

Let $\overline{X}_1 \triangleq 1 - X_1$, $\overline{X}_2 \triangleq 1 - X_2$, and $\overline{X}_\oplus \triangleq 1 - X_\oplus$. Let $Y \triangleq \overline{X}_1 \vee \overline{X}_2 \vee \overline{X}_\oplus$ and $\overline{Y} \triangleq 1 - Y$. Then using *Complementary Counting* and *Union Bound*, we have

$$\begin{aligned}
\mathbb{E}_\eta[Y] &\le \mathbb{E}_\eta[\overline{X}_1] + \mathbb{E}_\eta[\overline{X}_2] + \mathbb{E}_\eta[\overline{X}_\oplus] + 9\eta && (\textit{Union Bound}) \\
&\le (1 - \mathbb{E}_\eta[X_1]) + (1 - \mathbb{E}_\eta[X_2]) + (1 - \mathbb{E}_\eta[X_\oplus]) + 18\eta && (\textit{Complementary Counting}) \\
&\le 12\varepsilon + 15\delta + \beta + 24\eta.
\end{aligned}$$

Again, using *Complementary Counting*, we have $\mathbb{E}_\eta[\overline{Y}] \ge 1 - (12\varepsilon + 15\delta + \beta + 24\eta)$. By setting $\eta \triangleq \beta/100$, we have $\mathbb{E}_\eta[\overline{Y}] > 3\eta$.

By *Averaging Argument for Expectation*, there exists an assignment ρ such that $\mathbb{E}_{0.01}[\overline{Y}|_\rho] \ge 0.05$, or in other words, $\overline{Y}|_\rho = 1$ as its seed length is 0 after applying the restriction ρ. By the definition of the random variables, this indicates that

$$\begin{aligned}
b_1 &= C(x_1 \oplus r) \oplus C(r), \\
b_2 &= C(x_2 \oplus r) \oplus C(r), \\
b_\oplus &= C(x_1 \oplus x_2 \oplus (x_2 \oplus r)) \oplus C(x_2 \oplus r) = C(x_1 \oplus r) \oplus C(x_2 \oplus r).
\end{aligned}$$

It immediately follows that $b_\oplus = b_1 \oplus b_2$. □

4.5.4 Soundness of BLR Linearity Testing

Now we are ready to prove the soundness of the BLR linearity testing. At a high level, we will recover the string $z \in \{0,1\}^n$ that defines the linear function using the oracle circuit $g(\cdot)$. It is worth noting that the correctness proof of the string z is quite non-trivial: It crucially builds on the tools for random variables developed in Section 3, especially *Averaging Argument for Expectation*.

Theorem 4.12 (Soundness of BLR linearity testing)**.** APX_1 *proves the following statement. Let* $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$ *and* $C : \{0,1\}^n \to \{0,1\}$ *be a circuit. Let* $\varepsilon \in \mathbb{Q}$*. Assume that* $\varepsilon, \delta, \beta < 0.01$*. We define the following circuits:*

- *For every $z \in \{0,1\}^n$, let $T_{C,z}(x)$ be the circuit that outputs 1 if and only if $C(x) \neq \langle x, z\rangle \bmod 2$.*
- *Let $T_{C,\mathsf{BLR}}(x,y)$ be the circuit that outputs 1 if and only if $C(x) \oplus C(y) \neq C(x \oplus y)$.*

Suppose that $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \leq \varepsilon$. Then there exists a string $z \in \{0,1\}^n$ such that $\mathsf{P}_\delta(T_{C,z}) \leq 5\varepsilon + 6\delta + \beta$.

Proof. We argue in APX_1. Fix $n, \delta^{-1}, \beta^{-1} \in \mathsf{Log}$, the circuit C, and $\varepsilon \in \mathbb{Q}$. Let $T_{C,z}$ and $T_{C,\mathsf{BLR}}$ be the circuits as defined above, and $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Suppose that $\mathsf{P}_\delta(T_{C,\mathsf{BLR}}) \leq \varepsilon$, we know by the PRECISION CONSISTENCY AXIOM that

$$\mathsf{P}_\eta(T_{C,\mathsf{BLR}}) \leq \varepsilon + \delta + 2\eta. \tag{4.37}$$

Let $D_{x,b}(r) \triangleq C(x \oplus r) \oplus C(r) \oplus (1-b)$ be the circuit in Lemma 4.16, and $g(x)$ be the oracle circuit in Lemma 4.17. By Lemma 4.16, we have that for every $x \in \{0,1\}^n$, there is a $b \in \{0,1\}$ such that

$$\mathsf{P}_\eta(D_{x,b}) \geq 1 - 4(\varepsilon + \delta + 2\eta) - 5\eta \geq 1 - 4(\varepsilon+\delta) - 13\eta. \tag{4.38}$$

Note that we will choose η such that $13\eta < 0.01$. Therefore, by Lemma 4.17 that for every $x \in \{0,1\}^n$, the bit b satisfying Equation (4.38) is given by $g(x)$. Moreover, for every $x_1, x_2 \in \{0,1\}^n$, we have that $g(x_1) \oplus g(x_2) = g(x_1 \oplus x_2)$.

Let e_i be the string that is 0 on all but the i-th bit, and $z \triangleq g(e_1) \circ g(e_2) \circ \cdots \circ g(e_n)$. That is, $z_i = g(e_i)$ for every $i \in [n]$. Let X and Y be random variables over $\{0,1\}$ that take a seed $(x,r) \in \{0,1\}^n \times \{0,1\}^n$ of length $2n$ and are defined as follows.

- X outputs 1 if $C(x \oplus r) \oplus C(x) \neq C(r)$. That is, X is the indicator random variable of $T_{C,\mathsf{BLR}}$. By Proposition 3.12,
$$\mathbb{E}_\eta[X] \leq \mathsf{P}_\eta(T_{C,\mathsf{BLR}}) + 3\eta \leq \varepsilon + \delta + 5\eta. \tag{4.39}$$
- Y outputs 1 if $C(x) \neq \langle x, z\rangle \bmod 2$. Note that for every assignment r to the second part of its seed, $Y|_r$ is the indicator random variable of $T_{C,z}$.

Next, we will prove that for every $x \in \{0,1\}^n$,

$$|\mathbb{E}_\eta[X|_x] - \mathbb{E}_\eta[Y|_x]| \leq 4(\varepsilon+\delta) + 13\eta. \tag{4.40}$$

Recall that by the definition of X and $D_{x,b}$, we have that for every assignment x to the *first* part of their seeds, $X|_x$ is the indicator random variable of $D_{x,1}$, and for $\overline{X} \triangleq 1 - X$, $\overline{X}|_x$ is the indicator random variable of $D_{x,0}$. Therefore, by Lemma 4.16,

$$\max\{\mathbb{E}_\eta[X|_x], 1 - \mathbb{E}_\eta[X|_x]\} \geq 1 - 4(\varepsilon + \delta + 2\eta) - 5\eta \geq 1 - 4(\varepsilon+\delta) - 13\eta.$$

We consider the case that $\mathbb{E}_\eta[X|_x] \geq 1 - 4(\varepsilon+\delta) - 13\eta$, and the other case is similar. By the definition of g, we know that $g(x) = 1$, and subsequently

$$\langle x, z\rangle \bmod 2 = \sum_{i \in [n], x_i = 1} g(e_i) \bmod 2 = g(x),$$

where the last equality follows from Lemma 4.17 and the 1-INDUCTION AXIOM (note that the induction axiom suffices as $n \in \mathsf{Log}$). For any assignment r to the second part of the seed, we have $\mathbb{E}_\eta[Y|_x|_y] = 1$, which subsequently implies that $\mathbb{E}_\eta[Y|_x] = 1$. Therefore, for any assignment $x \in \{0,1\}^n$ to the first part of the seed,

$$|\mathbb{E}_\eta[X|_x] - \mathbb{E}_\eta[Y|_x]| \leq 4(\varepsilon+\delta) + 13\eta.$$

By the *Averaging Argument for Expectation*, we have $|\mathbb{E}_\eta[X] - \mathbb{E}_\eta[Y]| \leq 4(\varepsilon+\delta) + 19\eta$, and thus

$$\mathbb{E}_\eta[Y] \leq \mathbb{E}_\eta[X] + 4(\varepsilon+\delta) + 19\eta \leq e.$$

Again, by the *Averaging Argument for Expectation*, there is an assignment $r \in \{0,1\}^n$ to the second part of its seed such that

$$\mathbb{E}_\eta[Y|_r] \leq \mathbb{E}_\eta[Y] + 3\eta \leq 5(\varepsilon+\delta) + 27\eta. \tag{4.41}$$

Fix the assignment r. As mentioned above, $\mathbb{E}_\eta[Y|_r]$ is the indicator random variable of $T_{C,z}$, and thus by Proposition 3.12 and Precision Consistency Axiom,

$$\mathsf{P}_\delta(T_{C,z}) \le \mathsf{P}_\eta(T_{C,z}) + \delta + 2\eta \le \mathbb{E}_\eta[Y|_r] + \delta + 5\eta \le 5\varepsilon + 6\delta + 32\eta.$$

It completes the proof by taking $\eta \triangleq \beta/50$. □

5 Witnessing Theorems and Relative Strength of APX_1

In this section, we prove a witnessing theorem for APX_1 and consider its relation to other theories of bounded arithmetic, including PV_1 and APC_1.

5.1 Provably Total TFNP Problems in APX_1

In this subsection, we will introduce a witnessing theorem for the $\forall\Sigma_1^b$-consequences of APX_1 (i.e. provably total TFNP problems in APX_1).

5.1.1 A TFZPP Problem: Refuter(Yao)

We will first introduce a TFZPP problem[23] called *Refutation of Yao-Predictor Generators*; we denote it by Refuter(Yao). Recall that Yao's distinguisher-to-predictor transformation [Yao82] (see Section 4.1) shows that if a distribution $\mathcal{D}$ over $\{0,1\}^n$ is not ε-pseudorandom, i.e., there is a circuit $C : \{0,1\}^n \to \{0,1\}$ (called *distinguisher*) such that

$$\left|\Pr[C(\mathcal{D})] - \Pr[C(\mathcal{U}_n)]\right| > \varepsilon,$$

then there exists $i \in [n]$ and a *predictor* $P_i : \{0,1\}^{i-1} \to \{0,1\}$ such that

$$\Pr_{x\leftarrow\mathcal{D}}[P(x_{<i}) = x_i] \ge \frac{1}{2} + \frac{\varepsilon}{4n},$$

i.e., P predicts the i-th bit of $\mathcal{D}$ with advantage at least $\varepsilon/4n$. This transformation serves as a key step in the construction and analysis of pseudorandom generators (see, e.g., [NW94, IW97]): it shows that an unpredictable distribution is necessarily pseudorandom.

In the statement below, we say that a discrete probability distribution $\mathcal{D}$ is *flat* if it is uniform over its support, i.e., over the set of elements with non-zero probability over $\mathcal{D}$. The *size* of the distribution is the size of its support. We will represent flat distributions explicitly as a list of strings. In the subsequent discussions, we might tacitly assume that the relevant distribution is flat and explicitly represented.

Definition 5.1. The search problem Refuter(Yao) is defined as follows.

- (*Parameters*). Length of strings n, distribution size m, predictor size s, and advantage $\delta \in [0,1]$.
- (*Input*). A circuit $G : \{0,1\}^{nm} \to [n] \times \{0,1\}^s$ (called *predictor generator*).
- (*Solution*). Any explicit flat distribution $\mathcal{D} \in (\{0,1\}^n)^m$ of size m such that the following holds:

  Let $(i, P) \triangleq G(\mathcal{D})$, where $P : \{0,1\}^{i-1} \to \{0,1\}$ is parsed as a circuit of description length $\le s$. Then

  $$\Pr_{x\leftarrow\mathcal{D}}[P(x_{<i}) = x_i] < \frac{1}{2} + \delta.$$

  In other words, P is not a predictor of the i-th bit of $\mathcal{D}$ with advantage δ.

[23] A search problem is said to be in TFZPP if it is a TFNP problem solvable by randomized polynomial-time algorithms.

For concreteness, one may think of the parameter regime $m = n^{10}$, $s = n^2$, and $\delta = 0.1$. In this case, a random distribution of m strings of length n is likely $o(1)$-pseudorandom against any circuit of size s, and thus must be a solution of $\mathsf{Refuter}(\mathsf{Yao})$ no matter the input circuit G.

At a high level, $\mathsf{Refuter}(\mathsf{Yao})$ asks to generate a distribution $\mathcal{D}$ that is unpredictable against a *given* predictor generator G — a deterministic algorithm that aims to output a predictor P for $\mathcal{D}$. The distribution $\mathcal{D}$ is not necessarily an unpredictable (or equivalently, pseudorandom) distribution against *small circuits*; it suffices to fool the given *deterministic* predictor generator G. This makes it a special case of constructing targeted PRGs, which is known to be prBPP-complete (see, e.g., [Gol11, CT21, LPT24]).

5.1.2 Connection to $\mathsf{LossyCode}$

A closely related TFZPP relation is the *Lossy Code Problem*; we denote it by $\mathsf{LossyCode}$. Inspired by the literature in bounded arithmetic (see [Jeř07a, Section 3.1] and the discussion below), the problem is defined in [Kor22] as a more feasible variant of the Range Avoidance Problem; see [Kor25] and references therein for an introduction to this line of work.

Definition 5.2. The search problem $\mathsf{LossyCode}$ is defined as follows.

- (*Input*). Circuits $C : \{0,1\}^n \to \{0,1\}^{n-1}$ and $D : \{0,1\}^{n-1} \to \{0,1\}^n$. These two circuits are called *compressor* and *decompressor*, respectively.
- (*Output*). A string $x \in \{0,1\}^n$ such that $D(C(x)) \neq x$.

It is clear that $\mathsf{LossyCode} \in \mathsf{TFZPP}$. Indeed, Wilkie (unpublished) and Thapen [Tha02] proved that the problem captures the $\forall\Sigma_1^b$-fragment of the theory APC_1.

Theorem 5.3 ([Jeř04, Proposition 1.14], also see [LPT24, Theorem D.1])**.** *Let $\phi(x, y)$ be a quantifier-free formula in the language of APC_1 that only has x and y as open variables. If $\mathsf{APC}_1 \vdash \forall x\ \exists y\ \phi(x, y)$, then there is a deterministic polynomial-time reduction from the following problem to $\mathsf{LossyCode}$: Given $n \in \mathbb{N}$, output $m \in \mathbb{N}$ such that $\phi(n, m)$ is true in the standard model.*

Moreover, it has been recently discovered that some natural TFZPP-search problems admit deterministic reductions to $\mathsf{LossyCode}$ or its variants: constructing large prime numbers with factoring oracles [Kor22] and the simulation of catalytic logspace machines [CLMP25]. Variants of $\mathsf{LossyCode}$ are relevant to both full and partial derandomizations of prBPP; see [LPT24] for a comprehensive introduction.

Note that assuming $\mathsf{prBPP} = \mathsf{prP}$, both $\mathsf{LossyCode}$ and $\mathsf{Refuter}(\mathsf{Yao})$ are in FP. Nevertheless, it is interesting to discover the relative hardness of their derandomization. By adapting an idea from [Kor22], we show that $\mathsf{Refuter}(\mathsf{Yao})$ admits a deterministic polynomial-time reduction to $\mathsf{LossyCode}$. Therefore, showing that $\mathsf{Refuter}(\mathsf{Yao}) \in \mathsf{FP}$ is necessary before proving that $\mathsf{LossyCode} \in \mathsf{FP}$.

Theorem 5.4 (Implicit in the proof of [Kor22, Corollary 41])**.** *There is a deterministic polynomial-time mapping reduction from $\mathsf{Refuter}(\mathsf{Yao})$ with parameters $(\delta^2/10) \cdot m \geq s + \lceil \log n \rceil + 1$ to $\mathsf{LossyCode}$.*

Proof. Note that we can encode m-bit strings with Hamming weight at most k by $\log_2 \binom{m}{k} + O(\log k)$ bits, where the encoding and decoding algorithms run in polynomial time (see, e.g., [CLO24, Lemma 5.4]). In particular, when $k \triangleq (1/2 - \delta) \cdot m$ and m is sufficiently large, the encoding length is

$$\begin{aligned} &\log_2 \binom{m}{(1/2 - \delta) \cdot m} + O(\log(1/2 - \delta) + \log m) \\ &\leq m + \log_2(e^{-(2\delta)^2 m/4}) + O(\log(1/2 - \delta) + \log m) \\ &\leq m - (\delta^2/10) \cdot m, \end{aligned}$$

where the first inequality follows from the Chernoff bound.

Now we describe the reduction. Given any predictor generator $G : \{0,1\}^{nm} \to [n] \times \{0,1\}^s$, consider the following compressor $C : \{0,1\}^{nm} \to \{0,1\}^{nm-1}$ and decompressor $D : \{0,1\}^{nm-1} \to \{0,1\}^{nm}$:

- (*Compressor*). Given any $\mathcal{D} \in \{0,1\}^{nm}$, the compressor parses it as a distribution over n-bit strings of size m. It computes $(i, P) \triangleq G(\mathcal{D})$. If P fails to predict the i-th bit of $\mathcal{D}$ with advantage δ, it fails and aborts. Otherwise,

$$\Pr_{x \leftarrow \mathcal{D}}[P(x_{<i}) = x_i] \geq \frac{1}{2} + \delta.$$

  Let y be the m-bit string defined as $y_j \triangleq P(x^{(j)}_{<i}) \oplus x^{(j)}_i$, where $x^{(j)}$ is the j-th string in $\mathcal{D}$. Then y is a string of Hamming weight at most $(1/2-\delta)\cdot m$, and thus can be efficiently encoded using $m-(\delta^2/10)\cdot m$ bits.

  Let $\hat{y}$ be its encoding, and $\mathcal{D}_{-i} \in \{0,1\}^{m(n-1)}$ be the distribution $\mathcal{D}$ after removing the i-th bit from all strings. The compressor outputs the tuple $(i, P, \hat{y}, \mathcal{D}_{-i})$, which is of length at most

$$\lceil \log n \rceil + s + (m - (\delta^2/10) \cdot m) + (nm - m) < nm$$

  due to the assumption on parameters.

- (*Decompressor*). When the compressor does not fail, the decompressor can recover $\mathcal{D}$ from $(i, P, \hat{y}, \mathcal{D}_{-i})$ by first recovering y then computing the missing bits

$$x^{(j)}_i \triangleq P(x^{(j)}_{<i}) \oplus y_j.$$

Given a predictor generator G and parameters as above, the mapping reduction from Refuter(Yao) to LossyCode outputs (C, D) as an instance of LossyCode.

It suffices to prove that the reduction is correct. Given any string $\mathcal{D}$ such that $D(C(\mathcal{D})) \neq \mathcal{D}$, we know by the discussion above that the compressor must fail. In other words, $G(\mathcal{D})$ fails to produce a predictor with advantage δ. This means that $\mathcal{D}$ is a solution to the Refuter(Yao) instance and thus concludes the proof. □

5.1.3 The Witnessing Theorem

We are now ready to show the following witnessing theorem for APX_1: any provably total TFNP problem in APX_1 is deterministically reducible to Refuter(Yao).

Theorem 1.9 (Witnessing for APX_1)**.** *Let $\varphi(x, y)$ be a quantifier-free formula in the language of PV_1. If $\mathsf{APX}_1 \vdash \forall x\ \exists y\ \varphi(x, y)$, there exists a deterministic polynomial-time Turing reduction from the search problem defined by φ to* Refuter(Yao) *with parameters satisfying $(\delta^2/10) \cdot m \geq s + \lceil \log n \rceil + 1$.*

Note that the inequality $(\delta^2/10) \cdot m \geq s + \lceil \log n \rceil + 1$ implies that the Refuter(Yao) instance reduces to LossyCode and, in particular, it is a total search problem. A more refined analysis of our proof may lead to an improved trade-off between the parameters, which we leave for future work.

To prove this witnessing theorem, we will need the standard Herbrand's theorem for universal first-order theories and a lemma that extracts a predictor from an APX proof. The latter requires a proof-theoretic analysis and is deferred to the end of the section (see Section 5.4).

Theorem 5.5 (Herbrand's Theorem; see, e.g., [Bus94])**.** *Let $\mathcal{T}$ be a universal first-order theory and $\varphi(x, y)$ be a quantifier-free formula with only x and y as open variables. If $\mathcal{T} \vdash \forall x\ \exists y\ \varphi(x, y)$, there exists a constant $c \in \mathbb{N}$ and terms $t_1, t_2, \ldots, t_c$ such that*

$$\mathcal{T} \vdash \forall x\ \bigvee_{i=1}^{c} \varphi(x, t_i(y))\,.$$

Lemma 5.6 (Predictor Extraction Lemma)**.** *Let $t_1(\vec{x}) = t_2(\vec{x})$ be an equation provable in* APX. *Then there are polynomials $k(n)$, $m(n)$, and a deterministic polynomial-time algorithm E that satisfies the following conditions when n is sufficiently large:*

- (*Input*). *A string* $\vec{x} \in \{0,1\}^n$ *and a flat distribution* $\mathcal{D} \in (\{0,1\}^k)^m$ *of size* m *over* k*-bit strings.*
- (*Simulation of Terms*). *Recall that* t_1, t_2 *are interpreted as polynomial-time* P*-oracle algorithms in standard models. We will simulate the algorithms on input* $\vec{x}$ *as follows: For every oracle query* $\mathsf{P}(C, \Delta)$*, where* $C : \{0,1\}^t \to \{0,1\}$*, we will ensure that* $t \le k(n)$ *and answer the query by*

$$\mathsf{P}(C, \Delta) \triangleq \Pr_{u \leftarrow \mathcal{D}}[C(u_{\le t})]. \tag{5.1}$$

  We denote the output of t_1 *in the simulation as* $t_1^{\mathcal{D}}(\vec{x})$*, and the output of* t_2 *as* $t_2^{\mathcal{D}}(\vec{x})$*.*
- (*Output*). *Suppose that* $t_1^{\mathcal{D}}(\vec{x}) \neq t_2^{\mathcal{D}}(\vec{x})$*. Then* $E(\vec{x}, \mathcal{D})$ *outputs* $i \in [k(n)]$ *and a circuit* $P \colon \{0,1\}^{i-1} \to \{0,1\}$ *of size at most* s *such that* P *predicts the* i*-th bit of* $\mathcal{D}$ *with advantage* δ *such that*

$$(\delta^2/10) \cdot m(n) \ge s + \lceil \log k(n) \rceil + 1.$$

Proof of Theorem 1.9. Recall that APX_1 admits a universal axiomatization (see Proposition 2.8). Suppose that $\mathsf{APX}_1 \vdash \forall x\ \exists y\ \varphi(x, y)$. By Herbrand's theorem, there are terms $t_1, \ldots, t_c$ in the language of APX_1 such that

$$\mathsf{APX}_1 \vdash \forall x\ \bigvee_{i=1}^{c} \varphi(x, t_i(y)),$$

for some $c \in \mathbb{N}$.

Note that the language of APX_1 is the language of PV extended by the approximate counting oracle P; therefore, $t_1, \ldots, t_c$ are polynomial time P-oracle algorithms in the standard model. Let t_φ be a term in APX such that

$$\mathsf{APX}_1 \vdash \bigvee_{i=1}^{c} \varphi(x, t_i(x)) \leftrightarrow t_\varphi(x, t_1(x), \ldots, t_c(x)) = 1.$$

This can be done as φ is a quantifier-free formula; see, e.g., [Li25, Chapter 3]. Then we know that APX_1 proves that $\forall x\ t_\varphi(x, t_1(x), \ldots, t_c(x)) = 1$. As APX_1 is conservative over APX (see Proposition 2.9), we know that $\mathsf{APX} \vdash t_\varphi(x, t_1(x), \ldots, t_c(x)) = 1$.

By Lemma 5.6 (instantiated with $t_1 \triangleq t_\varphi(x, t_1(x), \ldots, t_c(x))$ and $t_2 \triangleq 1$), there are $k = k(n)$, $m = m(n)$, $s = s(n)$, $\delta = \delta(n)$ and a polynomial time E such that the following holds. Given $x \in \{0,1\}^n$ and a distribution $\mathcal{D} \in (\{0,1\}^k)^m$ of size m,

- either $t_\varphi^{\mathcal{D}}(x, t_1^{\mathcal{D}}(x), \ldots, t_k^{\mathcal{D}}(x)) = 1$; or
- $E(x, \mathcal{D})$ outputs $i \in [k(n)]$ and P that predicts the i-th bit of $\mathcal{D}$ with advantage δ, where $(\delta(n)^2/10) \cdot m(n) \ge s(n) + \lceil \log k(n) \rceil + 1$.

The reduction produces the circuit $E(x, \cdot)$ as an instance of $\mathsf{Refuter}(\mathsf{Yao})$, where the size-$m(n)$ distribution is supported over $k(n)$-bit strings, the predictor size is $s(n)$, and the advantage is $\delta(n)$. Given x of length n, for any solution $\mathcal{D}$ to the resulting instance $E(x, \cdot)$ of $\mathsf{Refuter}(\mathsf{Yao})$, we know by definition that $E(x, \mathcal{D})$ cannot output a predictor with advantage δ. As a result, the first bullet above must hold:

$$t_\varphi^{\mathcal{D}}(x, t_1^{\mathcal{D}}(x), \ldots, t_k^{\mathcal{D}}(x)) = 1.$$

Subsequently, given any solution $\mathcal{D}$ to the instance of $\mathsf{Refuter}(\mathsf{Yao})$, one of $t_1^{\mathcal{D}}(x), t_2^{\mathcal{D}}(x), \ldots, t_k^{\mathcal{D}}(x)$ must output y such that $\varphi(x, y)$ holds. This gives a correct reduction, as simulations of $t_1^{\mathcal{D}}(x), t_2^{\mathcal{D}}(x), \ldots, t_k^{\mathcal{D}}(x)$ can be implemented in deterministic polynomial time given x and the explicit description of $\mathcal{D}$. ☐

5.2 Relationship to PV_1: Is $\mathsf{prBPP} = \mathsf{prP}$ Feasibly Provable?

In this subsection, we introduce a few questions regarding the relative strength of APX_1 and PV_1. We will discuss their importance and connection to the program of proving $\mathsf{prBPP} = \mathsf{prP}$. No meaningful progress is reported in the paper; we believe the resolution of the questions, even conditionally, would advance our understanding of feasible mathematics and derandomization.

Feasible proof of $\mathsf{prBPP} = \mathsf{prP}$. One major open problem in complexity theory is whether derandomization is possible in general with polynomial runtime overhead. The seminal work of Nisan, Wigderson, and Impagliazzo [NW94, IW97] shows that $\mathsf{prBPP} = \mathsf{prP}$ follows from exponential circuit lower bounds for $\mathsf{E} = \mathsf{DTIME}[2^n]$; hence many researchers expect a positive answer. However, despite enormous efforts, both $\mathsf{prBPP} = \mathsf{prP}$ and the circuit lower bounds for E remain open.

From the perspective of meta-mathematics, an interesting question is to study whether $\mathsf{prBPP} = \mathsf{prP}$ is *(un)provable* in a weak arithmetic theory such as PV_1. A technical challenge is that, as the language of PV_1 is designed to capture *deterministic* polynomial time computable functions, it is a priori not obvious how to formalize the statement $\mathsf{prBPP} = \mathsf{prP}$, which involves the acceptance probability of circuits over inputs from a set of exponential size.

We propose the investigation of the following related question.

Open Problem 3. Is there a PV function symbol $\mathsf{P}(C, \Delta)$ such that the Basic Axiom, Boundary Axiom, Precision Consistency Axiom, and Local Consistency Axiom are provable in PV_1?

We note that an unconditional positive answer is unlikely to be obtained in the near future, as it immediately implies $\mathsf{prBPP} = \mathsf{prP}$ by the soundness of PV_1 and Theorem 2.5. Indeed, a positive answer shows, intuitively, that $\mathsf{prBPP} = \mathsf{prP}$ admits a *deterministic polynomial-time proof.* To our knowledge, it is unclear whether a positive or negative answer is more plausible.

Feasibly provable derandomization for deterministic statements. On the other hand, we may also consider a weaker collapse: it is in principle possible that, despite that there may not be a PV function symbol P such that the relevant axioms of APX_1 are provable in PV_1, the introduction of the oracle P does not help in proving any sentence that *does not* involve the oracle P. Formally:

Open Problem 4. Is APX_1 *conservative* over PV_1? In other words, is it the case that every first-order sentence in the language of PV that is provable in APX_1 is also provable in PV_1?

It is clear that a positive answer to Open Problem 3 implies a positive answer to Open Problem 4. Moreover, a positive answer of this open problem would immediately imply a witnessing theorem that improves Theorem 1.9: any APX_1 provably total TFNP relation (expressed by a quantifier-free formula in the language of PV_1) is in FP. This is because PV_1 provably total TFNP problems are in FP (see, e.g., [Oli25, Section 3.1]).

An interesting characteristic of Open Problem 4 is that it appears to be *incomparable* with $\mathsf{prBPP} = \mathsf{prP}$. If $\mathsf{prBPP} = \mathsf{prP}$ but the proof is not feasible, APX_1 may not be conservative over PV_1. More interestingly, if the answer to Open Problem 4 is positive, it is still unclear to us whether $\mathsf{prBPP} = \mathsf{prP}$ or any other nontrivial derandomization follows. Formally:

Open Problem 5. Suppose that APX_1 is conservative over PV_1. Does it follow that $\mathsf{prBPP} = \mathsf{prP}$, $\mathsf{prBPP} = \mathsf{prZPP}$, or any other unknown general derandomization result hold?

At a high level, this is to ask whether it is necessary to *derandomize computations in general* if we want to *derandomize proofs in general.* We contend that these problems are fundamental and merit deeper investigation.

5.3 Relationship to APC_1

We now study the relative strength of APX_1 and APC_1. We will show that, in a formal sense, APC_1 can be viewed as an extension of APX_1. We will then show that APC_1 is likely a *strict* extension of APX_1. Finally, we introduce a few open problems related to the relative strength of APC_1 and APX_1.

5.3.1 An Upper Bound: APC_1 Extends APX_1

We first prove an upper bound for APX_1 that is implicit in Jeřábek's results on approximate counting [Jeř07a]. In particular, this shows that a provable first-order sentence in the language of PV_1 is also provable in APC_1.

We start by defining a sentence $\mathsf{Hard}^{\mathsf{A}}_{\varepsilon}(\alpha)$ in the language of the relativized theory $\mathsf{PV}_1(\alpha)$. This sentence formalizes that $\alpha(1^{2^n})$ outputs a truth table of length 2^n that is $2^{-\varepsilon n}$-hard on average. Formally:

Definition 5.7 ([Jeř07a, Definition 2.1])**.** $\mathsf{Hard}^{\mathsf{A}}_{\varepsilon}(\alpha)$ is the following sentence in the language of $\mathsf{PV}_1(\alpha)$: For every $n \in \mathsf{LogLog}$ and x such that $||x|| = n$, $\alpha(x) \in \{0,1\}^{2^n}$ is a truth-table of a Boolean function f in n variables such that for every circuit $C : \{0,1\}^n \to \{0,1\}$ of size at most $2^{\varepsilon n}$,

$$\Pr_{x \leftarrow \{0,1\}^n}[C(x) = f(x)] \leq \frac{1}{2} + \frac{1}{2^{\varepsilon n}}.$$

Note that this probability is defined by a brute-force exact counting algorithm as $n \in \mathsf{LogLog}$.

Definition 5.8 ([Jeř07a, Definition 2.13])**.** The theory $\mathsf{HARD}^{\mathsf{A}}$ is defined as $\mathsf{PV}_1(\alpha) + \mathsf{dWPHP}(\mathsf{PV}(\alpha)) + \mathsf{Hard}^{\mathsf{A}}_{1/4}(\alpha)$, where $\mathsf{dWPHP}(\mathsf{PV}(\alpha))$ denotes the dual Weak Pigeonhole Principle for $\mathsf{PV}(\alpha)$ functions.

The following theorem can be proved using tools from [Jeř07a], where $\mathsf{NW}(\cdot,\cdot)$ is an instantiation of the Nisan-Wigderson PRG [NW94] with the hard truth table provided by $\alpha(\cdot)$. The proof is straightforward but requires familiarity with the theory APC_1; for completeness, we provide a proof of the theorem in Section 5.5.

Theorem 5.9 (Simulating $\mathsf{P}(C,\Delta)$ with $\mathsf{NW}(C,\Delta)$)**.** *For every $\varepsilon < 1/3$, there is a term $\mathsf{NW}(C,\Delta)$ in the language of $\mathsf{PV}_1(\alpha)$ such that* Basic Axiom*,* Boundary Axiom*,* Local Consistency Axiom*, and* Precision Consistency Axiom *are provable in $\mathsf{HARD}^{\mathsf{A}}$ when the oracle $\mathsf{P}(C,\Delta)$ is replaced by $\mathsf{NW}(C,\Delta)$.*

We note that by [Jeř07a, Theorem 2.13], the theory $\mathsf{HARD}^{\mathsf{A}}$ is a conservative extension of APC_1. It then immediately follows that:

Corollary 5.10. *Any first-order sentence in the language of PV_1 provable in APX_1 is also provable in APC_1.*

5.3.2 A Conditional Separation: APC_1 is Likely Stronger Than APX_1

As APX_1 is an alternative theory for polynomial-time approximate counting and probabilistic reasoning, an interesting question is whether it is *strictly* weaker than APC_1. We provide a positive answer under plausible assumptions, by adapting a technique from [ILW23].

A main technical tool is a KPT witnessing theorem (see [KPT91, Oli25]) for the theory APX_1, where the "student" is implemented by polynomial-size circuits. Formally:

Definition 5.11 (KPT Witnessing with Circuits)**.** Let $\mathcal{T}$ be an extension of PV_1. We say that $\mathcal{T}$ satisfies the *KPT witnessing property with circuits* if the following holds. Let $\varphi(\vec{x}, y, z)$ be any quantifier-free formula in the language of PV_1 such that $\mathcal{T} \vdash \forall \vec{x}\ \exists y\ \forall z\ \varphi(\vec{x}, y, z)$. Then there is a constant $k \in \mathbb{N}$ and functions $f_1, f_2, \ldots, f_k$ (in the standard model) such that the following holds.

For every vector $\vec{x}$ of strings and every $z_1, z_2, \ldots, z_k \in \{0,1\}^*$, it holds in the standard model that:

- either $\varphi(\vec{x}, f_1(\vec{x}), z_1)$ is true;
- or $\varphi(\vec{x}, f_2(\vec{x}, z_1), z_2)$ is true;
- or $\varphi(\vec{x}, f_3(\vec{x}, z_1, z_2), z_3)$ is true;
- $\ldots$;
- or $\varphi(\vec{x}, f_k(\vec{x}, z_1, \ldots, z_{k-1}), z_k)$ is true.

Moreover, over any fixed input length for $\vec{x}$, $f_1, \ldots, f_k$ are computable by *polynomial-size deterministic circuits.*

Theorem 5.12 (KPT Witnessing for APX_1)**.** APX_1 *admits the KPT witnessing property with circuits.*

The theorem can be proved using the standard KPT witnessing theorem (see, e.g., [Oli25, Theorem 3.2]) and the fact that the circuit acceptance probability problem is solvable by (non-uniform) polynomial-size circuits.[24] Put another way, Theorem 5.12 holds as we can hard-wire a sequence of explicit pseudorandom distributions to implement the approximate counting oracle P. Since the argument is standard, we defer the proof of the theorem to Section 5.6.

We will use the following result that is implicit in the proof of [ILW23, Theorem 24]; we refer readers to [ILW23] for precise statements of the assumptions.

Theorem 5.13 (Implicit in [ILW23, Theorem 24]). *Assume the existence of JLS-secure $i\mathcal{O}$ and that coNP is not contained infinitely often in $\mathsf{NP}_{/\mathsf{poly}}$. For any theory $\mathcal{T}$ extending PV_1 that satisfies the KPT witnessing property with circuits, there is a $\forall\Sigma_2^b$ sentence in the language of PV_1 that is provable in APC_1, but is unprovable in $\mathcal{T}$.*

Proof Sketch. [ILW23, Theorem 24] only proves the theorem for a theory called UAPC_1. Nevertheless, by a closer inspection, the only property of the theory used in the proof is that UAPC_1 satisfies the KPT witnessing property with circuits (see [ILW23, Theorem 25]). □

By combining Theorem 5.12 and 5.13, it immediately follows that:

Corollary 5.14. *Assume the existence of JLS-secure $i\mathcal{O}$ and that coNP is not contained infinitely often in $\mathsf{NP}_{/\mathsf{poly}}$. There is a $\forall\Sigma_2^b$ sentence in the language of PV_1 that is provable in APC_1, but is unprovable in APX_1.*

5.3.3 An Open Problem: Further Separations?

Corollary 5.14 shows that APC_1 is likely strictly stronger than APX_1. In other words, under computational assumptions, there are $\forall\Sigma_2^b(\mathsf{PV})$ sentences provable in APC_1 that are not provable in APX_1. An intriguing open problem is whether APC_1 is strictly stronger than APX_1 with respect to $\forall\Sigma_1^b(\mathsf{PV})$ sentences:

Open Problem 6. Is there a $\forall\Sigma_1^b$ sentence in the language of PV_1 that is provable in APC_1, but unprovable in APX_1? In other words, is there an APC_1 provably total TFNP problem (in the language of PV) that is not provably total in APX_1?

As $\mathsf{LossyCode}$ captures the $\forall\Sigma_1^b$-fragment of APC_1 (see Theorem 5.3), and $\mathsf{Refuter}(\mathsf{Yao})$ captures the $\forall\Sigma_1^b$-fragment of APX_1 (see Theorem 1.9), a related question in the theory of pseudorandomness is whether derandomizing $\mathsf{LossyCode}$ is harder than derandomizing $\mathsf{Refuter}(\mathsf{Yao})$. Formally:

Open Problem 7. Is there a deterministic polynomial-time reduction from $\mathsf{LossyCode}$ to $\mathsf{Refuter}(\mathsf{Yao})$? In other words, is there a converse to Theorem 5.4?

We note that these two open problems are technically incomparable. For instance, a positive answer to Open Problem 7 may not give a negative answer to Open Problem 6 if it does not have a feasible correctness proof. Nevertheless, it is conceivable that techniques developed for one of them are likely useful for the other.

5.4 Predictor Extraction Lemma: Proof of Lemma 5.6

Lemma 5.6 (Predictor Extraction Lemma). *Let $t_1(\vec{x}) = t_2(\vec{x})$ be an equation provable in APX. Then there are polynomials $k(n)$, $m(n)$, and a deterministic polynomial-time algorithm E that satisfies the following conditions when n is sufficiently large:*

- *(Input). A string $\vec{x} \in \{0,1\}^n$ and a flat distribution $\mathcal{D} \in (\{0,1\}^k)^m$ of size m over k-bit strings.*

[24] A similar KPT witnessing theorem is proved in [PS21, Theorem 4] (see also [ILW23, Theorem 25]) for the theory PV_1+ "uniform $\mathsf{dWPHP}(\mathsf{PV})$", which might be incomparable with APX_1.

- (*Simulation of Terms*). *Recall that t_1, t_2 are interpreted as polynomial-time* P*-oracle algorithms in standard models. We will simulate the algorithms on input $\vec{x}$ as follows: For every oracle query* $\mathsf{P}(C, \Delta)$*, where $C : \{0,1\}^t \to \{0,1\}$, we will ensure that $t \le k(n)$ and answer the query by*

$$\mathsf{P}(C, \Delta) \triangleq \Pr_{u \leftarrow \mathcal{D}}[C(u_{\le t})]. \tag{5.1}$$

We denote the output of t_1 in the simulation as $t_1^{\mathcal{D}}(\vec{x})$, and the output of t_2 as $t_2^{\mathcal{D}}(\vec{x})$.

- (*Output*). *Suppose that $t_1^{\mathcal{D}}(\vec{x}) \ne t_2^{\mathcal{D}}(\vec{x})$. Then $E(\vec{x}, \mathcal{D})$ outputs $i \in [k(n)]$ and a circuit $P \colon \{0,1\}^{i-1} \to \{0,1\}$ of size at most s such that P predicts the i-th bit of $\mathcal{D}$ with advantage δ such that*

$$(\delta^2/10) \cdot m(n) \ge s + \lceil \log k(n) \rceil + 1.$$

Before proving this lemma, we briefly explain the intuition. Recall that APX is defined as the extension of $\mathsf{PV}(\mathsf{P})$ by additional axioms: Basic Axiom, Boundary Axiom, Local Consistency Axiom, and Precision Consistency Axiom. An APX proof of the equation $t_1(x) = t_2(x)$ is, at a high level, a proof of the following statement: For every interpretation of the oracle P, either $t_1(x) = t_2(x)$, or the oracle P does not satisfy one of the axioms.

For our specific implementation of the oracle P in Lemma 5.6, Basic Axiom, Boundary Axiom and Precision Consistency Axiom are always satisfied, therefore only Local Consistency Axiom can be violated. In such cases, for a circuit $C : \{0,1\}^t \to \{0,1\}$ and strings Δ, B (constructed in the APX proof) such that

$$\left| \Pr_{u \leftarrow \mathcal{D}}[C(u_{\le t})] - \frac{1}{2}\left(\Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}0)] + \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}1)] \right) \right| > 2 \cdot \frac{1}{|\Delta|} + \frac{1}{|B|},$$

following a similar argument as in the proof of Yao's lemma (see, e.g., [AB09, Chapter 9] or Section 4.1), we can construct a predictor from C via a deterministic polynomial-time algorithm.

From a conceptual point of view, the argument crucially explores that a predictor can be constructed not only from the ability to distinguish a distribution from a random string, as in the standard formulation of Yao's lemma, but also from the ability to detect a *local inconsistency* when using the distribution as a random source for approximate counting. This is a perspective that might be of independent interest.

In order to implement this intuition, we prove the lemma using a careful proof-theoretic analysis. Formally, we will prove Lemma 5.6 by induction on the APX proof π of the equation $t_1(x) = t_2(x)$. The functions ℓ, k, m, s, δ and the algorithm E will be determined based on the last rule or axiom of π and the functions and algorithms obtained from the induction hypothesis.

Proof of Lemma 5.6. We will prove a stronger statement: For any provable equation $e(x) : t_1(\vec{x}) = t_2(\vec{x})$, there are non-decreasing polynomials $k_0(n), m_0(n)$ such that the lemma holds for every polynomials $k(n) \ge k_0(n)$ and $m(n) \ge m_0(n)$, when the function $m(n)$ is non-decreasing. We prove this by induction on the length of the proof.

Consider the axiom or rule used in the last line of the proof that concludes $e(x)$.

Basic Axiom. Suppose that e is a provable equation in $\mathsf{PV}(\mathsf{P})$ and it is introduced via the Basic Axiom.[25] Then, for any interpretation of P and any string x, $e(x)$ must be true. We set $k_0(n)$ to be sufficiently large such that $t \le k_0(n)$ for every $C : \{0,1\}^t \to \{0,1\}$ queried in simulations $t_1^{\mathcal{D}}(\vec{x}), t_2^{\mathcal{D}}(\vec{x})$ on $x \in \{0,1\}^n$; this is possible as t_1, t_2 are polynomial-time oracle algorithms. We set other functions and E arbitrarily as $t_1^{\mathcal{D}}(\vec{x}) = t_2^{\mathcal{D}}(x)$ always holds.

Suppose that e is one of the equations encoding $\mathsf{P}(C, \Delta) \in \mathbb{Q}$, $\mathsf{P}(C, \Delta) \le 1$, or $\mathsf{P}(C, \Delta) \ge 0$, where C, Δ are open variables. Let $k_0(n) \triangleq n$ and $m_0(n) = 1$. Similar to the previous case, one can see that for any $k(n) \ge k_0(n)$ and $m(n) \ge m_0(n)$, $e(\vec{x})$ must be true if we interpret P following Equation (5.1). We can set other functions and E arbitrarily as $t_1^{\mathcal{D}}(\vec{x}) = t_2^{\mathcal{D}}(\vec{x})$ always holds.

[25] Note that our proof does not look into the $\mathsf{PV}(\mathsf{P})$ proof of e; it works provided that e is a $\mathsf{PV}(\mathsf{P})$ provable equation.

Boundary Axiom. Suppose that $e(\vec{x})$ is an equation encoding that for any circuit $C \in B_n$, $\mathsf{IsConst}(C) \to \mathsf{P}(C, \Delta) = \mathsf{Bool}(C)$. Note that C and Δ are the only open variables of the equation. We can set $k_0(n) = n$ and $m_0(n) = 1$. For every $k(n) \geq k_0(n)$ and $m(n) \geq m_0(n)$, we can set E arbitrarily.

This is correct as for every $(C, \Delta) \in \{0,1\}^n$ and every distribution $\mathcal{D}$ of size $m(n)$ over $k(n) \geq n$ bits, when we interpret P following Equation (5.1), if $C : \{0,1\}^t \to \{0,1\}$ is a constant circuit, we have $t \leq n \leq k(n)$ and

$$\mathsf{P}(C, \Delta) = \Pr_{u \leftarrow \mathcal{D}}[C(u_{\leq t})] = \mathsf{Bool}(C).$$

In other words, $t_1^{\mathcal{D}}(C, \Delta) = t_2^{\mathcal{D}}(C, \Delta)$ is always true.

Precision Consistency Axiom. This is similar to the case for the Boundary Axiom. Indeed, when the oracle P is interpreted as Equation (5.1), $\mathsf{P}(C, \Delta_1) = \mathsf{P}(C, \Delta_2)$ for *any* Δ_1, Δ_2.

Local Consistency Axiom. In this case, $e(\vec{x})$ is an equation encoding the following sentence: For every circuit $C : \{0,1\}^t \to \{0,1\}$ and strings Δ, B, we have

$$\left| \mathsf{P}(C, \Delta) - \frac{\mathsf{P}(\mathsf{Fix}_0(C), \Delta) + \mathsf{P}(\mathsf{Fix}_1(C), \Delta)}{2} \right| \leq \frac{2}{|\Delta|} + \frac{1}{|B|},$$

where $\mathsf{Fix}_\sigma(C)$ is a PV-term that outputs the circuit obtained from C by fixing the rightmost input bit to $\sigma \in \{0,1\}$. In this equation, C, Δ, B are the only open variables. Let $k_0(n)$ and $m_0(n)$ be polynomials to be determined later.

Let $k(n) \geq k_0(n)$ and $m(n) \geq m_0(n)$. Let $\vec{x} = (C, \Delta, B) \in \{0,1\}^n$ and $\mathcal{D} \in (\{0,1\}^{k(n)})^{m(n)}$. Suppose that $t_1^{\mathcal{D}}(\vec{x}) \neq t_2^{\mathcal{D}}(\vec{x})$. Since the oracle P is interpreted following Equation (5.1), we have

$$\left| \Pr_{u \leftarrow \mathcal{D}}[C(u_{\leq t})] - \frac{\Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}0)] + \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}1)]}{2} \right| > \frac{2}{|\Delta|} + \frac{1}{|B|} \geq \frac{1}{n}.$$

This implies that

$$\begin{aligned} & \frac{1}{2} \left| \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}u_t)] - \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}\overline{u_t})] \right| \\ = & \left| \Pr_{u \leftarrow \mathcal{D}}[C(u_{\leq t})] - \frac{\Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}u_t)] + \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}\overline{u_t})]}{2} \right| \\ = & \left| \Pr_{u \leftarrow \mathcal{D}}[C(u_{\leq t})] - \frac{\Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}0)] + \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}1)]}{2} \right| \geq \frac{1}{n}, \end{aligned} \tag{5.2}$$

where in the second equality we used linearity of expectation and that for every fixed u, $C(u_{<t}u_t) + C(u_{<t}\overline{u_t}) = C(u_{<t}0) + C(u_{<t}1)$.

For simplicity, we only consider the case that

$$\Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}u_t)] - \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}\overline{u_t})] \geq \frac{2}{n}, \tag{5.3}$$

and the other case can be resolved accordingly. We can rewrite the equation above as follows:

$$\begin{aligned} & \Big(\Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}0) \oplus 1 = u_t \wedge u_t = 0] - \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}0) \oplus 1 \neq u_t \wedge u_t = 1] \Big) \\ + & \Big(\Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}1) \oplus 1 \neq u_t \wedge u_t = 1] - \Pr_{u \leftarrow \mathcal{D}}[C(u_{<t}1) \oplus 1 = u_t \wedge u_t = 0] \Big) \geq \frac{2}{n}. \end{aligned}$$

Therefore, one of the terms in the LHS must be at least $1/n$. Again, we will only consider the case that the first term is at least $1/n$, and the other term can be resolved accordingly.

Notice that

$$\begin{aligned}
&\Pr_{u\leftarrow\mathcal{D}}[C(u_{<t}0)\oplus 1 = u_t \wedge u_t = 0] - \Pr_{u\leftarrow\mathcal{D}}[C(u_{<t}0)\oplus 1 \neq u_t \wedge u_t = 1] \\
&= \Pr_{u\leftarrow\mathcal{D}}[C(u_{<t}0)\oplus 1 = u_t \wedge u_t = 0] + \Pr_{u\leftarrow\mathcal{D}}[C(u_{<t}0)\oplus 1 = u_t \wedge u_t = 1] - \Pr_{u\leftarrow\mathcal{D}}[u_t = 1] \\
&= \Pr_{u\leftarrow\mathcal{D}}[C(u_{<t}0)\oplus 1 = u_t] - \Pr_{u\leftarrow\mathcal{D}}[u_t = 1] \geq \frac{1}{n}.
\end{aligned}$$

Subsequently, either $\Pr_{u\leftarrow\mathcal{D}}[u_t = 1] \leq 1/2 - 1/(2n)$ or $\Pr_{u\leftarrow\mathcal{D}}[C(u_{<t}0)\oplus 1 = u_t] \geq 1/2 + 1/(2n)$. In either case – and we can efficiently determine which case holds since $\mathcal{D}$ is explicitly given – we can construct a circuit of size at most $|C| \leq s \triangleq n$ that predicts the t-th bit of $\mathcal{D}$ from the first $(t-1)$ bits with advantage at least $\delta \triangleq 1/(2n)$. By setting $m_0(n) \triangleq n^4$ and $k_0(n) \triangleq n$, we can ensure that

$$(\delta^2/10)\cdot m(n) \geq s + \lceil \log k(n) \rceil + 1$$

when n is sufficiently large.

Logical Rules. We will only consider the substitution rule $t_1 = t_2 \vdash t_1(x/t) = t_2(x/t)$; other logical rules can be resolved accordingly. In this case, $e(\vec{x})$ is of form $t_1(x/t) = t_2(x/t)$, where t is a term and x is one of the open variables of t_1 and t_2, and there is a shorter proof of the premise $t_1 = t_2$. Without loss of generality, we assume that the variable x does not occur in the term t.

Let $\vec{y}$ be the open variables in t but not in t_1, t_2, $\vec{z}$ be the open variables in t_1, t_2 (excluding x) but not in t, and $\vec{w}$ be the open variables in both t and t_1, t_2 (excluding x). By the induction hypothesis, there are polynomials $k'_0(n), m'_0(n)$ such that for every polynomials $k'(n) \geq k'_0(n), m'(n) \geq m'_0(n)$, there are $\delta'(n)$, $s'(n)$, and an algorithm $E'(\vec{x}, \mathcal{D})$ that satisfies the lemma for the equation

$$t_1(x, \vec{z}, \vec{w}) = t_2(x, \vec{z}, \vec{w}). \tag{5.4}$$

Let $\ell(n)$ be an upper bound on the output length of t when the input length is at most n (this is called the *bounding value* of the term, see [Coo75]). We define $k_0(n) \triangleq k'_0(n + \ell(n))$ and $m_0(n) \triangleq m'_0(n + \ell(n))$.

To show that this is correct, fix any $k(n) \geq k_0(n)$ and $m(n) \geq m_0(n)$, and let $s(n), \delta(n)$ be determined later. The algorithm E works as follows. Given any $(\vec{y}, \vec{z}, \vec{w}) \in \{0,1\}^n$ and $\mathcal{D} \in (\{0,1\}^{k(n)})^{m(n)}$ such that

$$t_1^{\mathcal{D}}(t(\vec{y}, \vec{w}), \vec{z}, \vec{w}) \neq t_2^{\mathcal{D}}(t(\vec{y}, \vec{w}), \vec{z}, \vec{w}), \tag{5.5}$$

our goal is to output a predictor of a bit of $\mathcal{D}$ with size $s(n)$ and advantage $\delta(n)$.

The algorithm first computes $x_{\mathcal{D}} \triangleq t^{\mathcal{D}}(\vec{y}, \vec{w})$, which is a string of length at most $\ell(n)$. By Equation (5.5) and the definition of the simulation,

$$t_1^{\mathcal{D}}(x_{\mathcal{D}}, \vec{z}, \vec{w}) \neq t_2^{\mathcal{D}}(x_{\mathcal{D}}, \vec{z}, \vec{w}).$$

Subsequently, by the induction hypothesis (with functions $k'(n + \ell(n)) \geq k(n)$ and $m'(n + \ell(n)) \geq m(n)$), $E'((x_{\mathcal{D}}, \vec{z}, \vec{w}), \mathcal{D})$ outputs (i, P) such that P is a circuit of size s that predicts the i-th bit of $\mathcal{D}$ with advantage δ such that

$$((\delta^2/10)\cdot m'(n + \ell(n)) \geq s + \lceil \log k'(n + \ell(n)) \rceil + 1.$$

It suffices to define $E((\vec{y}, \vec{z}, \vec{w}), \mathcal{D}) \triangleq E'((x_{\mathcal{D}}, \vec{z}, \vec{w}), \mathcal{D})$.

Induction Rule. In this case, e is of form $f_1(x, \vec{y}) = f_2(x, \vec{y})$ for $\mathsf{PV}(\mathsf{P})$ functions f_1, f_2, and there are $\mathsf{PV}(\mathsf{P})$ functions g, h_0, h_1 such that there are shorter proofs of equations

$$f_j(\varepsilon, \vec{y}) = g(\vec{y}) \tag{5.6}$$

$$f_j(s_i(x), \vec{y}) = h_i(x, \vec{y}, f_j(x, \vec{y})) \tag{5.7}$$

for $j \in \{1,2\}$ and $i \in \{0,1\}$. By the induction hypothesis, the lemma holds for each of the 6 equations above.

Let $k_0(n)$ and $m_0(n)$ are polynomials to be determined. For any polynomials $k(n) \geq k_0(n)$ and $m(n) \geq m_0(n)$, we will design an algorithm E that, given $(x, \vec{y}) \in \{0,1\}^n$ and $\mathcal{D} \in (\{0,1\}^{k(n)})^{m(n)}$ satisfying

$$f_1^{\mathcal{D}}(x, \vec{y}) \neq f_2^{\mathcal{D}}(x, \vec{y}),$$

it outputs (i, P) such that P is a circuit of size s that predicts the i-th bit of $\mathcal{D}$ with advantage δ such that $(\delta^2/10) \cdot m(n) \geq s + \lceil \log k(n) \rceil + 1$.

Case 1. Suppose that $f_1^{\mathcal{D}}(\varepsilon, \vec{y}) \neq f_2^{\mathcal{D}}(\varepsilon, \vec{y})$. Then there exists $j \in \{1,2\}$ such that $f_j^{\mathcal{D}}(\varepsilon, \vec{y}) \neq g^{\mathcal{D}}(\vec{y})$. Note that $\vec{y}$ is of length at most n. As Equation (5.6) admits a shorter proof, by the induction hypothesis, there are polynomials $k_0^{(j)}(n'), m_0^{(j)}(n')$ such that when

$$k(n') \geq k_0^{(j)}(n'), \quad m(n') \geq m_0^{(j)}(n'), \tag{5.8}$$

then $E_j(\vec{y}, \mathcal{D})$ outputs a size-s predictor with advantage δ such that $(\delta^2/10) \cdot m(n') \geq s + \lceil \log k(n') \rceil + 1$, where n' denotes the input length of $\vec{y}$. It then suffices to define

$$\begin{aligned} &E((x, \vec{y}), \mathcal{D}) \triangleq E_j(\vec{y}, \mathcal{D}) \\ &k_0(n) \triangleq k_0^{(j)}(n) \in \mathsf{poly}(n), \quad m_0(n) \triangleq m_0^{(j)}(n) \geq m_0^{(j)}(n'). \end{aligned}$$

Case 2. Let $t \triangleq |x|$. The algorithm E first finds the first index $i \leq t$ such that $f_1^{\mathcal{D}}(x_{\leq i}, \vec{y}) = f_2^{\mathcal{D}}(\vec{x}_{\leq i}, \vec{y})$ but $f_1^{\mathcal{D}}(x_{\leq i+1}, \vec{y}) \neq f_2^{\mathcal{D}}(\vec{x}_{\leq i+1}, \vec{y})$; such an index must exist as $f_1^{\mathcal{D}}(\varepsilon, \vec{y}) = f_2^{\mathcal{D}}(\varepsilon, \vec{y})$ and $f_1^{\mathcal{D}}(x, \vec{y}) \neq f_2^{\mathcal{D}}(x, \vec{y})$.

Let $\sigma \triangleq x_{i+1}$. Then there exists $j \in \{1,2\}$ such that

$$f_j^{\mathcal{D}}(s_\sigma(x_{\leq i}), \vec{y}) \neq h_\sigma^{\mathcal{D}}(x_{\leq j}, \vec{y}, f_j^{\mathcal{D}}(x_{\leq i}, \vec{y})).$$

That is, the string $(x_{\leq j}, \vec{y})$ of length at most n violates Equation (5.7) when the approximate counting oracle P is implemented using $\mathcal{D}$ following Equation (5.1). By the induction hypothesis applied to Equation (5.7), there are polynomials $k_0'(n)$ and $m_0'(n)$ such that when

$$k(n) \geq k_0'(n), \quad m(n) \geq m_0'(n), \tag{5.9}$$

there is an algorithm E' such that $E'((x_{\leq i}, \vec{y}), \mathcal{D})$ outputs a size-s predictor with advantage δ such that $(\delta^2/10) \cdot m(n) \geq s + \lceil \log k(n) \rceil + 1$. It then suffices to define $E((x, \vec{y}), \mathcal{D}) \triangleq E'((x_{\leq i}, \vec{y}), \mathcal{D})$.

Wrapping up. Finally, we set $k_0(n)$ and $m_0(n)$ to be sufficiently large polynomials such that when $k(n) \geq k_0(n)$ and $m(n) \geq m_0(n)$, both Equation (5.8) and (5.9) hold. Therefore, in either case, the algorithm E satisfies the requirement of the lemma. ☐

Remark 5.15. By looking into the proof, we note that the polynomials $k_0(n), m_0(n)$ (which define the minimum size of the distribution $\mathcal{D}$) and the running time of E depend on the APX proof; in particular, they may far exceed the running time of the terms t_1, t_2 in the equation e.

For instance, $k_0(n)$ is defined as $k_0'(n + \ell(n))$ in the case for logical rules, where $\ell(n)$ is the output length (i.e. bounding value) of a term t in the proof, and the term t does not necessarily appear in the final equation. At a high level, this is because we need to set the distribution $\mathcal{D}$ to be large enough to accommodate *all oracle queries* in the APX proof; we can then look through the proof and find a violation of the LOCAL CONSISTENCY AXIOM, which produces a predictor.

5.5 Simulating $\mathsf{P}(C,\Delta)$ with $\mathsf{NW}(C,\Delta)$: Proof of Theorem 5.9

We follow the notation in [Jeř07a]. A set X is said to be a *bounded set* defined by a circuit C if $X = \{x < a \mid C(x) = 1\}$. We use $x \in X$ to denote the formula $x < a \wedge C(x) = 1$, and $X \subseteq b$ to denote the formula $\forall x \in X\ x < b$. Note that bounded definable sets are *not* objects in the theory APC_1, but an abbreviation in the meta-theory. For two bounded definable sets $X \subseteq a$ and $Y \subseteq b$, we define

$$X \times Y \triangleq \{bx + a \mid x \in X, y \in Y\} \subseteq ab,$$
$$X \mathbin{\dot\cup} Y \triangleq X \cup \{y + a \mid y \in Y\} \subseteq a + b.$$

We say $C : X \to Y$ if C is a circuit from X to Y, i.e., for every $x \in X$, $C(x) \in Y$. We say $C : X \hookrightarrow Y$ if the circuit $C : X \to Y$ is injective, i.e., for $x_1, x_2 \in X$, $x_1 \neq x_2$, $C(x_1) \neq C(x_2)$. We use $C : X \twoheadrightarrow Y$ to denote that C is onto, i.e., for all $y \in Y$, there exists an $x \in X$ such that $C(x) = y$.

Definition 5.16 (in PV_1)**.** Let $X, Y \subseteq 2^n$ be definable sets, and $\varepsilon \le 1$. We say that X is ε-approximately smaller than Y, denoted by $X \lesssim_\varepsilon Y$, if there exists a circuit G and $v \neq 0$ such that

$$G : v \times (Y \mathbin{\dot\cup} \varepsilon 2^n) \twoheadrightarrow v \times X.$$

Definition 5.17 (in PV_1)**.** We say that X and Y are ε-approximately of equal size, denoted by $X \approx_\varepsilon Y$, if $X \lesssim_\varepsilon Y$ and $Y \lesssim_\varepsilon X$. In particular, we say that X is ε-approximately of size s if $X \approx_\varepsilon s$.

Lemma 5.18 ([Jeř07a, Lemma 2.10])**.** *Let $X, Y, X', Y', Z \subseteq 2^n$ and $W, W' \subseteq 2^n$ be bounded definable sets, and $\varepsilon, \delta \le 1$. The following statements are provable in PV_1.*

(1) If $X \lesssim_\varepsilon Y, \varepsilon \le \delta$, then $X \lesssim_\delta Y$.

(2) If $X \lesssim_0 Y$, then $X \lesssim_\varepsilon Y$.

(3) If $X \lesssim_\varepsilon Y$, $Y \lesssim_\delta Z$, then $X \lesssim_{\delta+\varepsilon} Z$.

(4) If $X \lesssim_\varepsilon X'$, $Y \lesssim_\delta Y'$, and X', Y' are separable by the set W (i.e., $X' \subseteq W$ and $Y' \subseteq 2^n \setminus W$), then $X \cup Y \lesssim_{\varepsilon+\delta} X' \cup Y'$.

(5) If $X \lesssim_\varepsilon X', W \lesssim_\delta W'$, then $X \times W \lesssim_{\varepsilon+\delta+\varepsilon\delta} X' \times W'$.

Lemma 5.19 ([Jeř07a, Lemma 2.11])**.** *Let $X, Y \subseteq 2^n$ be bounded definable sets, $s, t, u \le 2^n$, $\varepsilon, \delta, \eta, \xi \le 1$, and $\xi^{-1} \in \mathsf{Log}$. The following statements are provable in APC_1.*

(1) There exists $s \le 2^n$ such that $X \approx_\xi s$.

(2) $s \lesssim_\varepsilon X \lesssim_\delta t$ implies $s \le t + (\varepsilon + \delta + \xi) \cdot 2^n$.

(3) $X \lesssim_\xi Y$ or $Y \lesssim_\xi X$.

(4) $X \lesssim_\varepsilon Y$ implies $2^n \setminus Y \lesssim_{\varepsilon+\xi} 2^n \setminus X$.

(5) $X \approx_\varepsilon s$, $Y \approx_\delta t$, $X \cap Y \approx_\eta u$ imply $X \cup Y \approx_{\varepsilon+\delta+\eta+\xi} s + t - u$.

Lemma 5.20 (Implicit in [Jeř07a, Lemma 2.14])**.** *Let $\varepsilon < 1/3$. There is a $\mathsf{PV}(\alpha)$ function $\mathsf{Size}(\cdot,\cdot)$ such that the following sentence is provable in $\mathsf{HARD}^{\mathsf{A}}$: For every $n, \delta^{-1} \in \mathsf{Log}$ and set $X \subseteq 2^n$ defined by a circuit $C : \{0,1\}^n \to \{0,1\}$, the following holds:*

- *$0 \le \mathsf{Size}(C, 1^{\delta^{-1}}) \le 2^n$;*
- *$X \approx_\delta \mathsf{Size}(C, 1^{\delta^{-1}})$;*
- *If C is a constant circuit that always outputs 0 (resp. 1), then $\mathsf{Size}(C,\Delta) = 0$ (resp. $\mathsf{Size}(C,\Delta) = 2^n$).*

Proof Sketch. We assume some familiarity with [Jeř07a]. The first two bullets of the lemma hold for a $\mathsf{PV}(\alpha)$ function symbol as the only non-uniformity in [Jeř07a, Theorem 2.7] is the choice of the hard function, which is given by α. The last bullet holds as $\mathsf{Size}(C,\Delta)$ is obtained by computing the acceptance probability of C on a pseudorandom distribution produced via the Nisan-Wigderson PRG, and for a circuit that always accepts (resp. rejects), its acceptance probability on any distribution is always 1 (resp. 0).

We refer interested readers to [ILW23, Appendix D] for a self-contained presentation of the proof of [Jeř07a, Theorem 2.7]. □

Theorem 5.9 (Simulating $\mathsf{P}(C,\Delta)$ with $\mathsf{NW}(C,\Delta)$)**.** *For every $\varepsilon < 1/3$, there is a term $\mathsf{NW}(C,\Delta)$ in the language of $\mathsf{PV}_1(\alpha)$ such that* Basic Axiom*,* Boundary Axiom*,* Local Consistency Axiom*, and* Precision Consistency Axiom *are provable in $\mathsf{HARD}^{\mathsf{A}}$ when the oracle $\mathsf{P}(C,\Delta)$ is replaced by $\mathsf{NW}(C,\Delta)$.*

Proof. Let $\mathsf{Size}(\cdot,\cdot)$ be the function in Lemma 5.20. We define $\mathsf{NW}(C,\Delta) \triangleq \mathsf{Size}(C,1^{|\Delta|})/2^n$, where $C : \{0,1\}^n \to \{0,1\}$ is a circuit. Note that we can encode the rational number $\mathsf{Size}(C,1^{|\Delta|})/2^n$ *precisely* using n binary digits so that there is no rounding issue; recall that in the definition of APX, the output length of P could be as large as $|C|\cdot|\Delta|^2 > n$ (see Section 2.1).

It suffices to verify that the axioms are provable in $\mathsf{HARD}^{\mathsf{A}}$. In the rest of the proof, we argue in $\mathsf{HARD}^{\mathsf{A}}$.

- (Basic Axiom). It follows immediately from the first bullet of Lemma 5.20.
- (Boundary Axiom). It follows immediately from the third bullet of Lemma 5.20.
- (Local Consistency Axiom). Let $C : \{0,1\}^n \to \{0,1\}$ be a circuit, Δ, B be strings with $\delta = 1/|\Delta|$, $\beta = 1/|B|$, $C_0 \triangleq \mathsf{Fix}_0(C), C_1 \triangleq \mathsf{Fix}_1(C)$. We need to prove that

$$\left|\mathsf{NW}(C,\Delta) - \frac{\mathsf{NW}(C_0,\Delta)+\mathsf{NW}(C_1,\Delta)}{2}\right| \le \delta+\beta. \tag{5.10}$$

  Let $X, X_0, X_1 \subseteq \{0,1\}^n$ be the bounded sets defined by C, C_0, C_1, respectively. Let $s \triangleq \mathsf{Size}(C,1^{|\Delta|})$, $s_0 = \mathsf{Size}(C_0,1^{|\Delta|})$, $s_1 \triangleq \mathsf{Size}(C_1,1^{|\Delta|})$. By the second bullet of Lemma 5.20, we know that

$$X \approx_\delta s, \quad X_0 \approx_\delta s_0, \quad X_1 \approx_\delta s_2.$$

  Note that as $X_0 \cap X_1 = \varnothing$, we have that $X_0 \cap X_1 \approx_0 0$. By Lemma 5.19 (5), $X_0 \cup X_1 \approx_{2\delta} s_0 + s_1$. Let $f : v_1(s+\delta\cdot 2^n) \twoheadrightarrow v_1 \times X$ be the witness of $X \lesssim_\delta s$, and $g : v_2 \times ((X_0 \cup X_1) \uplus 2\delta\cdot 2^{n-1}) \twoheadrightarrow v_2(s_0+s_1)$ be the witness of $s_0+s_1 \lesssim_{2\delta} X_0 \cup X_1$. We define a function h

$$h : v_1v_2(s+2\delta\cdot 2^n) \twoheadrightarrow v_1v_2(s_0+s_1)$$

  as follows:

  (*i*) Let $i_1 < v_1, i_2 < v_2, j < s+\delta\cdot 2^n$, the tuple $(i_1,i_2,j) \in v_1v_2(s+2\delta\cdot 2^n)$. We compute $(i_1',x) \triangleq f(i_1,j) \in v_1 \times X$, where $x_{\le n-1} \in X_0 \cup X_1$. The algorithm then computes $(i_2',j') \triangleq g(i_2,x_{\le n-1}) \in v_2(s_0+s_1)$, and outputs $h(i_1,i_2,j) \triangleq (i_1',i_2',j')$.

  (*ii*) Let $i_1 < v_1, i_2 < v_2, s+\delta\cdot 2^n \le j < s+2\delta\cdot 2^n$, the tuple $(i_1,i_2,j) \in v_1v_2(s+2\delta\cdot 2^n)$. Note that $j-(s+\delta\cdot 2^n) \in \delta\cdot 2^n = 2\delta\cdot 2^{n-1}$. The algorithm then computes $(i_2',j') \triangleq g(i_2, j-(s+\delta\cdot 2^n)) \in v_2(s_0+s_1)$, and outputs $h(i_1,i_2,j) \triangleq (i_1,i_2',j')$.

  It can be verified that the function is indeed onto, and thus by definition, $s_0+s_1 \lesssim_{2\delta} s$. Similarly, we can prove that $s \lesssim_{2\delta} s_0+s_1$. Then we have

$$s \lesssim_0 s \lesssim_{2\delta} s_0+s_1 \quad \text{and} \quad s_0+s_1 \lesssim_0 s_0+s_1 \lesssim_{2\delta} s,$$

  by Lemma 5.19 (2), we have that $s \in (s_0+s_1) \pm (2\delta+\beta)\cdot 2^n$. This immediately implies Equation (5.10) as $\mathsf{NW}(C,\Delta) \triangleq \mathsf{Size}(C,1^{|\Delta|})/2^n$.
- (Precision Consistency Axiom). Let $C : \{0,1\}^n \to \{0,1\}$ be a circuit, Δ_1, Δ_2, B be strings and $\delta_1 \triangleq 1/|\Delta_1|, \delta_2 \triangleq 1/|\Delta_2|$, $\beta \triangleq 1/|B|$. We need to prove that

$$|\mathsf{NW}(C,\Delta_1) - \mathsf{NW}(C,\Delta_2)| \le \delta_1+\delta_2+\beta. \tag{5.11}$$

  Let $X \subseteq \{0,1\}^n$ be the bounded set defined by C, and let $s_i \triangleq \mathsf{Size}(C,1^{|\Delta_i|})$, $i \in \{1,2\}$. By the second bullet of Lemma 5.20,

$$X \approx_{\delta_1} s_1, \quad X \approx_{\delta_2} s_2.$$

  Therefore, we have $s_1 \lesssim_{\delta_1} X \lesssim_{\delta_2} s_2$, and by Lemma 5.19 (2), $s_1 \le s_2 + (\delta_1+\delta_2+\beta)\cdot 2^n$. Similarly, we can prove that $s_2 \le s_1 + (\delta_1+\delta_2+\beta)\cdot 2^n$. This immediately implies Equation (5.10) as $\mathsf{NW}(C,\Delta) \triangleq \mathsf{Size}(C,1^{|\Delta|})/2^n$.

This completes the proof. □

5.6 A KPT Witnessing Theorem for APX_1: Proof of Theorem 5.12

Recall the definition of the KPT witnessing property with circuits:

Definition 5.11 (KPT Witnessing with Circuits)**.** Let $\mathcal{T}$ be an extension of PV_1. We say that $\mathcal{T}$ satisfies the *KPT witnessing property with circuits* if the following holds. Let $\varphi(\vec{x}, y, z)$ be any quantifier-free formula in the language of PV_1 such that $\mathcal{T} \vdash \forall \vec{x}\ \exists y\ \forall z\ \varphi(\vec{x}, y, z)$. Then there is a constant $k \in \mathbb{N}$ and functions $f_1, f_2, \ldots, f_k$ (in the standard model) such that the following holds.

For every vector $\vec{x}$ of strings and every $z_1, z_2, \ldots, z_k \in \{0,1\}^*$, it holds in the standard model that:

- either $\varphi(\vec{x}, f_1(\vec{x}), z_1)$ is true;
- or $\varphi(\vec{x}, f_2(\vec{x}, z_1), z_2)$ is true;
- or $\varphi(\vec{x}, f_3(\vec{x}, z_1, z_2), z_3)$ is true;
- $\ldots$;
- or $\varphi(\vec{x}, f_k(\vec{x}, z_1, \ldots, z_{k-1}), z_k)$ is true.

Moreover, over any fixed input length for $\vec{x}$, $f_1, \ldots, f_k$ are computable by *polynomial-size deterministic circuits.*

Theorem 5.12 (KPT Witnessing for APX_1)**.** APX_1 *admits the KPT witnessing property with circuits.*

To prove Theorem 5.12, we will need the standard KPT witnessing theorem for universal first-order theories [KPT91]; interested readers are referred to [Oli25, Theorem 3.2] for detailed discussions.

Theorem 5.21 (KPT witnessing theorem)**.** *Let $\mathcal{T}$ be a universal theory. Let $\varphi(\vec{x}, y, z)$ be a quantifier-free formula in the language of $\mathcal{T}$ such that $\mathcal{T} \vdash \forall \vec{x}\ \exists y\ \forall z\ \varphi(\vec{x}, y, z)$. Then there is a constant $k \in \mathbb{N}$ and terms $t_1, t_2, \ldots, t_k$ (in the language of $\mathcal{T}$) such that the following statement is provable in $\mathcal{T}$:*

For every $\vec{x}$ and every $z_1, z_2, \ldots, z_k$, either $\varphi(\vec{x}, t_1(\vec{x}), z_1)$, or $\varphi(\vec{x}, t_2(\vec{x}, z_1), z_2)$, or $\varphi(\vec{x}, t_3(\vec{x}, z_1, z_2), z_3)$, $\ldots$, or $\varphi(\vec{x}, t_k(\vec{x}, z_1, \ldots, z_{k-1}), z_k)$.

Proof of Theorem 5.12. Recall that APX_1 admits a universal axiomatization (see Proposition 2.8). By Theorem 5.21 with $\mathcal{T} \triangleq \mathsf{APX}_1$, if $\mathsf{APX}_1 \vdash \forall \vec{x}\ \exists y\ \forall z\ \varphi(\vec{x}, y, z)$, there is a constant $k \in \mathbb{N}$ and terms $t_1, \ldots, t_k$ such that APX_1 proves the following sentence:

(Φ): For every $\vec{x}$ and $z_1, z_2, \ldots, z_k$, either $\varphi(\vec{x}, t_1(\vec{x}), z_1)$, or $\varphi(\vec{x}, t_2(\vec{x}, z_1), z_2)$, or $\varphi(\vec{x}, t_3(\vec{x}, z_1, z_2), z_3)$, $\ldots$, or $\varphi(\vec{x}, t_k(\vec{x}, z_1, \ldots, z_{k-1}), z_k)$.

Note that $t_1, \ldots, t_k$ are $\mathsf{PV}(\mathsf{P})$ terms, which can be interpreted as polynomial time P-oracle algorithms.

Similarly to the proof of Theorem 1.9, we can rewrite the universal sentence (Φ) as an equation e_Φ in APX, such that it is provable in APX. Therefore, we know by the soundness of APX (see Proposition 2.7) that e_Φ is true in any standard model $\mathbb{M}(\hat{\mathsf{P}})$.

Let CAPP be the search problem that, given any circuit C and a string Δ, outputs a number $p \in \Pr[C(x)] \pm 1/|\Delta|$. It is well-known that the problem is in prBPP (see, e.g., [Gol11]), and thus can be computable by a family of deterministic polynomial-size circuits. Fix any family of circuits $F(C, \Delta)$ that solves CAPP. By definition, $\mathbb{M}(F)$ is a standard model of APX. Subsequently, e_Φ is true in the model $\mathbb{M}(F)$.

The theorem follows by setting $f_1, f_2, \ldots, f_k$ as $t_1^{\mathbb{M}(F)}, t_2^{\mathbb{M}(F)}, \ldots, t_k^{\mathbb{M}(F)}$. Since for each fixed input length, there is a polynomial upper bound on the size of C and on the length of Δ in the oracle calls to $F(C, \Delta)$ during the computation of $t_1, \ldots, t_k$, this allows us to fix a family of polynomial-size circuits for $f_1, \ldots, f_k$. $\square$

6 Reverse Mathematics of Randomized and Average-Case Lower Bounds

The retraction weak pigeonhole principle for polynomial-time functions is one of the most important combinatorial principles that is known to be provable in APC_1, but unknown to be provable in APX_1. In this section, we explore counting variants of the retraction pigeonhole principle and characterize their equivalence class (with respect to provability in APX_1). We show that this class encompasses certain average-case and randomized communication complexity lower bounds, establishing that these results are all equivalent to appropriate variants of the retraction pigeonhole principle.

6.1 Variants of the Retraction Pigeonhole Principle

We start with the definition of the Retraction Weak Pigeonhole Principle $\mathsf{rWPHP}(\mathsf{PV})$. For simplicity, we introduce the following notation. We use $m : \mathsf{Log} \to \mathsf{Log}$ to denote a PV function symbol $m(n)$ whose input and output are encoded in unary. We use $\varepsilon : \mathsf{Log} \to \mathsf{Log}^{-1}$ to denote a PV function symbol $e(n)$ whose input and output are encoded in unary, and $\varepsilon(n)$ is an abbreviation of the rational number $1/e(n)$.

Definition 6.1 (retraction weak pigeonhole principle)**.** Let $m : \mathsf{Log} \to \mathsf{Log}$. The retraction weak pigeonhole principle $\mathsf{rWPHP}[m](\mathsf{PV})$ with stretch m denotes the following statement in the language of APX_1:

For every $n \in \mathsf{Log}$ and circuits $C : \{0,1\}^n \to \{0,1\}^{m(n)}$, $D : \{0,1\}^{m(n)} \to \{0,1\}^n$, if $m(n) < n$, then there exists a string $x \in \{0,1\}^n$ such that $D(C(x)) \neq x$.

We will define two variants of $\mathsf{rWPHP}(\mathsf{PV})$: an approximate counting version $\#\mathsf{rWPHP}(\mathsf{PV})$, and a randomized compression version $\mathsf{rrWPHP}(\mathsf{PV})$.

Definition 6.2 (approximate counting rWPHP)**.** Let $m : \mathsf{Log} \to \mathsf{Log}, \varepsilon : \mathsf{Log} \to \mathsf{Log}^{-1}$. The approximate counting retraction weak pigeonhole principle $\#\mathsf{rWPHP}[m, \varepsilon](\mathsf{PV})$ with stretch m and error ε denotes the following statement in the language of APX_1:

For every $n \in \mathsf{Log}$ and circuits $C : \{0,1\}^n \to \{0,1\}^{m(n)}$, $D : \{0,1\}^{m(n)} \to \{0,1\}^n$, let T be the circuit such that $T(x) = 1$ if $D(C(x)) \neq x$. Then, if $m(n) < n$ and $\varepsilon(n) < 1 - 2^{n-m(n)}$, there exists $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$ such that $\mathsf{P}_\delta(T) > \varepsilon(n) + \delta + \beta$.

Definition 6.3 (randomized compression rWPHP)**.** Let $m : \mathsf{Log} \to \mathsf{Log}$, $\varepsilon : \mathsf{Log} \times \mathsf{Log} \to \mathsf{Log}^{-1}$. The randomized compression retraction weak pigeonhole principle $\mathsf{rrWPHP}[m, \varepsilon](\mathsf{PV})$ with stretch m and error ε denotes the following statement in the language of APX_1:

For every $n, r \in \mathsf{Log}$ and circuits $C : \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^{m(n)}$, $D : \{0,1\}^{m(n)} \to \{0,1\}^n$, if $m(n) < n$ and $\varepsilon(n + r) < 1 - 2^{n-m(n)}$, then there exists an $x \in \{0,1\}^n$ and $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$ such that the following holds: Let $T_x : \{0,1\}^r \to \{0,1\}$ be the circuit such that $T_x(\mathsf{sd}) = 1$ if and only if $D(C(x, \mathsf{sd})) \neq x$. Then $\mathsf{P}_\delta(T_x) > \varepsilon(n, r) + \delta + \beta$.

It follows immediately from the definition that both variants of rWPHP are true statements in any standard model of APX_1, which is left as an exercise.

Proposition 6.4. *For every $m : \mathsf{Log} \to \mathsf{Log}$, $\varepsilon : \mathsf{Log} \to \mathsf{Log}^{-1}$, $\#\mathsf{rWPHP}[m, \varepsilon](\mathsf{PV})$ and $\mathsf{rrWPHP}[m, \varepsilon](\mathsf{PV})$ are true statements in any standard model $\mathbb{M}(\hat{\mathsf{P}})$ of APX_1.*

These principles can be viewed as the worst-case and (weak) average-case hardness of compression-decompression algorithms. Specifically:

- $\mathsf{rWPHP}[m](\mathsf{PV})$ says that for any deterministic compression-decompression pair (C, D) with compression rate m, there is an incompressible string;
- $\#\mathsf{rWPHP}[m, \varepsilon](\mathsf{PV})$ says that for any deterministic compression-decompression pair (C, D) with compression rate m, there is an ε-fraction of incompressible strings;
- $\mathsf{rrWPHP}_m(\mathsf{PV})$ says that for any (C, D) where C is a *randomized* compression algorithm and D is a *deterministic* decompression algorithm, there must be an input string over which the compression-decompression pair has *error probability* ε.

A classical result in bounded arithmetic is that the retraction weak pigeonhole principle admits a stretch reduction in PV. Concretely:

Theorem 6.5 ([Tha02, Jeř05])**.** *For any constant $\varepsilon \in (0, 1)$, $\mathsf{PV} + \mathsf{rWPHP}[n^\varepsilon](\mathsf{PV}) \vdash \mathsf{rWPHP}[n-1](\mathsf{PV})$.*

6.2 One-Way Communication Lower Bounds

We prove an equivalence result involving $\#\mathsf{rWPHP}(\mathsf{PV})$ and communication complexity lower bounds for *Set Disjointness* against *one-way protocols* with either *public randomness* or *private randomness*.

Formalization of One-Way Communication Protocols. We start with the formalization of a communication protocol. Let $n, m, r \in \mathsf{Log}$. A pair of circuits $g_{\mathsf{A}} : \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^m$ and $d_{\mathsf{B}} : \{0,1\}^n \times \{0,1\}^m \times \{0,1\}^r \to \{0,1\}$ defines a one-way randomized communication protocol as follows:

- (*Public Coin Model*). On any pair of inputs $(x, y) \in \{0,1\}^n \times \{0,1\}^n$ and a uniformly generated public random seed $\mathsf{sd} \in \{0,1\}^r$, Alice sends the message $\mathsf{msg} \triangleq g_{\mathsf{A}}(x, \mathsf{sd})$ to Bob, and Bob decides to accept if and only if $d_{\mathsf{B}}(y, \mathsf{msg}, \mathsf{sd}) = 1$.
- (*Private Coin Model*). On any pair of inputs $(x, y) \in \{0,1\}^n \times \{0,1\}^n$ and uniformly generated private random seeds $\mathsf{sd} \in \{0,1\}^r$, Alice sends the message $\mathsf{msg} \triangleq g_{\mathsf{A}}(x, \mathsf{sd})$ to Bob, and Bob decides to accept if and only if $d_{\mathsf{B}}(y, \mathsf{msg}, 0^r) = 1$.

Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a function specified by a circuit. For every $x, y \in \{0,1\}^n$, let $T^{\mathsf{pub}}_{f,x,y} : \{0,1\}^r \to \{0,1\}$ be the circuit such that $T^{\mathsf{pub}}_{f,x,y}(\mathsf{sd}) = 1$ if and only if the public-coin protocol outputs $1 - f(x, y)$ on the input (x, y) with seed sd, i.e.,

$$T^{\mathsf{pub}}_{f,x,y}(\mathsf{sd}) \triangleq \mathbb{I}[d_{\mathsf{B}}(y, g_{\mathsf{A}}(x, \mathsf{sd}), \mathsf{sd}) \neq f(x, y)] \in \{0,1\}. \tag{6.1}$$

Let $\varepsilon \in \mathbb{Q}$. We say that a public-coin protocol $(g_{\mathsf{A}}, d_{\mathsf{B}})$ computes the function f with error ε if for $x, y \in \{0,1\}^n$, $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$,

$$\mathsf{P}_\delta(T^{\mathsf{pub}}_{f,x,y}) \le \varepsilon + \delta + \beta. \tag{6.2}$$

Note that here we consider the two-sided error setting, while one can also naturally define the correctness in terms of one-sided error.

Accordingly, one may define $T^{\mathsf{priv}}_{f,x,y} : \{0,1\}^r \to \{0,1\}$ to be the circuit such that $T^{\mathsf{pub}}_{f,x,y}(\mathsf{sd}) = 1$ if and only if the private-coin protocol outputs $1 - f(x, y)$ on the input (x, y) with seed sd, i.e.,

$$T^{\mathsf{priv}}_{f,x,y}(\mathsf{sd}) \triangleq \mathbb{I}[d_{\mathsf{B}}(y, g_{\mathsf{A}}(x, \mathsf{sd}), 0^r) \neq f(x, y)] \in \{0,1\}. \tag{6.3}$$

We say that a private-coin protocol $(g_{\mathsf{A}}, d_{\mathsf{B}})$ computes the function f with error ε if for every $x, y \in \{0,1\}^n$, $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$,

$$\mathsf{P}_\delta(T^{\mathsf{priv}}_{f,x,y}) \le \varepsilon + \delta + \beta. \tag{6.4}$$

Communication Complexity Lower Bounds. Fix any function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, $n, m, r \in \mathsf{Log}$, and $\varepsilon \in \mathbb{Q}$. We define the sentence $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^f[n, m, r, \varepsilon]$ as follows: For every *public-coin* protocol $(g_{\mathsf{A}}, d_{\mathsf{B}})$ as defined above, $(g_{\mathsf{A}}, d_{\mathsf{B}})$ fails to compute f with error ε. In other words, there are $x, y \in \{0,1\}^n$ and $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$ such that $\mathsf{P}_\delta(T^{\mathsf{pub}}_{f,x,y}) > \varepsilon + \delta + \beta$.

Accordingly, we define the sentence $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^f[n, m, r, \varepsilon]$ as follows: For every *private-coin* protocol $(g_{\mathsf{A}}, d_{\mathsf{B}})$ as defined above, $(g_{\mathsf{A}}, d_{\mathsf{B}})$ fails to compute f with error ε.

Recall that the Set Disjointness function $\mathsf{SetDisj}(x, y)$ outputs 1 if and only if for every index $i \in [n]$, either $x_i = 0$ or $y_i = 0$, i.e., x and y have no common 1-index. Let $m : \mathsf{Log} \to \mathsf{Log}$, $\varepsilon : \mathsf{Log} \times \mathsf{Log} \to \mathsf{Log}^{-1}$ be functions. We define $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[m, \varepsilon]$ as the following sentence:

For $n, r \in \mathsf{Log}$, $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n, m(n), r, \varepsilon(n, r)]$.

In other words, every public-coin one-way protocol computing $\mathsf{SetDisj}$ with communication complexity $m(n)$ must have error probability at least $\varepsilon(n)$. As we will prove in Section 6.3, the lower bound is correct even for $m(n) = n - n^{\Omega(1)}$. Accordingly, we define $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[m, \varepsilon]$ as the following sentence:

For $n, r \in \mathsf{Log}$, $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n, m(n), r, \varepsilon(n, r)]$.

We also consider a weaker statement that, instead of formalizing the lower bound for a specific function, formalizes the *existence* of a function for which the lower bound holds. Let $m : \mathsf{Log} \to \mathsf{Log}, \varepsilon : \mathsf{Log} \times \mathsf{Log} \to \mathsf{Log}^{-1}$. We define $\mathsf{pub}\text{-}\mathsf{rLB}^{\mathsf{some}}[m, \varepsilon]$ as the following sentence:

For $n, r \in \mathsf{Log}$, there exists a circuit $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ such that $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^f[n, m(n), r, \varepsilon(n, r)]$ holds.

In other words, there exists a function f such that every public-coin one-way protocol computing f with communication complexity $m(n)$ must have a non-negligible error probability. This is implied by $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[m, \varepsilon]$ by fixing f to be $\mathsf{SetDisj}$. Accordingly, we can define $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[m, \varepsilon]$ as the following sentence:

For $n, r \in \mathsf{Log}$, there exists a circuit $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ such that $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^f[n, m(n), r, \varepsilon(n, r)]$ holds.

Upper bound for Equality. As a sanity check, we note that as there is a communication complexity *upper bound* for *Equality* with public randomness using linear hashing (see Theorem 4.5), the corresponding lower bound is unprovable in APX_1.

Theorem 6.6 (Upper Bound for Equality)**.** *There are* PV *functions* $m : \mathsf{Log} \to \mathsf{Log}$*,* $\varepsilon : \mathsf{Log} \to \mathsf{Log}^{-1}$ *satisfying that* $m(n) = \Theta(\log n)$*,* $\varepsilon(n) = 1 - 1/n^{\Theta(1)}$ *such that*

$$\mathsf{APX}_1 \vdash \forall n \ \neg \mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{EQ}}[n, m(n), n \cdot m(n), \varepsilon(n)]. \tag{6.5}$$

In particular, $\mathsf{APX}_1 \vdash \neg \mathsf{pub}\text{-}\mathsf{rLB}[m, \varepsilon]$*.*

Proof Sketch. We argue in APX_1 that Equation (6.5) holds, where m, r, ε will be determined later. As the cases when n is small can be proved in brute-force, it suffices to consider $n > n_0$, where $n_0 \in \mathbb{N}$ is a constant to be determined later.

Fix any $n > n_0$. The one-way communication works as follows. Let $x \in \{0,1\}^n$ be the input for Alice and $y \in \{0,1\}^n$ be that for Bob. They parse the public randomness as a matrix $A \in \{0,1\}^{m(n) \times n}$. Alice sends $Ax \in \{0,1\}^{m(n)}$ as the message, and Bob accepts if and only if $Ax = Ay$. It remains to prove that the protocol works with error at most $\varepsilon(n)$.

Fix any input $x, y \in \{0,1\}^n$. If $x = y$, the protocol always accepts. In other words, the circuit $T^{\mathsf{pub}}_{\mathsf{EQ},x,y}$ in Equation (6.1) is a constant circuit that always rejects. The correctness, i.e. Equation (6.2), follows immediately from the Boundary Axiom. For the case that $x \neq y$, the circuit $T^{\mathsf{pub}}_{\mathsf{EQ},x,y}$ is functionally equivalent to the negation of the circuit $T_{x,y}$ in Theorem 4.5. Therefore, by *Complementary Counting*, the theorem holds as long as we set $m(n) = n^{O(1)}$, $\varepsilon(n) > 1 - 0.51^{m(n)}$, and $n_0 \in \mathbb{N}$ be sufficiently large. This completes the proof. ☐

6.3 The Main Equivalence Result for Communication Complexity

We establish an equivalence between several statements with respect to their provability in APX_1.

Theorem 6.7. *The following statements are equivalent over* APX_1*:*

(1) $\#\mathsf{rWPHP}[n-1, n^{-k}](\mathsf{PV})$*, where* $k \in \mathbb{N}$ *is some constant;*
(2) $\#\mathsf{rWPHP}[n^{\varepsilon}, n^{-k}](\mathsf{PV})$*, where* $\varepsilon \in (0,1)$ *and* $k \in \mathbb{N}$ *are some constants;*
(3) $\mathsf{rrWPHP}[n-1, (n+r)^{-k}](\mathsf{PV})$*, where* $k \in \mathbb{N}$ *is some constant;*
(4) $\mathsf{rrWPHP}[n^{\varepsilon}, (n+r)^{-k}](\mathsf{PV})$*, where* $\varepsilon \in (0,1)$ *and* $k \in \mathbb{N}$ *are some constants;*
(5) $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n-1, (n+r)^{-k}]$*, where* $k \in \mathbb{N}$ *is some constant;*
(6) $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n^{\varepsilon}, (n+r)^{-k}]$*, where* $\varepsilon \in (0,1)$ *and* $k \in \mathbb{N}$ *are some constants;*

(7) $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n-1,(n+r)^{-k}]$, *where* $k \in \mathbb{N}$ *is some constant;*

(8) $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n^{\varepsilon},(n+r)^{-k}]$, *where* $\varepsilon \in (0,1)$ *and* $k \in \mathbb{N}$ *are some constants;*

(9) $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[n-1,(n+r)^{-k}]$, *where* $k \in \mathbb{N}$ *is some constant;*

(10) $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[n^{\varepsilon},(n+r)^{-k}]$, *where* $\varepsilon \in (0,1)$ *and* $k \in \mathbb{N}$ *are some constants;*

(11) $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[n-1,(n+r)^{-k}]$, *where* $k \in \mathbb{N}$ *is some constant;*

(12) $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[n^{\varepsilon},(n+r)^{-k}]$, *where* $\varepsilon \in (0,1)$ *and* $k \in \mathbb{N}$ *are some constants.*

Remark 6.8. In the statements above, the quantification over $k \in \mathbb{N}$ and $\varepsilon \in (0,1)$ takes place outside the theory. For instance, (1) $\Rightarrow$ (2) means that for every $k_1 \in \mathbb{N}$, there exists a $k_2 \in \mathbb{N}$ and $\varepsilon_2 \in (0,1)$ such that the sentence

$$\#\mathsf{rWPHP}[n-1,n^{-k_1}](\mathsf{PV}) \to \#\mathsf{rWPHP}[n^{\varepsilon_2},n^{-k_2}](\mathsf{PV}).$$

is provable in APX_1.

Trivial directions. Both (1) $\Rightarrow$ (2) and (3) $\Rightarrow$ (4) are straightforward. Indeed, a compression-decompression pair with small stretch can be converted into one with larger stretch by padding dummy bits. It is also easy to observe that statements (5) to (12) form a lattice isomorphic to a three-dimensional Boolean cube with respect to implication over APX_1, where (5) is the maximal element (i.e., the strongest lower bound) and (12) is the minimal element (i.e., the weakest lower bound). This is because lower bounds against public-coin protocols imply lower bounds against private-coin protocols; $n-1$ communication lower bounds imply $n^{\Omega(1)}$ communication lower bounds; and lower bounds for $\mathsf{SetDisj}$ imply lower bounds for some function f (by fixing f to be $\mathsf{SetDisj}$).

Non-trivial directions. Observe that, in order to complete the proof of Theorem 6.7, it suffices to establish the following implications: (12) $\Rightarrow$ (4), (1) $\Rightarrow$ (5), (2) $\Rightarrow$ (3), and (4) $\Rightarrow$ (1). The proof of these implications is provided in the subsequent sections.

6.3.1 Compression Implies Communication Upper Bound: (12) $\Rightarrow$ (4)

Lemma 6.9. *For every* $\varepsilon_{12} \in (0,1)$ *and* $k_{12} \in \mathbb{N}$, *there are* $\varepsilon_4 \in (0,1)$ *and* $k_4 \in \mathbb{N}$ *such that*

$$\mathsf{APX}_1 + \mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[n^{\varepsilon_{12}},(n+r)^{-k_{12}}] \vdash \mathsf{rrWPHP}[n^{\varepsilon_4},(n+r)^{-k_4}](\mathsf{PV}).$$

Proof. Fix any constant $\varepsilon_6 \in (0,1)$, $k_{12} \in \mathbb{N}$, let $\varepsilon_2 \triangleq \varepsilon_6$ and $k_4 \triangleq k_{12}$. Let $m_4(n) \triangleq n^{\varepsilon_4}$ and $m_{12}(n) \triangleq n^{\varepsilon_{12}}$. We will prove in APX_1 that $\neg\mathsf{rrWPHP}[n^{\varepsilon_4},n^{-k_4}](\mathsf{PV})$ implies $\neg\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[n^{\varepsilon_{12}},n^{-k_{12}}]$.

Suppose that $\mathsf{rrWPHP}[m_4,n^{-k_4}](\mathsf{PV})$ does not hold. Then there are $n,r \in \mathsf{Log}$ and circuits $C:\{0,1\}^n \times \{0,1\}^r \to \{0,1\}^{m_2(n)}, D:\{0,1\}^{m_2(n)} \to \{0,1\}^n$ such that the following holds. Let T_x be the circuit that $T_x(\mathsf{sd}) = 1$ if $D(C(x,\mathsf{sd})) \neq x$. Then for every $\delta^{-1},\beta^{-1} \in \mathsf{Log}$ and $x \in \{0,1\}^n$,

$$\mathsf{P}_\delta(T_x) \le (n+r)^{-k_4} + \delta + \beta. \tag{6.6}$$

Fix n,r,C,D as described above.

We will now prove that $\neg\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{some}}[n^{\varepsilon_{12}},n^{-k_{12}}]$. In particular, we will prove that for every $f:\{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, $\mathsf{priv}\text{-}\underrightarrow{\mathsf{rLB}}^{f}[n^{\varepsilon_{12}},n^{-k_{12}}]$ does not hold. Fix any circuit $f:\{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. Our goal is to construct a private-coin communication protocol with communication complexity $n^{\varepsilon_{12}}$ that computes f with error $n^{-k_{12}}$. The protocol works as follows:

- Given $x \in \{0,1\}^n$ and uniformly random seed sd, Alice sends the message $g_{\mathsf{A}}(x,\mathsf{sd}) \triangleq C(x,\mathsf{sd})$.
- Given $y \in \{0,1\}^n$ and the message msg, Bob accepts if and only if $d_{\mathsf{B}}(y,\mathsf{msg},\mathsf{sd}) \triangleq f(D(\mathsf{msg}),y) = 1$.

To prove that the protocol computes f with error $n^{-k_{12}}$, fix any $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$. Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later, and $T^{\mathsf{priv}}_{f,x,y}(\mathsf{sd})$ be the circuit as defined in Equation (6.3). It can be verified that for every $x, y \in \{0,1\}^n$ and $\mathsf{sd} \in \{0,1\}^r$, if $T^{\mathsf{priv}}_{f,x,y}(\mathsf{sd}) = 1$, then $T_x(\mathsf{sd}) = 1$. By the *Monotonicity of Approximate Counting*,

$$\mathsf{P}_\eta(T^{\mathsf{priv}}_{f,x,y}) \le \mathsf{P}_\eta(T_x) + 3\eta \le (n+r)^{-k_4} + 5\eta,$$

where the last inequality follows from Equation (6.6). It then follows from the Precision Consistency Axiom that

$$\mathsf{P}_\delta(T^{\mathsf{priv}}_{f,x,y}) \le \delta + \mathsf{P}_\eta(T^{\mathsf{priv}}_{f,x,y}) \le (n+r)^{-k_4} + \delta + 5\eta \le (n+r)^{-k_4} + \delta + \beta$$

by setting $\eta \triangleq \beta/5$. This completes the proof. □

6.3.2 Compression from Communication Upper Bound: (1) ⇒ (5)

Lemma 6.10. *For every constant $k_1 \in \mathbb{N}$, there exists a $k_5 \in \mathbb{N}$ such that* $\mathsf{APX}_1 + \#\mathsf{rWPHP}[n-1, n^{-k_1}](\mathsf{PV}) \vdash \mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n-1, (n+r)^{-k_5}]$.

Proof. Fix any $k_1 \in \mathbb{N}$ and let $k_5 \in \mathbb{N}$ be determined later. We argue in APX_1 that $\neg\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n-1, n^{-k_5}]$ implies $\neg\#\mathsf{rWPHP}[n-1, n^{-k_1}](\mathsf{PV})$.

Suppose that $\mathsf{pub}\text{-}\underrightarrow{\mathsf{rLB}}^{\mathsf{SetDisj}}[n-1, n^{-k_5}]$ does not hold. Then there are $n, r \in \mathsf{Log}$ and a one-way public-coin protocol $g_\mathsf{A} : \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^{n-1}$, $d_\mathsf{B} : \{0,1\}^n \times \{0,1\}^{n-1} \times \{0,1\}^r \to \{0,1\}$ such that the following holds: For every $x, y \in \{0,1\}^n$ and $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, let $T^{\mathsf{pub}}_{\mathsf{SetDisj},x,y}(\mathsf{sd})$ be the circuit defined as Equation (6.1), then

$$\mathsf{P}_\delta(T^{\mathsf{pub}}_{\mathsf{SetDisj},x,y}) \le (n+r)^{-k_5} + \delta + \beta. \tag{6.7}$$

Our goal is to construct a compression-decompression scheme that violates $\#\mathsf{rWPHP}[n-1, n^{-k_1}](\mathsf{PV})$.

Construction of the compression scheme. We construct a pair of circuits $C : \{0,1\}^{n+r} \to \{0,1\}^{n+r-1}$, $D : \{0,1\}^{n+r-1} \to \{0,1\}^{n+r}$ as follows.

- (*Compression*): The circuit C parses the input as $(x, \mathsf{sd}) \in \{0,1\}^n \times \{0,1\}^r$ and computes $\sigma \in \{0,1\}^n$ defined as
$$\sigma_i \triangleq d_\mathsf{B}(e_i, g_\mathsf{A}(x, \mathsf{sd}), \mathsf{sd}) \oplus x_i \oplus 1, \tag{6.8}$$
where e_i denotes the string with the i-th bit being its only 1-index. If $e \neq 0^n$, the compression fails and it outputs 0^n. Otherwise, it outputs the concatenation of $\mathsf{msg} \triangleq g_\mathsf{A}(x, \mathsf{sd})$ and sd.
- (*Decompression*): The circuit D parses the input as the concatenation of msg and sd as mentioned above, computes $y \in \{0,1\}^n$ as
$$y_i \triangleq d_\mathsf{B}(e_i, \mathsf{msg}, \mathsf{sd}) \oplus 1,$$
and outputs the concatenation of y and sd.

It is clear that when $\sigma = 0^n$, the compression-decompression scheme is correct.

Analysis of the error probability. We will prove that (C, D) is a compression-decompression scheme that violates $\#\mathsf{rWPHP}[n-1, n^{-k_1}](\mathsf{PV})$. Fix any $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and let $T : \{0,1\}^{n+r} \to \{0,1\}$ be the circuit that $T(z)$ outputs 1 if $D(C(z)) \neq z$. Our goal is to prove that $\mathsf{P}_\delta(T) \le n^{-k_1} + \delta + \beta$.

Let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later, and $T' : \{0,1\}^{n+r} \to \{0,1\}$ be the following circuit: Given $(x, \mathsf{sd}) \in \{0,1\}^n \times \{0,1\}^r$, it computes σ via Equation (6.8), and outputs 1 if and only if $\sigma \neq 0^n$. As mentioned above, for every x and sd, $T(x \circ \mathsf{sd}) = 1$ implies that $T'(x \circ \mathsf{sd}) = 1$. Therefore, by the *Monotonicity of Approximate Counting*, we have

$$\mathsf{P}_\eta(T) \le \mathsf{P}_\eta(T') + 3\eta. \tag{6.9}$$

Let $V \triangleq \{0,1\}$ and $X_1, X_2, \ldots, X_n$ be the random variables supported over $\{0,1\}^{n+r}$ such that $X_i = 1$ if and only if $e_i \neq 0$. Let $F_i(x, \mathsf{sd})$ be the circuit that defines X_i for every $i \in [n]$. It is clear that $T'(x, \mathsf{sd})$ is the circuit that outputs 1 if and only if $X_i = 1$ for some $i \in [n]$. Let Y be the random variable defined by $(V, n+r, T')$. By the *Union Bound*, we have

$$\mathsf{P}_\eta(T') \leq \mathbb{E}_\eta[Y] + 3\eta \leq \mathbb{E}_\eta[X_1] + \cdots + \mathbb{E}_\eta[X_n] + 3\eta \cdot (n+1), \tag{6.10}$$

where the first inequality follows from Proposition 3.12.

In addition, for every $i \in [n]$ and every $x \in \{0,1\}^n$, we can see that $F_i(x, \cdot)$ is functionally equivalent to $T^{\mathsf{pub}}_{\mathsf{SetDisj},x,e_i}(\cdot)$. Let $X_i|_x$ be the random variable obtained by fixing the first part of the seed to be x. Then for $x \in \{0,1\}^n$,

$$\mathbb{E}_\eta[X_i|_x] \leq \mathsf{P}_\eta(F_i(x,\cdot)) + 3\eta \leq \mathsf{P}_\eta(T^{\mathsf{pub}}_{\mathsf{SetDisj},x,e_i}) + 6\eta \leq (n+r)^{-k_5} + 8\eta,$$

where the last inequality follows from Equation (6.7). By *Averaging Argument for Expectation*, we have

$$\mathbb{E}_\eta[X_i] \leq (n+r)^{-k_5} + 8\eta + 3\eta \leq n^{-k_5} + 11\eta. \tag{6.11}$$

Combining the results above, we have:

$$\begin{aligned}
\mathsf{P}_\delta(T) &\leq \mathsf{P}_\eta(T) + \delta + 2\eta && (\textsc{Precision Consistency Axiom}) \\
&\leq \mathsf{P}_\eta(T') + \delta + 5\eta && (\text{Equation } (6.9)) \\
&\leq \mathbb{E}_\eta[X_1] + \cdots + \mathbb{E}_\eta[X_n] + \delta + 3\eta \cdot (n+1) + 5\eta && (\text{Equation } (6.10)) \\
&\leq \delta + ((n+r)^{-k_5} + 11\eta) \cdot n + 3\eta \cdot (n+1) + 5\eta && (\text{Equation } (6.11)) \\
&\leq (n+r)^{-k_1} + \delta + \beta,
\end{aligned}$$

where the last inequality follows by setting $k_5 \triangleq k_1 + 1$ and $\eta \triangleq \beta/(50n)$. This violates $\#\mathsf{rWPHP}[n-1, n^{-k_1}](\mathsf{PV})$ and thus completes the proof. ☐

6.3.3 Stretch Reduction for Compression: (2) ⇒ (3)

Lemma 6.11. *For any $\varepsilon_2 \in (0,1)$ and $k_2 \in \mathbb{N}$, there exists $k_3 \in \mathbb{N}$ such that $\mathsf{APX}_1 + \#\mathsf{rWPHP}[n^{\varepsilon_2}, n^{-k_2}](\mathsf{PV}) \vdash \mathsf{rrWPHP}[n-1, (n+r)^{-k_3}](\mathsf{PV})$.*

Proof. Fix any constant $\varepsilon_2 \in (0,1)$ and $k_2 \in \mathbb{N}$, and let $k_3 \in \mathbb{N}$ be determined later. We argue in APX_1 that if $\mathsf{rrWPHP}[n-1, (n+r)^{-k_3}](\mathsf{PV})$ does not hold, then $\#\mathsf{rWPHP}[n^{\varepsilon_2}, n^{-k_2}](\mathsf{PV})$ does not hold.

Suppose that $\mathsf{rrWPHP}[n-1, n^{-k_3}](\mathsf{PV})$ does not hold. Then there are $n, r \in \mathsf{Log}$ and circuits $C : \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^{n-1}$, $D : \{0,1\}^{n-1} \to \{0,1\}^n$ such that for every $x \in \{0,1\}^n$ and every $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, let $T_x : \{0,1\}^r \to \{0,1\}$ be the circuit such that $T_x(\mathsf{sd}) = 1$ if and only if $D(C(x, \mathsf{sd})) \neq x$, then

$$\mathsf{P}_\delta(T_x) \leq (n+r)^{-k_3} + \delta + \beta. \tag{6.12}$$

In other words, there is a one-bit randomized compression scheme that is worst-case correct with error n^{-k_3}. Our goal is to construct a deterministic and average-case compression-decompression algorithm that violates $\#\mathsf{rWPHP}[n^{\varepsilon_2}, n^{-k_2}](\mathsf{PV})$.

Compression and decompression circuits. Let $\ell \in \mathsf{Log}$ and $d \in \mathsf{LogLog}$ be parameters to be determined later. The compression circuit takes an $(\ell + dr)$-bit string as input, parses it as $z \in \{0,1\}^\ell$ and $(\mathsf{sd}_1, \ldots, \mathsf{sd}_d) \in \{0,1\}^r$, and runs a d-round iterative compression algorithm.

Initialize $z_0 \leftarrow z$. In the i-th round, the iterative algorithm works as follows:

1. Parse z_{i-1} as $x_1 \circ x_2 \circ \cdots \circ x_k \circ y_i$, where $k \triangleq \lfloor |z_{i-1}|/n \rfloor$ and $x_1, \ldots, x_k \in \{0,1\}^n$.
2. For every $j \in [k]$, compute $x'_j \triangleq C(x_j, \mathsf{sd}_i)$.

3. Set $z_i \leftarrow x'_1 \circ x'_2 \circ \cdots \circ x'_k$.

Finally, the compression circuit outputs the encoding of the tuple $(z_d, y_1, \ldots, y_d, \mathsf{sd}_1, \ldots, \mathsf{sd}_d)$.

The decompression circuit takes $(z_d, y_1, \ldots, y_d, \mathsf{sd}_1, \ldots, \mathsf{sd}_d)$ and works reversely via a d-round iterative algorithm. In the i-th iteration, the algorithm works as follows:

1. Parse z_{d+1-i} as $x'_1 \circ x'_2 \circ \cdots \circ x'_k$, where $k \triangleq \lfloor |z_{d+1-i}|/(n-1) \rfloor$ and $x'_1, \ldots, x'_k \in \{0,1\}^{n-1}$.
2. For every $j \in [k]$, compute $x_j \triangleq D(x'_j)$.
3. Set $z_{d-i} \leftarrow x_1 \circ x_2 \circ \cdots \circ x_k \circ y_{d+1-i}$.

We now set the parameters ℓ and d such that the compression scheme above has stretch at least $(\ell+dr)^{\varepsilon_2}$. Let $z_0, z_1, \ldots, z_d, y_1, \ldots, y_d$ be the strings obtained by the compression algorithm, it is clear that

$$|z_i| \le |z_{i-1}| \cdot \left(1 - \frac{1}{n}\right), |y_i| \le n,$$

and thus the output length of the compression circuit is at most

$$\ell \cdot \left(1 - \frac{1}{n}\right)^d + d \cdot (n + r).$$

We can set $\ell = (n+r)^{10/\varepsilon_2}$ and $d = 10 \cdot n \log \ell$ such that the output length is at most $O(n(n+r)\log \ell) \ll \ell^{\varepsilon_2} \le (\ell + dr)^{\varepsilon_2}$. Therefore, the compression stretch is $(\ell+dr)^{\varepsilon_2}$ for sufficiently large n and r; the cases when n, r are small can be proved by a brute-force case study.

Analysis of the error probability. Fix $\ell \in \mathsf{Log}$ and $d \in \mathsf{LogLog}$ as above. Let $C' : \{0,1\}^{\ell+dr} \to \{0,1\}^{(\ell+dr)^{\varepsilon_2}}$, $D' : \{0,1\}^{(\ell+dr)^{\varepsilon_2}} \to \{0,1\}^{\ell+dr}$ be the compression and decompression algorithms mentioned above. Let $T : \{0,1\}^{\ell+dr}$ be the circuit that $T(z) = 1$ if and only if $D'(C'(z)) \ne z$, i.e., the compression scheme fails. Our goal is to prove that for every $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $\mathsf{P}_\delta(T) \le (\ell+dr)^{-k_2} + \delta + \beta$.

Fix any $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Let $V \triangleq \{0,1\}$. For every $i \in [d]$ and $j \le \lfloor \ell/n \rfloor$, we define $F_{ij}(z, \mathsf{sd}_1, \ldots, \mathsf{sd}_d)$ be the circuit that outputs 1 if and only if the following holds:

- In the i-th round of the compression algorithm, let $k \triangleq \lfloor |z_{i-1}|/n \rfloor$, then $j \le k$ and $D(C(x_j, \mathsf{sd}_i)) \ne x_j$.

Let X_{ij} be the random variable defined by $(V, \ell+dr, F_{ij})$. Let $F(z, \mathsf{sd}_1, \ldots, \mathsf{sd}_d)$ be the circuit that outputs 1 if and only if $F_{ij}(z, \mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 1$ for some $i \in [d]$ and $j \le \lfloor \ell/n \rfloor$, and Y be the random variable defined by $(V, \ell+dr, F_{ij})$. By the *Union Bound*, we have

$$\mathsf{P}_\eta(F) \le \mathbb{E}_\eta[Y] + 3\eta \le \sum_{ij} \mathbb{E}_\eta[X_{ij}] + 3\eta \cdot (d\ell + 1), \tag{6.13}$$

where the first inequality follows from Proposition 3.12.

It is clear that PV proves that for every $z \in \{0,1\}^\ell, \mathsf{sd}_1, \ldots, \mathsf{sd}_d \in \{0,1\}^r$, if $T(z, \mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 1$, then $F(z, \mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 1$. To see this, notice that if $F(z, \mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 0$, we can prove by induction on i that if we run the iterative compression algorithm on the input $z \circ \mathsf{sd}_1 \circ \cdots \circ \mathsf{sd}_d$ for i rounds, and run the iterative decompression algorithm starting from the $d - i$ round, it will be correctly decompressed. This can be implemented by induction on a PV term, which is available in PV. Subsequently, by the *Monotonicity of Approximate Counting*,

$$\mathsf{P}_\eta(T) \le \mathsf{P}_\eta(F) + 3\eta. \tag{6.14}$$

Next, we prove an upper bound on $\mathbb{E}_\eta[X_{ij}]$. Fix any $i \in [d]$ and $j \le \lfloor \ell/n \rfloor$. Let

$$\rho = (z, \mathsf{sd}_1, \ldots, \mathsf{sd}_{i-1}, \mathsf{sd}_{i+1}, \ldots, \mathsf{sd}_d)$$

be an arbitrary assignment to all but the interval sd_i in the seed of X_{ij}. Let x_j be the string in the i-th round of the compression algorithm on the input z and using $\mathsf{sd}_1, \ldots, \mathsf{sd}_{i-1}$ in the first $i-1$ rounds. Note that x_j can be computed by a PV term given ρ. Recall that $T_{x_j}(\mathsf{sd})$ is the circuit that outputs 1 if and only if $D(C(x_j, \mathsf{sd})) \neq x_j$. It can be proved that $T_{x_j}(\mathsf{sd}_i) = 1$ if and only if $F_{ij}(\rho \cup \mathsf{sd}_i) = 1$, i.e., $X_{ij}|_\rho$ is the indicator variable of $T_{x_j}(\mathsf{sd}) = 1$. Subsequently,

$$\mathbb{E}_\eta[X_{ij}|_\rho] \leq \mathsf{P}_\eta(T_{x_j}) + 6\eta \leq (n+r)^{-k_3} + 8\eta, \tag{6.15}$$

where the first inequality follows from Proposition 3.12 and *Global Consistency*, and the second inequality follows from Equation (6.12).

Note that Equation (6.15) holds for any assignment ρ. By *Averaging Argument for Expectation*, we can further deduce that

$$\mathbb{E}_\eta[X_{ij}] \leq n^{-k_3} + 8\eta + 3\eta \leq n^{-k_3} + 11\eta. \tag{6.16}$$

Combining the results above, we can now calculate

$$\begin{aligned}
\mathsf{P}_\delta(T) &\leq \mathsf{P}_\eta(T) + \delta + 2\eta && \text{(Precision Consistency Axiom)}\\
&\leq \mathsf{P}_\eta(F) + \delta + 5\eta && \text{(Equation (6.14))}\\
&\leq \sum_{ij} \mathbb{E}_\eta[X_{ij}] + 3\eta \cdot (d\ell + 1) + \delta + 5\eta && \text{(Equation (6.13))}\\
&\leq ((n+r)^{-k_3} + 11\eta) \cdot d \cdot \ell + 3\eta \cdot (d\ell + 1) + \delta + 5\eta && \text{(Equation (6.16))}\\
&\leq (\ell + dr)^{-k_2} + \delta + \beta,
\end{aligned}$$

where the last inequality follows by setting $\eta \triangleq \beta/(100(d\ell+1))$ and $k_3 \triangleq 100k_2/\varepsilon_2 + 10k_2 + 10$. This shows that the pair of circuits C', D' violates $\#\mathsf{rWPHP}[n^{\varepsilon_2}, n^{-k_2}](\mathsf{PV})$ and thus completes the proof. □

6.3.4 Worst-Case to Average-Case Reduction: (4) ⇒ (1)

Lemma 6.12. *For any $\varepsilon_4 \in (0,1)$ and $k_4 \in \mathbb{N}$, there exists $k_1 \in \mathbb{N}$ such that $\mathsf{APX}_1 + \mathsf{rrWPHP}[n^{\varepsilon_4}, (n+r)^{-k_4}](\mathsf{PV}) \vdash \#\mathsf{rWPHP}[n-1, n^{-k_1}](\mathsf{PV})$.*

We will use the iterative compression algorithm in Lemma 6.11 to boost the stretch to m_2, while a new trick is required to construct worst-case compression from average-case compression algorithm. At a high level, we observe that the compression-decompression problem with large stretch admits *random self-reducibility* that is provably correct via the *Re-randomization Lemma*.

Proof of Lemma 6.12. Fix any constant $\varepsilon_4 \in (0,1)$, $k_4 \in \mathbb{N}$, and let $k_1 \in \mathbb{N}$ be determined later. We argue in APX_1 that assuming $\#\mathsf{rWPHP}[n-1, n^{-k_4}](\mathsf{PV})$ does not hold, $\mathsf{rrWPHP}[n^{\varepsilon_4}, (n+r)^{-k_4}](\mathsf{PV})$ also does not hold. In other words, we will construct a polynomial-stretch randomized worst-case compression scheme from a one-bit deterministic average-case compression scheme.

Assume for contradiction that $\#\mathsf{rWPHP}[n-1, n^{-k_1}](\mathsf{PV})$ does not hold. Then there is an $n \in \mathsf{Log}$ and circuits $C : \{0,1\}^n \to \{0,1\}^{n-1}$, $D : \{0,1\}^{n-1} \to \{0,1\}^n$ such that the following holds. Let T be the circuit that $T(x) = 1$ if $D(C(x)) \neq x$. Then for every $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$, $\mathsf{P}_\delta(T) \leq n^{-k_1} + \delta + \beta$. By the *Re-randomization Lemma*, we know that for every $x \in \{0,1\}^n$, let $T_x^\oplus$ be the circuit $T_x^\oplus(\mathsf{sd}) \triangleq T(x \oplus \mathsf{sd})$, then

$$\mathsf{P}_\delta(T_x^\oplus) \leq \mathsf{P}_\delta(T) + 2\delta + \beta \leq n^{-k_1} + 3\delta + 2\beta. \tag{6.17}$$

Note that we can assume without loss of generality that n is larger than any fixed standard integer $n_0 \in \mathbb{N}$, as the cases when $n \leq n_0$ can be resolved in brute force.

Compression and decompression circuits. Let $\ell \in \mathsf{Log}$ and $d \in \mathsf{LogLog}$ be parameters to be determined later. The compression circuit takes an ℓ-bit string as input z, an nd-bit random seed $(\mathsf{sd}_1, \ldots, \mathsf{sd}_d) \in \{0,1\}^r$, and performs the following d-round iterative algorithm. It initializes $z_0 \leftarrow z$. In the i-th round, the algorithm works as follows:

1. Parse z_{i-1} as $x_1 \circ x_2 \circ \cdots \circ x_k \circ y_i$, where $k \triangleq \lfloor |z_{i-1}|/n \rfloor$ and $x_1, \ldots, x_k \in \{0,1\}^n$;
2. For every $j \in [k]$, compute $x'_j \triangleq C(x_j \oplus \mathsf{sd}_i)$.
3. Set $z_i \leftarrow x'_1 \oplus x'_2 \oplus \cdots \oplus x'_k$.

Finally, the compression circuit outputs the encoding of the tuple $(z_d, y_1, \ldots, y_d, \mathsf{sd}_1, \ldots, \mathsf{sd}_d)$.

The decompression circuit takes $(z_d, y_1, \ldots, y_d, \mathsf{sd}_1, \ldots, \mathsf{sd}_d)$ and works reversely via a d-round iterative algorithm. In the i-th iteration, the algorithm works as follows:

1. Parse z_{d+1-i} as $x'_1 \circ x'_2 \circ \cdots \circ x'_k$, where $k \triangleq \lfloor |z_{d+1-i}|/(n-1) \rfloor$ and $x'_1, \ldots, x'_k \in \{0,1\}^{n-1}$.
2. For every $j \in [k]$, compute $x_j \triangleq D(x'_j) \oplus \mathsf{sd}_i$.
3. Set $z_{d-i} \leftarrow x_1 \circ x_2 \circ \cdots \circ x_k \circ y_{d+1-i}$.

Similar to the proof of Lemma 6.11, we can set the parameters $\ell \triangleq n^{10/\varepsilon_4}$ and $d \triangleq 10 \cdot n \log \ell$ such that the compression scheme above has stretch at least ℓ^{ε_4}. The length of random string of the compression scheme is $r \triangleq dn$.

Analysis of the error probability. Fix $\ell \in \mathsf{Log}$ and $d \in \mathsf{LogLog}$ as above. Let $C' : \{0,1\}^\ell \times \{0,1\}^{dn} \to \{0,1\}^{\ell^{\varepsilon_4}}$, $D' : \{0,1\}^{\ell^{\varepsilon_4}} \to \{0,1\}^\ell$ be the compression and decompression algorithms mentioned above. Let $T_z : \{0,1\}^{dn} \to \{0,1\}$ be the circuit that parses the input as $\mathsf{sd} \triangleq (\mathsf{sd}_1, \ldots, \mathsf{sd}_d) \in \{0,1\}^{dn}$ and outputs 1 if and only if $D'(C'(z, \mathsf{sd})) \neq z$, i.e., the compression scheme fails on the input z. Our goal is to prove that for every $\delta^{-1}, \beta^{-1} \in \mathsf{Log}$ and $z \in \{0,1\}^\ell$, $\mathsf{P}_\delta(T_z) \leq (\ell + dn)^{-k_4} + \delta + \beta$.

Fix any $\delta^{-1}, \beta^{-1} \in \mathsf{Log}, z \in \{0,1\}^\ell$ and let $\eta^{-1} \in \mathsf{Log}$ be a parameter to be determined later. Let $V \triangleq \{0,1\}$. For every $i \in [d]$ and $j \leq \lfloor \ell/n \rfloor$, we define $F_{ij}(\mathsf{sd}_1, \ldots, \mathsf{sd}_d)$ be the circuit that outputs 1 if and only if the following holds:

- In the i-th round of the compression algorithm, let $k \triangleq \lfloor |z_{i-1}|/n \rfloor$, then $j \leq k$ and $D(C(x_j \oplus \mathsf{sd}_i)) \neq x_j \oplus \mathsf{sd}_i$.

Let X_{ij} be the random variable defined by (V, dn, F_{ij}). Let $F(\mathsf{sd}_1, \ldots, \mathsf{sd}_d)$ be the circuit that outputs 1 if and only if $F_{ij}(\mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 1$ for some $i \in [d]$ and $j \leq \lfloor \ell/n \rfloor$, and Y be the random variable defined by (V, dn, F_{ij}). By the *Union Bound*, we have

$$\mathsf{P}_\eta(F) \leq \mathbb{E}_\eta[Y] + 3\eta \leq \sum_{ij} \mathbb{E}_\eta[X_{ij}] + 3\eta \cdot (d\ell + 1), \tag{6.18}$$

where the first inequality follows from Proposition 3.12.

It is clear that PV proves that for every $\mathsf{sd}_1, \ldots, \mathsf{sd}_d \in \{0,1\}^r$, if $T_z(\mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 1$, it follows that $F(\mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 1$. To see this, notice that if $F(\mathsf{sd}_1, \ldots, \mathsf{sd}_d) = 0$, we can prove by induction on i that if we run the iterative compression algorithm on the input z for i rounds, and run the iterative decompression algorithm starting from the $d - i$ round, it will be correctly decompressed. This can be implemented by induction on a PV term, which is available in PV. Subsequently, by the *Monotonicity of Approximate Counting*,

$$\mathsf{P}_\eta(T_z) \leq \mathsf{P}_\eta(F) + 3\eta. \tag{6.19}$$

Next, we prove an upper bound on $\mathbb{E}_\eta[X_{ij}]$. Fix any $i \in [d]$ and $j \leq \lfloor \ell/n \rfloor$. Let

$$\rho = (\mathsf{sd}_1, \ldots, \mathsf{sd}_{i-1}, \mathsf{sd}_{i+1}, \ldots, \mathsf{sd}_d)$$

be an arbitrary assignment to all but the interval sd_i in the seed of X_{ij}. Let x_j be the string in the i-th round of the compression algorithm on the input z and using $\mathsf{sd}_1, \ldots, \mathsf{sd}_{i-1}$ in the first $i-1$ rounds. Note that x_j can be computed by a PV term given ρ. Recall that $T_x^\oplus(\mathsf{sd})$ is the circuit that outputs 1 if and only if $D(C(x \oplus \mathsf{sd})) \neq x \oplus \mathsf{sd}$. It can be proved that $T_{x_j}^\oplus(\mathsf{sd}_i) = 1$ if and only if $F_{ij}(\rho \cup \mathsf{sd}_i) = 1$, i.e., $X_{ij}|_\rho$ is the indicator variable of $T_{x_j}^\oplus(\mathsf{sd}) = 1$. Subsequently,

$$\mathbb{E}_\eta[X_{ij}|_\rho] \leq \mathsf{P}_\eta(T_{x_j}^\oplus) + 6\eta \leq n^{-k_1} + 11\eta, \tag{6.20}$$

where the first inequality follows from Proposition 3.12 and *Global Consistency*, and the second inequality follows from Equation (6.17).

Note that Equation (6.20) holds for any assignment ρ. By *Averaging Argument for Expectation*, we can further deduce that

$$\mathbb{E}_\eta[X_{ij}] \le n^{-k_1} + 11\eta + 3\eta \le n^{-k_1} + 14\eta. \tag{6.21}$$

Combining the results above, we can now calculate

$$\begin{aligned}
\mathsf{P}_\delta(T) &\le \mathsf{P}_\eta(T) + \delta + 2\eta && \text{(Precision Consistency Axiom)}\\
&\le \mathsf{P}_\eta(F) + \delta + 5\eta && \text{(Equation (6.19))}\\
&\le \sum_{ij} \mathbb{E}_\eta[X_{ij}] + 3\eta \cdot (d\ell + 1) + \delta + 5\eta && \text{(Equation (6.18))}\\
&\le \left(n^{-k_1} + 14\eta\right) \cdot d \cdot \ell + 3\eta \cdot (d\ell + 1) + \delta + 5\eta && \text{(Equation (6.21))}\\
&\le n^{-k_1} \cdot d\ell + \delta + \beta,
\end{aligned}$$

where the last inequality follows by setting $\eta \triangleq \beta/(100(d\ell + 1))$. Recall that $\ell = n^{10/\varepsilon_4}$ and $d = 10 \cdot n \log \ell$, we have

$$n^{-k_1} \cdot d\ell = n^{-k_1} \cdot n^{10/\varepsilon_4} \cdot \frac{100n}{\varepsilon_4} \cdot \log n \le n^{-k_4} \le (\ell + dn)^{-k_4}.$$

by setting $k_1 \triangleq 100/\varepsilon_4 + 10k_4 + 10$ when n is sufficiently large. This shows that C', D' violates $\mathsf{rrWPHP}_{m_2}(\mathsf{PV})$ and thus completes the proof. ☐

References

[AAdRK25] Noel Arteche, Albert Atserias, Susanna F. de Rezende, and Erfan Khaniki. The proof analysis problem. In *Symposium on Theory of Computing* (STOC), 2025.

[AAI+01] Manindra Agrawal, Eric Allender, Russell Impagliazzo, Toniann Pitassi, and Steven Rudich. Reducing the complexity of reductions. *Comput. Complex.*, 10(2):117–138, 2001.

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach.* Cambridge University Press, 2009.

[ABM23] Albert Atserias, Sam Buss, and Moritz Müller. On the consistency of circuit lower bounds for non-deterministic time. In *Symposium on Theory of Computing* (STOC), pages 1257–1270, 2023.

[Agr01] Manindra Agrawal. Towards uniform AC^0-isomorphisms. In *Conference on Computational Complexity* (CCC), pages 13–20, 2001.

[AS16] Noga Alon and Joel H. Spencer. *The probabilistic method.* John Wiley & Sons, 2016.

[AT25] Albert Atserias and Iddo Tzameret. Feasibly constructive proof of Schwartz-Zippel lemma and the complexity of finding hitting sets. In *Symposium on Theory of Computing* (STOC), pages 1096–1107, 2025.

[AvM11] Scott Aaronson and Dieter van Melkebeek. On circuit lower bounds from derandomization. *Theory Comput.*, 7(1):177–184, 2011.

[AW89] Miklós Ajtai and Avi Wigderson. Deterministic simulation of probabilistic constant depth circuits. *Adv. Comput. Res.*, 5:199–222, 1989.

[BCH+96] Mihir Bellare, Don Coppersmith, Johan Håstad, Marcos A. Kiwi, and Madhu Sudan. Linearity testing in characteristic two. *IEEE Trans. Inf. Theory*, 42(6):1781–1795, 1996.

[BKKK20] Sam Buss, Valentine Kabanets, Antonina Kolokolova, and Michal Koucký. Expander construction in VNC^1. *Annals of Pure and Applied Logic*, 171(7):102796, 2020.

[BKT14] Samuel R. Buss, Leszek A. Kołodziejczyk, and Neil Thapen. Fragments of approximate counting. *Journal of Symbolic Logic*, 79(2):496–525, 2014.

[BLR93] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *J. Comput. Syst. Sci.*, 47(3):549–595, 1993.

[Bus86] Samuel R. Buss. *Bounded Arithmetic*. Bibliopolis, 1986.

[Bus94] Samuel R. Buss. On Herbrand's theorem. In *Logic and Computational Complexity* (LCC), volume 960 of *Lecture Notes in Computer Science*, pages 195–209. Springer, 1994.

[Bus97] Samuel R. Buss. Bounded arithmetic and propositional proof complexity. In *Logic of Computation*, pages 67–121. Springer Berlin Heidelberg, 1997.

[CK07] Stephen A. Cook and Jan Krajíček. Consequences of the provability of $\mathsf{NP} \subseteq \mathsf{P/poly}$. *Journal of Symbolic Logic*, 72(4):1353–1371, 2007.

[CKK+25] Marco Carmosino, Valentine Kabanets, Antonina Kolokolova, Igor C. Oliveira, and Dimitrios Tsintsilidas. Provability of the circuit size hierarchy and its consequences. In *Innovations in Theoretical Computer Science Conference* (ITCS), volume 325, pages 30:1–30:22, 2025.

[CKKO21] Marco Carmosino, Valentine Kabanets, Antonina Kolokolova, and Igor C. Oliveira. Learn-uniform circuit lower bounds and provability in bounded arithmetic. In *Symposium on Foundations of Computer Science* (FOCS), 2021.

[CL24] Yilei Chen and Jiatu Li. Hardness of range avoidance and remote point for restricted circuits via cryptography. In *Symposium on Theory of Computing* (STOC), pages 620–629, 2024.

[CLMP25] James Cook, Jiatu Li, Ian Mertz, and Edward Pyne. The structure of catalytic space: Capturing randomness and time via compression. In *Symposium on Theory of Computing,* (STOC), pages 554–564, 2025.

[CLO24] Lijie Chen, Jiatu Li, and Igor C. Oliveira. Reverse mathematics of complexity lower bounds. In *Symposium on Foundations of Computer Science* (FOCS), pages 505–527, 2024.

[CLO25] Lijie Chen, Jiatu Li, and Igor C. Oliveira. On the unprovability of circuit size bounds in intuitionistic S^1_2. *Log. Methods Comput. Sci.*, 21(3), 2025.

[CN10] Stephen A. Cook and Phuong Nguyen. *Logical Foundations of Proof Complexity.* Cambridge University Press, 2010.

[Coo75] Stephen A. Cook. Feasibly constructive proofs and the propositional calculus (preliminary version). In *Symposium on Theory of Computing* (STOC), pages 83–97, 1975.

[CRT25] Lijie Chen, Ron D. Rothblum, and Roei Tell. Fiat-Shamir in the plain model from derandomization (or: Do efficient algorithms believe that NP = PSPACE?). In *Symposium on Theory of Computing* (STOC), pages 977–985, 2025.

[CT21] Lijie Chen and Roei Tell. Hardness vs randomness, revised: Uniform, non-black-box, and instance-wise. In *Symposium on Foundations of Computer Science* (FOCS), pages 125–136, 2021.

[CT23] Lijie Chen and Roei Tell. Guest column: New ways of studying the BPP = P conjecture. *SIGACT News*, 54(2):44–69, 2023.

[CTW23] Lijie Chen, Roei Tell, and Ryan Williams. Derandomization vs refutation: A unified framework for characterizing derandomization. In *Symposium on Foundations of Computer Science* (FOCS), pages 1008–1047, 2023.

[For01] Lance Fortnow. Comparing notions of full derandomization. In *Conference on Computational Complexity* (CCC), pages 28–34, 2001.

[FSS84] Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Math. Syst. Theory*, 17(1):13–27, 1984.

[Gay24] Azza Gaysin. Proof complexity of universal algebra in a CSP dichotomy proof, 2024.

[GC25] Stefan Grosser and Marco Carmosino. Student-teacher constructive separations and (un)provability in bounded arithmetic: Witnessing the gap. In *Symposium on Theory of Computing* (STOC), pages 1341–1347, 2025.

[Gol11] Oded Goldreich. In a world of p=bpp. In Oded Goldreich, editor, *Studies in Complexity and Cryptography*, volume 6650 of *Lecture Notes in Computer Science*, pages 191–232. Springer, 2011.

[Har04] Prahladh Harsha. *Robust PCPs of proximity and shorter PCPs.* PhD thesis, Massachusetts Institute of Technology, 2004.

[HP93] Petr Hájek and Pavel Pudlák. *Metamathematics of first-order arithmetic.* Springer-Verlag, 1993.

[IK06] Russell Impagliazzo and Bruce M. Kapron. Logics for reasoning about cryptographic constructions. *J. Comput. Syst. Sci.*, 72(2):286–320, 2006.

[IKW02] Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: exponential time vs. probabilistic polynomial time. *J. Comput. Syst. Sci.*, 65(4):672–694, 2002.

[ILW23] Rahul Ilango, Jiatu Li, and Ryan Williams. Indistinguishability obfuscation, range avoidance, and bounded arithmetic. In *Symposium on Theory of Computing* (STOC), pages 1076–1089. ACM, 2023.

[IW97] Russell Impagliazzo and Avi Wigderson. P = BPP if E requires exponential circuits: Derandomizing the XOR lemma. In *Symposium on the Theory of Computing* (STOC), pages 220–229, 1997.

[Jeř04] Emil Jeřábek. Dual weak pigeonhole principle, Boolean complexity, and derandomization. *Annals of Pure and Applied Logic*, 129(1-3):1–37, 2004.

[Jeř05] Emil Jeřábek. *Weak pigeonhole principle, and randomized computation.* PhD thesis, Charles University in Prague, 2005.

[Jeř06] Emil Jeřábek. The strength of sharply bounded induction. *Mathematical Logic Quarterly*, 52(6):613–624, 2006.

[Jeř07a] Emil Jeřábek. Approximate counting in bounded arithmetic. *Journal of Symbolic Logic*, 72(3):959–993, 2007.

[Jeř07b] Emil Jeřábek. On independence of variants of the weak pigeonhole principle. *J. Log. Comput.*, 17(3):587–604, 2007.

[Jeř09] Emil Jeřábek. Approximate counting by hashing in bounded arithmetic. *Journal of Symbolic Logic*, 74(3):829–860, 2009.

[Jeř23] Emil Jeřábek. Elementary analytic functions in VTC^0. *Annals of Pure and Applied Logic*, 174(6), 2023. Article no. 103269, 50 pp.

[JJ22] Abhishek Jain and Zhengzhong Jin. Indistinguishability obfuscation via mathematical proofs of equivalence. In *Symposium on Foundations of Computer Science* (FOCS), pages 1023–1034, 2022.

[JJMP25] Abhishek Jain, Zhengzhong Jin, Surya Mathialagan, and Omer Paneth. On succinct obfuscation via propositional proofs. In *Symposium on Foundations of Computer Science* (FOCS), 2025.

[JKLM25] Zhengzhong Jin, Yael Tauman Kalai, Alex Lombardi, and Surya Mathialagan. Universal SNARGs for NP from proofs of correctness. In *Symposium on Theory of Computing* (STOC), 2025.

[JKLV24] Zhengzhong Jin, Yael Kalai, Alex Lombardi, and Vinod Vaikuntanathan. SNARGs under LWE via propositional proofs. In *Symposium on Theory of Computing* (STOC), pages 1750–1757, 2024.

[Kha24] Erfan Khaniki. Jump operators, interactive proofs and proof complexity generators. In *Symposium on Foundations of Computer Science,* (FOCS), pages 573–593, 2024.

[KKMP21] Robert Kleinberg, Oliver Korten, Daniel Mitropolsky, and Christos H. Papadimitriou. Total functions in the polynomial hierarchy. In *Innovations in Theoretical Computer Science Conference* (ITCS), pages 44:1–44:18, 2021.

[Kor21] Oliver Korten. The hardest explicit construction. In *Symposium on Foundations of Computer Science* (FOCS), pages 433–444, 2021.

[Kor22] Oliver Korten. Derandomization from time-space tradeoffs. In *Computational Complexity Conference* (CCC), pages 37:1–37:26, 2022.

[Kor25] Oliver Korten. Range avoidance and the complexity of explicit constructions. *Bull. EATCS*, 145, 2025.

[KPT91] Jan Krajíček, Pavel Pudlák, and Gaisi Takeuti. Bounded arithmetic and the polynomial hierarchy. *Annals of Pure and Applied Logic*, 52(1-2):143–153, 1991.

[Kra95] Jan Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory.* Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1995.

[Kra19] Jan Krajíček. *Proof Complexity.* Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2019.

[Kra25] Jan Krajíček. *Proof Complexity Generators.* Cambridge University Press, 2025.

[LC11] Dai Tri Man Le and Stephen A. Cook. Formalizing randomized matching algorithms. *Log. Methods Comput. Sci.*, 8(3), 2011.

[Li25] Jiatu Li. An introduction to feasible mathematics and bounded arithmetic for computer scientists. *Electron. Colloquium Comput. Complex.*, TR25-086, 2025.

[LLR24] Jiawei Li, Yuhao Li, and Hanlin Ren. Meta-mathematics of resolution lower bounds: A TFNP perspective. Preprint, 2024.

[LO23] Jiatu Li and Igor C. Oliveira. Unprovability of strong complexity lower bounds in bounded arithmetic. In *Symposium on Theory of Computing* (STOC), pages 1051–1057, 2023.

[LP22] Yanyi Liu and Rafael Pass. Characterizing derandomization through hardness of Levin-Kolmogorov complexity. In *Computational Complexity Conference* (CCC), volume 234, pages 35:1–35:17, 2022.

[LPT24] Jiatu Li, Edward Pyne, and Roei Tell. Distinguishing, predicting, and certifying: On the long reach of partial notions of pseudorandomness. In *Symposium on Foundations of Computer Science* (FOCS), pages 1–13, 2024.

[Lê14] Dai Tri Man Lê. *Bounded Arithmetic and Formalizing Probabilistic Proofs*. PhD thesis, University of Toronto, Canada, 2014.

[MDS25] Yaohua Ma, Chenxin Dai, and Elaine Shi. Quasi-linear indistinguishability obfuscation via mathematical proofs of equivalence and applications. In *International Conference on the Theory and Applications of Cryptographic Techniques* (EUROCRYPT), pages 157–186, 2025.

[MP20] Moritz Müller and Ján Pich. Feasibly constructive proofs of succinct weak circuit lower bounds. *Annals of Pure and Applied Logic*, 171(2), 2020.

[Mul18] Wolfgang Mulzer. Five proofs of Chernoff's bound with applications. *Bull. EATCS*, 124, 2018.

[Nis92] Noam Nisan. Pseudorandom generators for space-bounded computation. *Comb.*, 12(4):449–461, 1992.

[NW94] Noam Nisan and Avi Wigderson. Hardness vs randomness. *J. Comput. Syst. Sci.*, 49(2):149–167, 1994.

[Oja04] Kerry Ojakian. *Combinatorics in Bounded Arithmetic*. PhD thesis, Carnegie Mellon University, 2004.

[Oli25] Igor C. Oliveira. Meta-mathematics of computational complexity theory. *SIGACT News*, 56(1):41–68, 2025.

[Pic14] Ján Pich. *Complexity Theory in Feasible Mathematics*. PhD thesis, Charles University in Prague, 2014.

[Pic15a] Ján Pich. Circuit lower bounds in bounded arithmetics. *Annals of Pure and Applied Logic*, 166(1):29–45, 2015.

[Pic15b] Ján Pich. Logical strength of complexity theory and a formalization of the PCP theorem in bounded arithmetic. *Logical Methods in Computer Science*, 11(2), 2015.

[PS21] Ján Pich and Rahul Santhanam. Strong co-nondeterministic lower bounds for NP cannot be proved feasibly. In *Symposium on Theory of Computing* (STOC), pages 223–233, 2021.

[Pud90] Pavel Pudlák. Ramsey's theorem in bounded arithmetic. In *Computer Science Logic* (CSL), volume 533 of *Lecture Notes in Computer Science*, pages 308–317, 1990.

[PWW88] Jeff B. Paris, A. J. Wilkie, and Alan R. Woods. Provability of the pigeonhole principle and the existence of infinitely many primes. *J. Symb. Log.*, 53(4):1235–1244, 1988.

[Raz95] Alexander A. Razborov. Bounded arithmetic and lower bounds in Boolean complexity. In *Feasible Mathematics II*, pages 344–386. Birkhäuser, 1995.

[Rii93] Søren M. Riis. Making infinite structures finite in models of second order bounded arithmetic. In P. Clote and J. Krajíček, editors, *Arithmetic, Proof Theory, and Computational Complexity*, volume 23 of *Oxford Logic Guides*, pages 289–319. Oxford University Press, Oxford, 1993.

[RSW22] Hanlin Ren, Rahul Santhanam, and Zhikun Wang. On the range avoidance problem for circuits. In *Symposium on Foundations of Computer Science* (FOCS), 2022.

[Tel19] Roei Tell. Proving that prBPP=prP is as hard as proving that "almost NP" is not contained in P/poly. *Inf. Process. Lett.*, 152, 2019.

[Tha02] Neil Thapen. *The weak pigeonhole principle in models of bounded arithmetic*. PhD thesis, University of Oxford, 2002.

[Tha24] Neil Thapen. How to fit large complexity classes into TFNP. *CoRR*, abs/2412.09984, 2024.

[Tha25] Neil Thapen. On the consistency of stronger lower bounds for NEXP. *CoRR*, abs/2504.03320, 2025.

[Vad12] Salil P. Vadhan. Pseudorandomness. *Found. Trends Theor. Comput. Sci.*, 7(1-3):1–336, 2012.

[Wil14] Ryan Williams. Nonuniform ACC circuit lower bounds. *J. ACM*, 61(1):2:1–2:32, 2014.

[Yao82] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *Symposium on Foundations of Computer Science* (FOCS), pages 80–91, 1982.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il