SHORT TURING REDUCTIONS AND PARAMETERIZED PROBABILITY AMPLIFICATION

J. ANDRÉS MONTOYA

ABSTRACT. In this paper we study the parameterized complexity of probability amplification for some parameterized probabilistic classes. We prove that the classes FPT and $W[P]$ have probability amplification for one side error probabilistic computations. We also prove that it is very unlikely that $W[P]$ has probability amplification for two side error probabilistic computations

1. INTRODUCTION

In this paper we study the parameterized complexity of probability amplification, specifically we study the parameterized complexity of probability amplification for the parameterized class $W[P]$.

It is usual that arguments and techniques that work in the classical setting don't work in the parameterized setting, in the remaining cases the arguments have to be modified and many technical details have to be considered. For example, while probability amplification is an easy task in the classical setting, it is an open problem, for most of the parameterized probabilistic classes defined by means of the BP operator, if we can amplify them. Furthermore, when we can amplify, we have to use sophisticated techniques based on pseudorandom generators.

It was proved in [Mo] that:

(1) (parameterized Toda's theorem) If $\oplus \cdot FPT$ is *maj-closed*, then
$A \subseteq PH[P] \subseteq \langle p\text{-}\#WSAT(CIRC)\rangle_R$
where A is the A hierarchy of parameterized complexity theory and $\langle p\text{-}\#WSAT(CIRC)\rangle_R$ denotes the closure of $p\text{-}\#WSAT(CIRC)$ under parameterized random reductions.

(2) (parameterized Valiant-Vazirani theorem) If $\oplus \cdot FPT$ is $\vee$-*closed*, then $W[P] \subseteq BP \cdot \oplus \cdot FPT$.

(3) (parameterized Stockmeyer's theorem) If $W[P]$ is $\wedge$-*closed*, then approximate counting of problems in $\#W[P]$ belongs to the second level of the $PH[P]$ hierarchy, (the $PH[P]$ hierarchy, a parameterized analogue of the polynomial hierarchy, was introduced in [Mo]).

Note that in each one of the theorems listed above we have used an additional hypothesis concerning the closure of some parameterized class with respect to some special type of true table reductions. Why have we needed these closure assumptions? We have used these hypothesis to amplify probabilities. A natural question is the following one: To what extent does probability amplification depend on these

Date: june 2008.

Key words and phrases. computational complexity, parameterized complexity, probabilistic classes, probability amplification.

closure assumptions? This paper is a first step into this line of research, we strongly believe that this question deserves further investigation.

1.1. **Parameterized Complexity.** Parameterized complexity theory provides a framework for a refined analysis of hard algorithmic problems. Let us quote Flum and Grohe ([FG], preface page v).

Classical complexity theory analyses problems by the amount of a resource, usually time or space, that is required by algorithms solving them. The amount of the resource required is measured as a function of the input size. Measuring complexity only in terms of the input size means ignoring any structural information about the input instances in the resulting complexity theory. Sometimes, this makes problems appear harder than they typically are. Parameterized complexity theory measures complexity not only in terms of the input size, but in addition in terms of a parameter, which is a numerical value that may depend on the input in an arbitrary way. The main intention of the theory is to address complexity issues in situations where we know that the parameter is comparatively small.

A good example is the problem of evaluating database queries. From the classical point of view this problem is tractable only in very restrictive cases, (the evaluation of conjunctive queries is already NP hard!). If one review the hardness proofs for the database evaluation problem, it is easy to note that it is necessary to consider instances where the size of the query non trivially depends on the size of the database. In real life databases are huge and queries are small, it suggests that we can consider the size of the query as a parameter and measure the complexity of the problem in terms of two independent quantities, database size and query size, if we want to obtain something new we have to consider a new (parameterized) notion of tractability.

The central notion of parameterized complexity theory is *fixed parameter tractability.* It relaxes the classical notion of tractability, polynomial time solvability, by admitting algorithms whose nonpolynomial behavior is restricted only by the parameter, in addition the theory provide us with a *parameterized intractability theory* allowing us to prove the intractability of certain problems by classifying them into parameterized complexity classes by means of suitable parameterized reductions.

1.2. **Relations to previous work.** This research is related to previous work in counting complexity and in parameterized randomization.

1.2.1. *Probabilistic classes and probability amplification.* Probabilistic computation is a major topic in Complexity theory, there is a lot of work on the fine structure of probabilistic complexity classes. Parameterized probabilistic classes are introduced and studied in this research. We introduce parameterized probabilistic classes through a parameterized analogue of the classical BP and RP operators.

There is some previous work in parameterized randomization. In [AvR] a randomized *fpt* algorithm, which compute approximations to some $\#W[1]$ hard counting problems, was presented and analyzed. In [AR] the class $RP \cdot FPT$ was implicitly defined and the hypothesis $RP \cdot FPT \neq FPT$ was used to show that resolution can not be automatized. It was proved in [Mu2] that any counting problem in $\#W[P]$ can be probabilistic approximated in *fpt* time, if oracle access to $p\text{-}WSAT(CIRC)$ is provided. This theorem is a weak parameterized analogue of Stockmeyer's theorem. In [DFR] a weak parameterized analogue of the theorem of Valiant and Vazirani was proved. It was proved in [DFR] that, if there exist

suitable probability amplification algorithms, then there is a *fpt* many-one random reduction of $W[t]$ to Unique $W[t]$. There exist suitable *fpt* probability amplification algorithms? This is an important question which was left open in [DFR]. We cope, in this paper and in previous research [Mo], with a related problem: Given a parameterized class $\mathcal{C}$, does $\mathcal{C}$ have the probability amplification property? Holding $\mathcal{C}$ the probability amplification property means that this class is robust, that is to say, given L a problem in $BP\cdot\mathcal{C}$ we can decrease its error probability by reducing L to some other problem in $\mathcal{C}$. In this paper we focus our research on the class $W[P]$.

1.2.2. *Counting complexity.* A typical class of computational problems is the class of counting problems. Counting problems are at least as hard as decision problems, because if we can count the number of solutions we can decide if there exists at least one solution. Counting Complexity, the complexity analysis of counting problems, was developed by L. Valiant with a series of ground breaking articles published in 1979, [V1], [V2]. Valiant proved that some counting problems are harder than expected, he proved that the problem of counting the number of perfect matchings in a graph is $\#P$ complete, this is surprising because the corresponding decision problem, the problem of deciding if a graph has at least one perfect matching, belongs to P. The big surprise came next when S. Toda proved that every problem in the polynomial hierarchy can be reduced to any $\#P$ complete problem [T], that implies that every problem in the polynomial hierarchy can be reduced to the problem of counting the number of perfect matchings. Thus, we can conclude:

(1) Hard counting problems are much more difficult than the corresponding decision problems, (if the polynomial hierarchy does not collapse).
(2) There are counting problems which are hard, although the corresponding decision problems are tractable.

When we cope with counting problems we have the following alternative: We can try to compute approximate solutions instead of computing exact solutions. Approximating a counting problem is easier than computing exact solutions to the problem. Stockmeyer proved [S] that approximating the problem $\#SAT$ can be done in probabilistic polynomial time if oracle access to the decision problem SAT is provided, from this theorem Stockmeyer obtained, as a corollary, that approximate counting belongs to the second level of the polynomial hierarchy.

The problem considered in this paper arises in the struggle for parameterized analogues of Toda's and Stockmeyer's Theorems. This is the case because if one try to mimic Toda's proof (or Stockmeyer's proof), one has to prove that some parameterized probabilistic classes defined using the BP operator have nice probability amplification properties. Parameterized Counting Complexity is the topic of Author's Ph.D. thesis. In his thesis the author has tried to obtain parameterized analogues of some structural theorems of classical Counting Complexity. To this end, he has introduced and studied some parameterized operators analogous to the classical operators used in the definitions of the classes of the polynomial hierarchy and in the definitions of most of the classical probabilistic classes. The operator calculus that arises from these definitions is related to the operator calculus of Downey and Fellows, (see [DF] chapter 16). We consider that our approach is more naive than Downey-Fellow's approach and because of this it has some technical advantages, but it is important to remark that, while our operator calculus

works well only for parameterized classes above $W[P]$, the calculus of Downey and Fellows can be used to study the classes between FPT and $W[P]$, including the classes of the W hierarchy [DF].

This paper is related to classical Counting Complexity, (Toda's work [T] and Stockmeyer's work [S]). There are few connections between this paper and previous work on parameterized Counting Complexity. Parameterized Counting Complexity is not yet a mature theory, there are few works on the topic [FG1], [M], and no structural theorems like the one of Toda or the one of Stockmeyer. It is important to remark that one of the main open problems in parameterized Counting Complexity is the proof of a parameterized analogue of Valiant's Theorem on the complexity of counting matchings. However, Flum and Grohe in [FG1] were able to prove that the counting of cycles and paths is hard from the parameterized point of view although the corresponding decision problems are fix parameter tractable.

1.3. **Organization of the paper.** The paper is organized into seven sections. In section two we introduce the basic definitions of parameterized complexity theory. In section three we introduce the basic operators, which we use to define the parameterized probabilistic classes studied in this paper. In section four we introduce the classes $BP \cdot FPT$ and $BP \cdot W[P]$. Additionally we introduce, in section four, the main technical notion in this paper, the *pam* property. In section five we say some things about the classes $RP \cdot FPT$ and $RP \cdot W[P]$. In section six we introduce the notion of short Turing reduction, we study three specific types of short Turing reductions: Majority reductions, conjunctive reductions and disjunctive reductions. We study the relations between these reductions and probability amplification. We are able to prove that FPT has both, the *pam* property and the *R-pam* property. Furthermore we prove that $W[P]$ has the *R-pam* property, and we prove that, if $W[P]$ is closed under conjunctive reductions, then $W[P]$ has the *pam* property. In section seven we study the following question: To what extent does the *pam* property of $W[P]$ depend on $W[P]$ being closed under conjunctive reductions?

2. A technical preface

In this section we introduce the basic definitions of Parameterized Complexity Theory, much more information can be found in [DF] and in [FG].

Notation 1. *Σ^* is the set of finite 0-1 words.*

Definition 1. *A parameterized problem is a subset of $\Sigma^* \times \mathbb{N}$.*

Example 1. *$p - WSAT(CIRC)$ is the following parameterized problem*

- *Instances: (C, k), where C is a boolean circuit and $k \in \mathbb{N}$.*
- *Parameter: k.*
- *Problem: Decide if there exists a satisfying assignment of C whose **Hamming weight** is equal to k.*

The first important definition is the definition of efficient algorithm, efficient algorithms will be called fpt algorithms.

Definition 2. *An fpt algorithm is an algorithm M such that, on input (x, k), the running time of M is upperbounded by $f(k) \cdot p(|x|)$, for some computable function f and some polynomial p.*

Now we can use the notion of efficient algorithm to define a suitable notion of parameterized feasible problems. To this end we define the parameterized class FPT whose elements are the parameterized feasible problems.

Definition 3. *The parameterized class FPT is the class of parameterized problems which can be decided using an fpt algorithm.*

The next important definition is the notion of reducibility that will be used in most places of the paper.

Definition 4. *Given L, L^* two parameterized languages, L is fpt many-one reducible to L^*, (symbolically $L \preceq_{fpt} L^*$), if and only if there exist a computable function g and an fpt algorithm M such that, on input $(x,k) \in \Sigma^* \times \mathbb{N}$, the algorithm M computes a pair (x^*,k^*) that satisfies:*

1. $k^* \leq g(k)$.
2. $(x,k) \in L$ *if and only if* $(x^*,k^*) \in L^*$.

The following is a technical definition that will be used to define the parameterized class $W[P]$, the parameterized analogue of NP.

Definition 5. *Given L a parameterized problem,* $\langle L \rangle_{fpt} := \{L^* : L^* \preceq_{fpt} L\}$.

Definition 6. $W[P] := \langle p - WSAT(CIRC) \rangle_{fpt}$.

The class $W[P]$ has a good machine characterization

Definition 7. *A $W[P]$ restricted Turing machine $\mathbb{M}$ is a Turing machine for which we have:*

1. *There exist a computable function f and a polynomial p such that, on every run of $\mathbb{M}$ with input (x,k), the running time of $\mathbb{M}$ is upperbounded by $f(k) \cdot p(|x|)$.*
2. *there exists a computable function g such that, on every run of $\mathbb{M}$ with input (x,k), the machine $\mathbb{M}$ guesses at most $g(k) \cdot \log(|x|)$ nondeterministic bits.*

Theorem 1. *$L \in W[P]$ if and only if there exists a $W[P]$ restricted Turing machine $\mathbb{M}$ that decides L.*

A proof of this theorem can be found in ([FG] theorem 3.14).
Consider the following problem.

Definition 8. *p-$CLOGSAT$ is the following parameterized problem*

- *Instances: (C,k), where $k \in \mathbb{N}$ and C is boolean circuit such that the number of its input gates is upperbounded by $k \log(|C|)$.*
- *Parameter: k.*
- *Problem: Decide if (C,k) is satisfiable.*

It is easy to verify that p-$CLOGSAT$ belongs to $W[P]$, we will prove that p-$CLOGSAT$ is $W[P]$ complete, that is we will prove that p-$WSAT(CIRC)$ is *fpt* many-one reducible to p-$CLOGSAT$.

Lemma 1. *p-$CLOGSAT$ is $W[P]$ complete.*

Proof. It is straightforward to verify that p-$CLOGSAT$ belongs to $W[P]$. We only have to prove that p-$CLOGSAT$ is $W[P]$ hard. To this end we will show that p-$WSAT(CIRC)$ is *fpt* many one reducible to p-$CLOGSAT$. The proof is an easy application of the $k \log(n)$ trick of Downey and Fellows, ([FG] page 52).

Let (C,k) be an instance of $p\text{-}WSAT(CIRC)$ and let m be the number of input gates of C. We can compute in *fpt* time a circuit $D_{C,k}$ which maps $\{0,1\}^{k\log(m)}$ onto $\{s\in\{0,1\}^m : \text{the hamming weight of } s \text{ is less than or equal to } k\}$.

If we hardwire the output gates of $D_{C,k}$ and the input gates of C we obtain a circuit $H_{C,k}$ such that

(1) $H_{C,k}$ is satisfiable if and only if $(C,k)\in p\text{-}WSAT(CIRC)$.
(2) The size of $H_{C,k}$ is bigger than the size of C.
(3) The number of input gates of $H_{C,k}$ is equal to $k\log(m)$ and $k\log(m)\leq k\log(|H_{C,k}|)$.

Thus, $(H_{C,k},k)$ is an instance of $p\text{-}CLOGSAT$ such that

(1) $(C,k)\in p\text{-}WSAT(CIRC)$ if and only if $(H_{C,k},k)\in p\text{-}CLOGSAT$.
(2) $(H_{C,k},k)$ can be computed from (C,k) in *fpt* time.

Hence, we have proven that $p\text{-}WSAT(CIRC)$ is *fpt* many one reducible to $p\text{-}CLOGSAT$ □

Definition 9. *A parameterized counting problem is a function* $h:\{0,1\}^*\times\mathbb{N}\to\mathbb{N}$.

Given h,h^* two parameterized counting problems, h is *parsimonious reducible* to h^*, (symbolically $h\preceq_{par}h^*$), if and only if there exist a computable function g and an *fpt* algorithm M such that, on input (x,k), the algorithm M computes a pair (x^*,k^*) which satisfies:

(1) $k^*\leq g(k)$.
(2) $h((x,k))=h^*((x^*,k^*))$.

Definition 10. *Given a parameterized counting problem* h, *we define* $\langle h\rangle_{par}:=\{h^*:h^*\preceq_{par}h\}$.

Definition 11. $p\text{-}\#WSAT(CIRC)$ *is the following parameterized counting problem:*

- *Instances: A circuit* C *and* $k\in\mathbb{N}$.
- *Parameter:* k.
- *Problem: Compute the number of satisfying assignments of* C *whose Hamming weight is equal to* k.

Definition 12. $\#W[P]:=\langle p\text{-}\#WSAT(CIRC)\rangle_{par}$.

3. The basic operators

In this section we introduce the basic parameterized operators. We introduce these operators to define some parameterized classes, these classes are analogous to the classical classes defined by means of the classical operators BP, $\exists$, $\forall$, RP and $\oplus$.

Notation 2. *In the following, if it is clear from the context,* $\{0,1\}^f$ *will denote the set* $\{0,1\}^{f(k)\cdot\log(|x|)}$.

Definition 13. *Given* L *a parameterized language and given* $\mathcal{C}$ *a parameterized class (i.e. a set of parameterized languages closed under fpt many one reductions).*

(1) $L\in\exists\cdot\mathcal{C}$ *if and only if there exist* $\Omega\in\mathcal{C}$ *and a computable function* f *such that*
$$(x,k)\in L\iff\exists y\in\{0,1\}^f((x,y,k)\in\Omega).$$

(2) $L \in \forall \cdot \mathcal{C}$ *if and only if there exist* $\Omega \in \mathcal{C}$ *and a computable function* f *such that*
$(x,k) \in L \iff \forall y \in \{0,1\}^{f}\left((x,y,k) \in \Omega\right).$
(3) $L \in BP \cdot \mathcal{C}$ *if and only if there exist* $\Omega \in \mathcal{C}$ *and a computable function* f *such that*
- $(x,k) \in L \Rightarrow \Pr_{y \in \{0,1\}^{f}}\left[(x,y,k) \in \Omega\right] \geq \frac{3}{4}$.
- $(x,k) \notin L \Rightarrow \Pr_{y \in \{0,1\}^{f}}\left[(x,y,k) \in \Omega\right] \leq \frac{1}{4}$.

(4) $L \in RP \cdot \mathcal{C}$ *if and only if there exist* $\Omega \in \mathcal{C}$ *and a computable function* f *such that*
- $(x,k) \in L \Rightarrow \Pr_{y \in \{0,1\}^{f}}\left[(x,y,k) \in \Omega\right] \geq \frac{1}{2}$.
- $(x,k) \notin L \Rightarrow \Pr_{y \in \{0,1\}^{f}}\left[(x,y,k) \in \Omega\right] = 0$.

(5) $L \in \oplus \cdot \mathcal{C}$ *if and only if there exist* $\Omega \in \mathcal{C}$ *and a computable function* f *such that*
$(x,k) \in L \iff \left|\left\{y \in y \in \{0,1\}^{f} : (x,y,k) \in \Omega\right\}\right| \equiv 1\,(2)\,.$

Remark 1. *It follows easily from the machine characterization of* $W[P]$ *that* $W[P] = \exists \cdot FPT$.

4. The classes $BP \cdot FPT$ and $BP \cdot W[P]$

In this section we consider the classes $BP \cdot FPT$ and $BP \cdot W[P]$. The class $BP \cdot FPT$ is the class of parameterized problems which can be solved in *fpt* time using a two side error randomized algorithm. The class $BP \cdot FPT$ is, in some sense, the class of fix parameter tractable problems. The class $BP \cdot W[P] = BP \cdot \exists \cdot FPT$ is the parameterized analogue of the Arthur-Merlin class $BP \cdot \exists \cdot P$.

4.1. **A problem in $BP \cdot W[P]$.** In this section we prove that approximate counting of problems in $\#W[P]$ belongs to $BP \cdot W[P]$. We prove that, given χ a counting problem in $\#W[P]$ a related parameterized gap problem, which we call *p-approx* (χ), belongs to $BP \cdot W[P]$.

Notation 3. *Let* $A = \{n \in \mathbb{N} : n \geq 1\} \cup \left\{\frac{1}{2}\right\}$.

Given χ a counting problem in $\#W[P]$ we define a parameterized gap problem *p-approx* (χ) in the following way.

Definition 14. *p-approx* (χ) *is the parameterized gap problem defined by*
Instances: (x,k,c), *where* (x,k) *is an instance of* χ *and* $c \in A$.
Parameter: k.
Yes-instances: (x,k,c) *is a Yes-instance if and only if* $\chi(x,k) \geq 2c$.
No-instances: (x,k,c) *is a Not-instance if and only if* $\chi(x,k) \leq c$.

It was proved in [Mo] that, given $\chi \in \#W[P]$, the counting problem χ can be approximated in *fpt* time using an oracle for *p-approx*(χ). We prove that *p-approx*$(\chi) \in BP \cdot \exists \cdot FPT$. The core of the argument, in the proof of this theorem, is an standard hashing argument. Hashing allow us to transform a dicothomy of the form:

Either there are so many certificates, (more than $2c$*) or there are so few, (less than* c*).*

Into a dicothomy of the form:

The probability that there are at least one certificate is either very high (bigger then $\frac{3}{4}$*) or very small, (less than* $\frac{1}{4}$*).*

Note that this is the type of transformation that we need to prove that $p\text{-}approx(\chi) \in BP \cdot \exists \cdot FPT$. The proof resembles the final part of the proof showing that the problem *non-graphisomorphism* belongs to $BP \cdot \exists \cdot P$, (see [KST]). We begin remembering some things about U_2-hashing families. Some of the material concerning hashing is standard, we have included it for sake of completeness, more information can be found in [G].

Definition 15. *Given A, B two sets, a set $H_{A,B} \subseteq B^A$ is an U_2-hashing family if and only if for all $a, b \in A$, with $a \neq b$, and for all $c, d \in B$*
$\Pr_{h \in H_{A,B}} [h(a) = c \ \& \ h(b) = d] = \frac{1}{|B|^2}$.

From here on, the letters A, B will denote two sets, (we suppose $|A| \geq 2$), and $H_{A,B}$ will denote an U_2-universal family of hashing functions from A to B.

Proposition 1. *Given a and b two elements of A, if $a \neq b$ we have*
$\Pr_{h \in H_{A,B}} [h(a) = h(b)] = \frac{1}{|B|}$.

Proof. $\Pr_{h \in H_{A,B}} [h(a) = h(b)] = \sum_{c \in B} \Pr_{h \in H_{A,B}} [h(a) = c \ \& \ h(b) = c] = |B| \frac{1}{|B|^2} = \frac{1}{|B|}$ □

Proposition 2. *Given $a \in A$ and given $c \in B$ we have*
$\Pr_{h \in H_{A,B}} [h(a) = c] = \frac{1}{|B|}$.

Proof. Pick $b \in A - \{a\}$, we have that
$\Pr_{h \in H_{A,B}} [h(a) = c] = \sum_{d \in B} \Pr_{h \in H_{A,B}} [h(a) = c \ \& \ h(b) = d] = |B| \frac{1}{|B|^2} = \frac{1}{|B|}$ □

Suppose that A, B are two sets and suppose that $H_{A,B} \subseteq B^A$ is an U_2-hashing family. Given $\alpha \in B$ and $S \subseteq A$, we consider the random variable $Y_{S,\alpha}$, with domain $H_{A,B}$, defined by
$Y_{S,\alpha}(h) = \left| S \cap h^{-1}(\alpha) \right|$.

Lemma 2. *$E[Y_{S,\alpha}]$, the expected value of $Y_{S,\alpha}$, is equal to $\frac{|S|}{|B|}$.*

Proof. Given $a \in S$, we consider the indicator variable Y_a, with domain $H_{A,B}$, defined by
$Y_a(h) = 1$ if and only if $h(a) = \alpha$.
It is clear that $Y_{S,\alpha} = \sum_{a \in S} Y_a$ and $E[Y_{S,\alpha}] = \sum_{a \in S} E[Y_a]$. We only have to compute $E[Y_a]$ for each $a \in S$. Fixing $a \in S$, we have that $\Pr_h [h(a) = \alpha] = \frac{1}{|B|}$, (since $H_{A,B}$ is a U_2-hashing family, proposition 2). Hence, $E[Y_a] = \frac{1}{|B|}$ and $E[Y_{S,\alpha}] = \frac{|S|}{|B|}$ □

Lemma 3. *(Letfover hashing lemma)*
Given $A, B, S, \alpha, H_{A,B}$ and $Y_{S,\alpha}$ as above, we have that
$\Pr_h \left[\left| Y_{S,\alpha} - \frac{|S|}{|B|} \right| \gtrneqq \epsilon \frac{|S|}{|B|} \right] \lneqq \frac{|B|}{\epsilon^2 |S|}$.

Proof. We note that

(1) The indicator variables $(Y_a)_{a \in S}$ are pairwise independent.
(2) For all $a \in S$ we have that
$$\sigma^2_{Y_a} = E\left[Y_a^2\right] - E[Y_a]^2 = E[Y_a](1 - E[Y_a]) \leq E[Y_a] = \frac{1}{|B|}.$$

(3) $\sigma^2_{Y_{S,\alpha}} = \sum_{a \in S} \sigma^2_{Y_a} \leq \frac{|S|}{|B|}$ (lemma 6).

We can use Chebyshev's inequality (lemma 7) and facts 1-3 to claim that

$\Pr_h \left[\left| Y_{S,\alpha} - \frac{|S|}{|B|} \right| \gneqq \epsilon \frac{|S|}{|B|} \right] \lneqq \frac{\sigma^2_{Y_{S,\alpha}}}{\left(\epsilon \frac{|S|}{|B|}\right)^2} \leq \frac{|B|}{\epsilon^2 |S|}$ □

Definition 16. *Given $m \leq n$ two natural numbers, $H^*_{n,m}$ is the U_2-hashing family* $\{h_{a,b} : a, b \in \{0,1\}^n \ \& \ h_{a,b}(z) := (az + b) \restriction_m\}$.

where, given $y \in \{0,1\}^n$, we have that $y \restriction_m$ is the boolean vector of length m whose entries are the first m entries of y.

Remark 2. *Note that we can identify the sets $H^*_{n,m}$ and $\{0,1\}^{2n}$.*

Remark 3. *Note that if we fix $S \subseteq \{0,1\}^n$, and we consider the random variable $Y_{S,0^m}$ with domain $H^*_{n,m}$, we have that $E[Y_{S,0^m}] = \frac{|S|}{2^m}$.*

Let us begin with the proof. First we have to define the meaning of a gap problem belonging to $BP \cdot \exists \cdot FPT$.

We will say that *p-approx*$(\chi) \in BP \cdot \exists \cdot FPT$ if and only if there exist $\Omega \in FPT$ and two computable functions h, g such that:

- If (x,k,c) is a Yes-instance of *p-approx* (χ), then
  $\Pr_{z \in \{0,1\}^g} \left[\exists y \in \{0,1\}^h ((x,y,z,c,k) \in \Omega) \right] \geq \frac{3}{4}$
- If (x,k,c) is a No-instance of *p-approx*(χ), then
  $\Pr_{z \in \{0,1\}^g} \left[\exists y \in \{0,1\}^h ((x,y,z,c,k) \in \Omega) \right] \leq \frac{1}{4}$

Now we prove that given $\chi \in \#W[P]$, the parameterized gap problem *p-approx* (χ) belongs to $BP \cdot \exists \cdot FPT$. The proof relies on the leftover hashing lemma.

Given χ a problem in $\#W[P]$, there exist $\Omega \in FPT$ and a computable function h such that $\chi(x,k) = |S_{x,k}|$, where $S_{x,k}$ is the set $\left\{ y \in \{0,1\}^h : (x,y,k) \in \Omega \right\}$. We consider the language $\Omega^6 \in FPT$ defined by

$\Omega^6 := \left\{ (x, y_1, ..., y_6, k) : y_1, ..., y_6 \in \{0,1\}^h \ \& \ (x,y_1,k) \in \Omega, ..., (x,y_6,k) \in \Omega \right\}$.

Let $S^6_{x,k} = \left\{ (y_1, ..., y_6) : (x, y_1, ..., y_6, k) \in \Omega^6 \right\} = (S_{x,k})^6$.

Fact 1. *Given (x,k,c) an instance of p-approx (χ)*

(1) *If (x,k,c) is a Yes-instance of p-approx(χ), then $|S^6_{x,k}| \geq 2^6 c^6$.*
(2) *If (x,k,c) is a No-instance of p-approx(χ), then $|S^6_{x,k}| \leq c^6$.*

In next two lemmas we will set $n = 6h(k)\log(|x|)$ and $m = \log\left(4c^6\right)$. We can suppose, without loss of generality, that $m \leq n$.

Lemma 4. *If (x,k,c) is a Yes-instance of p-approx(χ), then*
$\Pr_{r \in H^*_{n,m}} \left[\exists y \in \{0,1\}^{6 \cdot h} \left(y \in S^6_{x,k} \ \& \ r(y) = 0^m \right) \right] \geq \frac{3}{4}$.

Proof. Suppose that (x,k,c) is a Yes-instance of *p-approx*(χ). Let Y_m be the random variable, with domain $H^*_{n,m}$, defined by

$Y_m(r) := \left| S^6_{x,k} \cap r^{-1}(0^m) \right|$

That is, we are setting $A = \{0,1\}^n$, $B = \{0,1\}^m$, $H_{A,B} = H^*_{n,m}$, $S = S^6_{x,k}$, $\alpha = 0^m$ and $Y_{S,\alpha} = Y_m$. For ρ_m, the expected value of Y_m, we have that $\rho_m \geq 16$. Now, if we use the leftover hashing lemma, choosing $\epsilon = \frac{1}{2}$, we obtain

$\Pr_{r\in H^*_{n,m}}[Y_m(r)=0]\leq \Pr_{r\in H^*_{n,m}}\left[|Y_m(r)-\rho_m|\gneqq \frac{1}{2}\rho_m\right]\leq \frac{1}{4}$.
Hence, we have
$\Pr_{r\in H^*_{n,m}}\left[\exists y\in\{0,1\}^{6\cdot h}\left(y\in S^6_{x,k}\ \&\ r(y)=0^m\right)\right]\geq 1-\frac{1}{4}=\frac{3}{4}$ □

Lemma 5. *If (x,k,c) is a No-instance of p-approx(χ), then*
$\Pr_{r\in H^*_{n,m}}\left[\exists y\in\{0,1\}^{6\cdot h}\left(y\in S^6_{x,k}\ \&\ r(y)=0^m\right)\right]\leq \frac{1}{4}$.

Proof. Suppose that (x,k,c) is a No-instance of *p-approx*(χ). First we note that for all $y\in\{0,1\}^{6\cdot h}$
$\Pr_{r\in H^*_{n,m}}[r(y)=0^m]\leq 2^{-m}$
since $H^*_{n,m}$ is a U_2-hashing family (proposition 2). Therefore
$\Pr_{r\in H^*_{n,m}}\left[\exists y\in S^6_{x,k}\ (\ r(y)=0^m)\right]\leq \sum_{y\in S^6_{x,k}}\Pr_{r\in H^*_{n,m}}[r(y)=0^m]$
$\leq \left|S^6_{x,k}\right|2^{-m}\leq c^6 2^{-m}\leq \frac{1}{4}$ □

Theorem 2. *Given $\chi\in \#W[P]$, the gap language p-approx(χ) belongs to $BP\cdot\exists\cdot FPT$*

Proof. We consider the language Ω^* defined by
$\Omega^*:=\{(x,y_1,...,y_6,c,r,k):\varphi\ \&\ \psi_c\}$
where
(1) $\varphi:=y_1,...,y_6\in\{0,1\}^h\ \&\ (x,y_1,k),...,(x,y_6,k)\in\Omega$.
(2) $\psi_c:=r\in H^*_{n,m}\ \&\ r(y_1,...,y_6)=0^i\ \&\ m=\log\left(4c^6\right)\ \&\ n=6h(k)\log(|x|)$.
(3) h is a computable function.
(4) $\Omega\in FPT$ is a language such that for every instance (x,k) of χ we have that
$\chi(x,k)=\left|\left\{y\in\{0,1\}^h:(x,y,k)\in\Omega\right\}\right|$.
Note that $\Omega^*\in FPT$. Last two lemmas imply
- If (x,k,c) is a Yes-instance of *p-approx*(χ), then
$\Pr_{r\in H^*_{n,m}}\left[\exists y_1,...,y_6\in\{0,1\}^h((x,y_1,...,y_6,c,r,k,)\in\Omega^*)\right]\geq\frac{3}{4}$.
- If (x,k,c) is a No-instance of *p-approx*(χ), then
$\Pr_{r\in H^*_{n,m}}\left[\exists y_1,...,y_6\in\{0,1\}^h((x,y_1,...,y_6,c,r,k,)\in\Omega^*)\right]\leq\frac{1}{4}$.

So, we have proven that *p-approx*(χ) belongs to $BP\cdot\exists\cdot FPT$ □

4.2. **The *pam* property.** A probabilistic parameterized class $BP\cdot\mathcal{C}$ is well behaved if $BP\cdot\mathcal{C}$ has some type of probability amplification, i.e. $BP\cdot\mathcal{C}$ is well behaved if given $L\in BP\cdot\mathcal{C}$ we can decrease the error probability associated to L by reducing L to some other problem in the class $BP\cdot\mathcal{C}$. Here, we introduce a formal notion of *well behavedness* that we call the *pam property*.

Definition 17. *Given $\mathcal{C}$ a parameterized class, $\mathcal{C}$ has the pam property if and only if for all $L\in BP\cdot\mathcal{C}$ and for all computable function g, there exist $\Omega\in\mathcal{C}$ and a computable function f such that*
- $(x,k)\in L\Rightarrow \Pr_{y\in\{0,1\}^f}[(x,y,k)\in\Omega]\geq 1-2^{-g(k)\log(|x|)}$.
- $(x,k)\notin L\Rightarrow \Pr_{y\in\{0,1\}^f}[(x,y,k)\in\Omega]\leq 2^{-g(k)\log(|x|)}$.

The following theorem was proved in [Mo].

Theorem 3. *If $\mathcal{C}$ has the pam property we have*

(1) $\exists \cdot BP \cdot \mathcal{C} \subseteq BP \cdot \exists \cdot \mathcal{C}$.
(2) *(The Arthur-Merlin hierarchy based on $\mathcal{C}$ collapses) For any $n \geq 1$ we have* $\underbrace{BP \cdot \exists ... BP \cdot \exists \cdot \mathcal{C}}_{n\text{-}times} \subseteq BP \cdot \exists \cdot \mathcal{C}$.

5. The classes $RP \cdot FPT$ and $RP \cdot W[P]$

In this section we consider the classes $RP \cdot FPT$ and $RP \cdot W[P]$. The class $RP \cdot FPT$ is constituted by the parameterized problems that can be solved in *fpt* time using an one side error randomized algorithm. The class $RP \cdot FPT$ contains some interesting problems. Consider the following example.

Suppose C is an arithmetic circuit. The multiplication depth of C, which we denote $d(C)$, is equal to the maximum number of multiplication gates met on some path from an input to the output.

Fact 1. *(p-PIT, polynomial identity testing)*

- *Instances: An arithmetic circuit C.*
- *Parameter: $d(C)$.*
- *Problem: Is p_C nonzero?*

It is proved in [Mu] that *p-PIT* belongs to $RP \cdot FPT$.

The definition of the classes $RP \cdot \mathcal{C}$ suggests, that we should consider a second type of amplification property.

Definition 18. *Given $\mathcal{C}$ a parameterized class, we say that $\mathcal{C}$ has the R-pam property if and only if given L,Σ two languages and given f,g two computable functions, if*

(1) $L \in \mathcal{C}$.
(2) $(x,k) \in \Sigma \Rightarrow \Pr_{y \in \{0,1\}^f}[(x,y,k) \in L] \geq \frac{1}{g(k)\log(|x|)}$.
(3) $(x,k) \notin \Sigma \Rightarrow \Pr_{y \in \{0,1\}^f}[(x,y,k) \in L] = 0$.

Then, we have that $\Sigma \in RP \cdot \mathcal{C}$.

In next section we will prove that the classes FPT and $W[P]$ have the *R-pam* property.

Remark 4. *Up to the moment we know that*

(1) *There are natural problems in the class $BP \cdot \exists \cdot FPT$, since we have proven that approximate counting problems related to $\#W[P]$ counting problems belong to $BP \cdot \exists \cdot FPT$.*
(2) *There are natural problems in the class $RP \cdot FPT$, for example p-PIT.*
(3) *There are interesting problems in the class $BP \cdot FPT$. Gap problems related to parameterized approximation problems are natural members of the class $BP \cdot FPT$.*

Are there interesting-natural problems in $RP \cdot \exists \cdot FPT$?

6. Short Turing reductions

In this section we introduce the notion of short Turing reductions. Short Turing reductions are parameterized Turing reductions [DF] for which the number of queries is small, that is the number of queries is upperbounded for a quantity of

the form $f(k)\log(|x|)$. Note that, the definition of parameterized (standard) Turing reductions allows us to consider reductions for which the number of queries is lowerbounded for a quantity of the form $f(k)\,p(|x|)$, where p is some polynomial.

Definition 19. *Given L,Σ two languages, we say that L is ST-reducible (short Turing reducible) to Σ if and only if there exists an fpt algorithm M such that*

1. *M has oracle access to Σ.*
2. *M decides L.*
3. *There exists a computable function f such that, on input (x,k), the algorithm M makes at most $f(k)\log(|x|)$ oracle queries.*
4. *There exists a computable function g such that any query (y,k^*), performed during the computation of M, on input (x,k), satisfies $k^* \leq g(k)$.*

Remark 5. *It is important to remark that FPT is closed under parameterized Turing reductions.*

In this paper we will only consider short true table reductions. We do not give a explicit definition of such reductions, but we consider and analyze three different types (examples) of short true table reductions.

6.1. **Example 1: Majority reductions.** In this section we introduce the notion of majority reductions and we explore the relation between the closure of $\mathcal{C}$ under majority reductions and the probability-amplification properties of $BP\cdot\mathcal{C}$. We prove that if $\mathcal{C}$ is *maj*-closed, then $\mathcal{C}$ has the *pam* property. This theorem was proved in [Mo2], we will present the proof for sake of completeness. The proof of this theorem is very similar to the classical analogue, but in addition we have to use in the proof the pseudorandom generator of Ajtai, Komlos and Szemeredi [G] in order to save random bits.

Notation 4. *From here on $\otimes$ will denote the boolean operator Majority.*

Definition 20. *L is majority reducible to L^* if and only if there exist an fpt algorithm M and two computable functions f,g such that, on input (x,k), the algorithm M computes a sequence*
$(x_1,k_1),...,\left(x_{f(k)\log(|x|)},k_{f(k)\log(|x|)}\right)$
that satisfies:

1. $(x,k)\in L \Leftrightarrow \bigotimes_{j\leq f(k)\log(|x|)}(x_j,k_j)\in L^*$.
2. *For all $i\leq f(k)\log(|x|)$ we have that $k_i\leq g(k)$.*

We will say that $\mathcal{C}$ is maj-closed if and only if $\mathcal{C}$ is closed under majority reductions.

The following theorem says us that in order to amplify the success probability (equivalently, to decrease the error probability), we can make a big saving of random bits if we use a suitable *pseudorandom generator*.

Theorem 4. *(AKS theorem)*
There exist an algorithm, namely AKS, and constants $N_1,N_2\in\mathbb{N}$, such that for every $m,i\in\mathbb{N}$ and for all $a\in\{0,1\}^{N_1(m+i)}$, on input (a,i,m), the algorithm AKS computes a sequence $a_1,...,a_{iN_2}\in\{0,1\}^m$ that satisfies: Given $A\subset\{0,1\}^m$

1. $|A|\geq\frac{3}{4}2^m \Rightarrow \Pr_{a\in\{0,1\}^{N_1(m+i)}}\left[\bigotimes_{j\leq iN_2}a_j\in A\right]\geq 1-2^{-i}$.
2. $|A|\leq\frac{1}{4}2^m \Rightarrow \Pr_{a\in\{0,1\}^{N_1(m+i)}}\left[\bigotimes_{j\leq iN_2}a_j\in A\right]\leq 2^{-i}$.

(3) *The running time of AKS is bounded by a polynomial* $p(m,i)$.

Remark 6. *The algorithm AKS is the pseudorandom generator of Ajtai, Komlos and Szemeredi, which is based on expander graphs* [G].

Using AKS theorem, we can easily prove the following theorem which says us that there exists a deep relation between the closure under majority reductions and probability amplification.

Theorem 5. *If* $\mathcal{C}$ *is maj-closed, then* $\mathcal{C}$ *has the pam property.*

Proof. Let L, Ω be languages such that $L \in BP \cdot \mathcal{C}$, $\Omega \in \mathcal{C}$ and

- $(x,k) \in L \Rightarrow \Pr_{y \in \{0,1\}^f}[(x,y,k) \in \Omega] \geq \frac{3}{4}$.
- $(x,k) \notin L \Rightarrow \Pr_{y \in \{0,1\}^f}[(x,y,k) \in \Omega] \leq \frac{1}{4}$.

where f is some suitable computable function.

Given g a computable function we define Ω^g in the following way

$$\Omega^g := \left\{ (x,y,k) : y \in \{0,1\}^{N_1(f+g)} \ \& \bigotimes_{j \leq N_2 g(k)\log(|x|)} (x, z_j, k) \in \Omega \right\}$$

where $z_1, \ldots, z_{N_2 g(k)\log(|x|)}$ is the output-sequence of the algorithm AKS on input $(y, g(k)\log(|x|), f(k)\log(|x|))$. $\Omega^g \in \mathcal{C}$ because $\mathcal{C}$ is maj-closed and it follows from the AKS theorem that

- $(x,k) \in L \Rightarrow \Pr_{y \in \{0,1\}^{N_1(f+g)}}[(x,y,k) \in \Omega^g] \geq 1 - 2^{-g(k)\log(|x|)}$.
- $(x,k) \notin L \Rightarrow \Pr_{y \in \{0,1\}^{N_1(f+g)}}[(x,y,k) \in \Omega^g] \leq 2^{-g(k)\log(|x|)}$.

Therefore, $\mathcal{C}$ has the *pam* property □

Corollary 1. (1) *FPT has the pam property.*

(2) *If* $W[P]$ *is maj-closed* $W[P]$ *has the pam property*

(3) *If* $W[P]$ *is maj-closed the Arthur-Merlin hierarchy collapses*

Proof. FPT is closed under majority reductions □

Remark 7. *It was proved in* [Mo] *that*

(the parameterized theorem of Toda) If $\oplus \cdot FPT$ *is maj-closed, then any problem in the A hierarchy is random reducible to p-*$\#WSAT(CIRC)$*, (the A hierarchy is one of the two most studied hierarchies in parameterized complexity, more information can be found in* [DF]*).*

6.2. **Example 2: Conjunctive reductions.** In this section we introduce the notion of conjunctive reduction and we study the power of this type of reductions.

Definition 21. *L is conjunctive-reducible to* L^* *if and only if there exist an fpt algorithm M and two computable functions* f, g *such that, on input* (x,k)*, the algorithm M computes a sequence* $(x_1,k_1), ..., \left(x_{f(k)\log(|x|)}, k_{f(k)\log(|x|)}\right)$ *which satisfies:*

(1) $(x,k) \in L \Leftrightarrow \bigwedge_{i \leq f(k)\log(|x|)} (x_i, k_i) \in L^*$.

(2) *For all* $i \leq f(k)log(|x|)$ *we have that* $k_i \leq g(k)$.

We will say that $\mathcal{C}$ is $\wedge$-*closed* if and only if for all L, if there exists $L^* \in \mathcal{C}$ such that L is conjunctive-reducible to L^*, then $L \in \mathcal{C}$. Next proposition says that, if $W[P]$ is $\wedge-$closed, then $W[P]$ is *maj*-closed, (the proposition was proved in [Mo2]).

Proposition 3. *If $W[P]$ is $\wedge-$closed, then $W[P]$ is maj-closed.*

From last proposition, we obtain the following interesting corollaries.

Corollary 2. (1) *If $W[P]$ is $\wedge-$closed, then $W[P]$ has the pam property.*
(2) *If $W[P]$ is $\wedge-$closed, then the Arthur Merlin hierarchy collapses.*

Remark 8. *It was proved in* [Mo] *that*

(1) *(the parameterized theorem of Stockmeyer) If $W[P]$ is $\wedge$-closed, approximate counting of problems in $\#W[P]$ belongs to $\forall \cdot \exists \cdot FPT$.*
(2) *(the parameterized theorem of Lautemann and Sipser) If $W[P]$ is $\wedge$-closed, then $BP \cdot W[P] \subseteq \forall \cdot \exists \cdot FPT$.*

6.3. **Example 3: Disjunctive reductions.** In this section we introduce the notion of disjunctive reductions. We study the relation between the *R-pam* property and the closure under disjunctive reductions. Actually, we prove that, if $\mathcal{C}$ is closed under disjunctive reductions, then $\mathcal{C}$ has the *R-pam* property.

Definition 22. *Given two parameterized languages Γ and Γ^*, the language Γ is disjunctive reducible to Γ^* if and only if there exist an fpt algorithm M and two computable functions f, g such that, on input (x,k), the algorithm M computes a sequence $(x_1,k_1),...,\left(x_{f(k)\log(|x|)},k_{f(k)\log(|x|)}\right)$ which satisfies:*

(1) $(x,k) \in \Gamma \Longleftrightarrow \bigvee_{i \leq f(k)\log(|x|)} (x_i,k_i) \in \Gamma^*$.
(2) *For all $i \leq f(k)\log(|x|)$, we have that $k_i \leq g(k)$.*

Given $\mathcal{C}$ a parameterized class, we say that $\mathcal{C}$ is $\vee$-*closed* if and only if $\mathcal{C}$ is closed under disjunctive reductions. We prove that, given $\mathcal{C}$ a parameterized class, if $\mathcal{C}$ is $\vee$-closed, then $\mathcal{C}$ has the *R-pam* property. To this end, we use the two bit sampling amplification technique [Mo].

Given $i \in \mathbb{N}$ and $j \in \mathbb{F}_{2^i}$, (where $\mathbb{F}_{2^i}$ is the Galois field of size 2^i), we consider the random variable Y_j^i, with domain $(\mathbb{F}_{2^i})^2$, defined by

$Y_j^i((a,b)) := aj + b$

where the arithmetical operations are the operations of the Galois field $\mathbb{F}_{2^i}$.

Fact 2. *For all $i \geq 0$ and for all $j \in \mathbb{F}_{2^i}$, the random variable Y_j^i is uniformly distributed on $\mathbb{F}_{2^i}$, i.e. for all $c \in \mathbb{F}_{2^i}$ we have that*

$\Pr_{a,b}\left[Y_j^i((a,b)) = c\right] = \frac{1}{2^i}$.

Proof. Fix i, j, a and c; we have that $Y_j^i((a,b)) = c$ if and only if $b = c - aj$. Hence, we have

$\Pr_{a,b}\left[Y_j^i((a,b)) = c\right] = \frac{2^i}{2^{2i}} = \frac{1}{2^i}$ □

Fact 3. *The sequence $Y_1^i,...,Y_{2^i}^i$ is **pairwise independent**, i.e. for all $c,d \in \mathbb{F}_{2^i}$ and for all $j_1, j_2 \in \mathbb{F}_{2^i}$, if $j_1 \neq j_2$. Then*

$\Pr_{a,b}\left[Y_{j_1}^i((a,b)) = c \;\&\; Y_{j_2}^i((a,b)) = d\right] = \frac{1}{2^{2i}}$.

Proof. Fix i, j_1, j_2 and c, with $j_1 \neq j_2$; we have that $Y_{j_1}^i((a,b)) = c$ and $Y_{j_2}^i((a,b)) = d$ if and only if (a,b) is a solution of the linear system

$$\begin{bmatrix} j_1 & 1 \\ j_2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}$$

Note that this system has exactly one solution, namely (a_0,b_0), since

$$\det\left(\begin{bmatrix} j_1 & 1 \\ j_2 & 1 \end{bmatrix}\right) = j_1 - j_2 \neq 0$$

Thus, we have that
$\Pr_{a,b}\left[Y^i_{j_1}((a,b)) = c \;\&\; Y^i_{j_2}((a,b)) = d\right] = \Pr_{a,b}\left[a = a_0 \;\&\; b = b_0\right] = \frac{1}{2^{2i}}$ □

Lemma 6. *Given $X_1, ..., X_m$ a sequence of pairwise independent random variables and $X = X_1 + ... + X_m$, we have that*
$\sigma^2_X = \sum_{i\leq m} \sigma^2_{X_i}$.

Proof. $\sigma^2_X = E\left[\left(\sum_{i\leq m} X_i\right)^2\right] - E\left[\sum_{i\leq m} X_i\right]^2 =$

$$\sum_{i\leq m} E\left[X_i^2\right] + \sum_{i\neq j} 2E\left[X_iX_j\right] - \left(\sum_{i\leq m} E\left[X_i\right]^2 + \sum_{i\neq j} 2E\left[X_i\right]E\left[X_j\right]\right)$$

If we use the pairwise independence of the sequence $X_1, ..., X_m$, we have that given $i, j \leq m$, if $i \neq j$, then $E\left[X_iX_j\right] = E\left[X_i\right]E\left[X_j\right]$. It implies that
$\sigma^2_X = \sum_{i\leq m} E\left[X_i^2\right] - \sum_{i\leq m} E\left[X_i\right]^2 = \sum_{i\leq m} E\left[X_i^2\right] - E\left[X_i\right]^2 = \sum_{i\leq m} \sigma^2_{X_i}$ □

Lemma 7. *(Chebyshev's Inequality)*
Given X a random variable with expected value ρ_X and standard deviation equal to σ_X, we have that
$\Pr\left[|X - \rho_X| \geq \lambda\sigma_X\right] \leq \frac{1}{\lambda^2}$.

We are ready to use two bit sampling to prove that, if $\mathcal{C}$ is $\vee$-closed, then $\mathcal{C}$ has the *R-pam* property.

Let L be a language for which there exist $\Omega \in \mathcal{C}$ and two computable functions f and g such that, given (x,k) an instance of L we have

- $(x,k) \in L \Rightarrow \Pr_{y\in\{0,1\}^f}\left[(x,y,k) \in \Omega\right] \geq \frac{1}{g(k)\log(|x|)}$.
- $(x,k) \notin L \Rightarrow \Pr_{y\in\{0,1\}^f}\left[(x,y,k) \in \Omega\right] = 0$.

We will prove that L belongs to $RP \cdot \mathcal{C}$ (we are supposing that $\mathcal{C}$ is $\vee$-closed). The following technical theorem is the core of our two bit sampling amplification algorithm.

Let (x,k) be a positive instance of L and let $S_{x,k} = \left\{y \in \{0,1\}^f : (x,y,k) \in \Omega\right\}$.

Theorem 6. *(Two bit sampling amplification)*
If $2^{f(k)\log(|x|)} \geq 3g(k)\log(|x|)$ we have
$\Pr_{a,b\in\{0,1\}^f}\left[\bigvee_{j\leq 4g(k)\log(|x|)} (aj+b) \in S_{x,k}\right] \geq \frac{1}{2}$

Proof. Pick $A \subseteq S_{x,k}$ such that for some $\varepsilon \in \left[0, 2^{-f(k)\log(|x|)}\right]$ we have that
$\Pr_y\left[y \in A\right] = \frac{1}{g(k)\log(|x|)} + \varepsilon$.

Given $j \leq 4g(k)\log(|x|)$, the random variable X_j is the bivalued random variable, with domain $\left(\mathbb{F}_{2^{f(k)\log(|x|)}}\right)^2$, defined by
$X_j((a,b)) := 1$ if and only if $(aj+b) \in A$
For any $j \leq 4g(k)\log(|x|)$, we have

(1) $E(X_j) = \frac{1}{g(k)\log(|x|)} + \varepsilon$.
(2) $\sigma^2_{X_j} = \frac{g(k)\log(|x|)(1-2\varepsilon)-1}{(g(k)\log(|x|))^2} + \left(\varepsilon - \varepsilon^2\right)$.

Let X be the random variable $X_1+...+X_{4g(k)\log(|x|)}$. It is clear that $E(X)=4+4g(k)\log(|x|)\varepsilon$. Moreover we have, as a consequence of the pairwise independence of the sequence $X_1,...,X_{4g(k)\log(|x|)}$, that

$\sigma_X^2=\frac{4(g(k)\log(|x|)(1-2\varepsilon)-1)}{g(k)\log(|x|)}+4g(k)\log(|x|)\left(\varepsilon-\varepsilon^2\right)$

Given $\lambda:=\frac{\sqrt{12}}{\sigma_X}$ we have

$\Pr[X=0]\leq\Pr\left[|X-E(X)|\geq\sqrt{12}\right]=\Pr[|X-E(X)|\geq\lambda\sigma_X]$

$\leq\frac{1}{\lambda^2}=\frac{g(k)\log(|x|)(1-2\varepsilon)-1}{3g(k)\log(|x|)}+\frac{1}{3}g(k)\log(|x|)\left(\varepsilon-\varepsilon^2\right)\leq$

$\frac{1}{3}+\frac{1}{3}\left(g(k)\log(|x|)\,2^{-f(k)\log(|x|)}\right)\leq\frac{4}{9}$.

Thus, $\Pr[X=0]\lneq\frac{1}{2}$ and

$\Pr_{a,b\in\{0,1\}^f}\left[\bigvee_{j\leq 4g(k)\log(|x|)}(aj+b)\in S_{x,k}\right]\geq$

$\Pr_{a,b\in\{0,1\}^f}\left[\bigvee_{j\leq 4g(k)\log(|x|)}(aj+b)\in A\right]\geq\frac{1}{2}$ □

Theorem 7. *If $\mathcal{C}$ is $\vee$-closed, then $\mathcal{C}$ has the R-pam property.*

Proof. Suppose that $\Omega\in\mathcal{C}$ and suppose that L is a language for which there exist two computable functions f,g which satisfy:

- $(x,k)\in L\Rightarrow\Pr_{y\in\{0,1\}^f}[(x,y,k)\in\Omega]\geq\frac{1}{g(k)\log(|x|)}$.
- $(x,k)\notin L\Rightarrow\Pr_{y\in\{0,1\}^f}[(x,y,k)\in\Omega]=0$

We consider the language Ω^* defined by

$(x,y,k)\in\Omega^*$ if and only if $y\in\{0,1\}^{2f}$ and $\bigvee_{j\leq 4g(k)\log(|x|)}(x,y_1j+y_2,k)\in\Omega$

where $y=y_1y_2$ and $y_1,y_2\in\{0,1\}^f$ and the arithmetical operations are computed in the field $\mathbb{F}_{2^{f(k)\log(|x|)}}$. The language Ω^* belongs to $\mathcal{C}$, since $\mathcal{C}$ is $\vee$-closed. Given (x,k) an instance of L, (we can suppose without loss of generality that $2^{f(k)\log(|x|)}\geq 3g(k)\log(|x|)$), we have that

- $(x,k)\in L\Rightarrow\Pr_{y\in\{0,1\}^{2f}}[(x,y,k)\in\Omega^*]\geq\frac{1}{2}$.
- $(x,k)\notin L\Rightarrow\Pr_{y\in\{0,1\}^{2f}}[(x,y,k)\in\Omega^*]=0$.

Thus, we have proven that L belongs to $\mathcal{C}$ and we can conclude that $\mathcal{C}$ has the *R-pam* property □

Proposition 4. *$W[P]$ is $\vee$-closed.*

Proof. Given $L\in W[P]$ there exist a $W[P]$ restricted Turing machine $\mathbb{M}$ and a computable function f such that for all x,k

(1) $(x,k)\in L$ if and only if $\mathbb{M}$ accepts (x,k).
(2) For every run of $\mathbb{M}$, on input (x,k), the machine $\mathbb{M}$ makes $f(k)log(|x|)$ nondeterministic guesses at the beginning of the computation.

Let L^* be a language such that L^* is disjunctive reducible to L and let M,g,h be the algorithm and the functions in the definition of disjunctive reduction. We will define a $W[P]$ restricted Turing machine $\mathbb{M}^*$ that decides the language L^*. $\mathbb{M}^*$ is the following machine

On input (x,k)

(1) $\mathbb{M}^*$ guesses $i\in\{1,...,h(k)log(|x|)\}$.
(2) $\mathbb{M}^*$ computes (x_i,k_i) the ith element of the output sequence of M on input (x,k).
(3) $\mathbb{M}^*$ guesses $y\in\{0,1\}^{f(k_i)\log(|x_i|)}$.

(4) $\mathbb{M}^*$ simulates the deterministic part of the computation of $\mathbb{M}$, on input (x_i, k_i), when $\mathbb{M}$ uses the nondeterministic guesses codified by y.

It is clear that $(x,k) \in L^*$ if and only if $\mathbb{M}^*$ accepts (x,k) □

Corollary 3. (1) *$W[P]$ has the R-pam property.*
(2) *FPT has the R-pam property.*

Remark 9. *It was proved in* [Mo] *that*

(1) *If $\oplus \cdot FPT$ is $\vee$-closed, then $W[P] \subseteq BP \cdot \oplus \cdot FPT$.*
(2) *(the parameterized theorem of Valiant and Vazirani) If $\oplus \cdot FPT$ is $\vee$-closed, then $W[P] \subseteq RP \cdot U \cdot FPT$, where $U \cdot FPT$ is a parameterized analogue of UP, (unambiguous polynomial time).*

6.4. **A partial summary.** Up to the moment we have proven that

Theorem 8. *(Probability amplification theorem)*

(1) *FPT has both, the pam property and the R-pam property.*
(2) *$W[P]$ has the R-pam property.*
(3) *If $W[P]$ is $\wedge$-closed, then $W[P]$ has the property.*

Last theorem suggests the following question:

To what extent, does the *pam* property of $W[P]$ depend on $W[P]$ being $\wedge$-closed?

In next section we cope with this problem.

7. The parameterized hardness of probability amplification.

In this section we prove that it is very unlikely that the parameterized Arthur-Merlin class $BP \cdot W[P]$ has the *pam* property.

Let N_1, N_2 be two natural numbers.

Definition 23. *A (N_1, N_2)-sampling algorithm is an algorithm M such that for every $y \in \{0,1\}^{N_1 k \log(n)}$ and for every $k \in \mathbb{N}$, on input (y,k), the algorithm M computes a sequence $y_1, ..., y_{N_2 k \log(n)}$ of elements of $\{0,1\}^{k \log(n)}$.*

The notion of sampling algorithm is a very general notion. The AKS algorithm and the two bit sampling algorithm used in previous sections are sampling algorithms. From a naive point of view a sampling algorithm corresponds to the first half of typical probability amplification algorithms, which are designed according to the following schema:

(1) Do sampling.
(2) Do voting, (according to some prefixed voting schema).

To begin, we consider the following language

Definition 24. *The parameterized gap problem gap-p-$CLOGSAT$ is the following problem*

- *Instances:* $(C(X,Y), k)$, *where $k \in \mathbb{N}$ and $C(X,Y)$ is a boolean circuit whose input gates are partitioned in two blocks X, Y such that $|X|, |Y| \leq k \log(|C|)$.*
- *Parameter:* k.
- *Yes-instances:* $(C(X,Y), k)$ *such that* $\Pr_{y \in \{0,1\}^{k \log(|C|)}} [(C(X,y), k) \in p\text{-}CLOGSAT] \geq \frac{3}{4}$.

- *No-instances:* $(C(X,Y),k)$ *such that* $\Pr_{y\in\{0,1\}^{k\log(|C|)}}\left[(C(X,y),k)\in p\text{-}CLOGSAT\right]\leq\frac{1}{4}$.

It is straightforward to verify that $gap\text{-}p\text{-}CLOGSAT\in BP\cdot\exists\cdot FPT$.

Suppose that $\mathcal{F}=\{f_n\}_{n\geq 1}$ is a sequence of boolean functions such that for any $n\geq 1$ the domain of f_n is the set $\{0,1\}^n$. An $\mathcal{F}$-amplification algorithm for $gap\text{-}p\text{-}CLOGSAT$ is an algorithm M_A such that for some (N_1,N_2)-sampling algorithm M the algorithm M_A works, on input $(C(X,Y),k)$, in the following way

(1) M_A chooses $y\in\{0,1\}^{N_1k\log(|C|)}$.
(2) M_A simulates the computation of M on input (y,k).
(3) Given $y_1,...,y_{N_2k\log(|C|)}$ the output sequence of M, the algorithm M_A computes $v=f_{N_2k\log(|C|)}\left(z_1^y,...,z_{N_2k\log(|C|)}^y\right)$, where for any $i\leq N_2k\log(|C|)$ we have that $z_i^y=1$ if and only if $(C(X,y_i),k)\in p\text{-}CLOGSAT$.
(4) M_A uses v to take a decision.

Let M be (N_1,N_2)-sampling algorithm and let $\mathcal{F}=\{f_n\}_{n\geq 1}$. We say that $gap\text{-}p\text{-}CLOPGSAT$ is amplified by the pair $(M,\mathcal{F})$ if and only if

(1) For any Yes-instance $(C(X,Y),k)$
$\Pr_{y\in\{0,1\}^{N_1k\log(|C|)}}\left[f_{N_2k\log(|C|)}\left(z_1^y,...,z_{N_2k\log(|C|)}^y\right)\right]\geq 1-2^{-\log(|C|)}$.
(2) For any No-instance $(C(X,Y),k)$
$\Pr_{y\in\{0,1\}^{N_1k\log(|C|)}}\left[f_{N_2k\log(|C|)}\left(z_1^y,...,z_{N_2k\log(|C|)}^y\right)\right]\leq 2^{-\log(|C|)}$.
(3) The language $\{(C_y(X,Y),k):\varphi\}$ belongs to $W[P]$, where $y\in\{0,1\}^{N_1k\log(|C|)}$, $(C_y(X,Y),k)$ is equal to $\left\langle C(X,y_1),...,C\left(X,y_{N_2\log(|C|)}\right),k\right\rangle$ and φ is equal to:

$(C(X,Y),k)$ is an instance of $gap{-}p{-}CLOGSAT$ (either a Yes-instance or a No-instance)

&

$$f_{N_2k\log(|C|)}\left(z_1^y,...,z_{N_2k\log(|C|)}^y\right)$$

Item 3 will be very important in the proof of our main result. Note that items 1 and 2 can be fulfilled using a sampling algorithm like the one of Ajtai, Komlos and Szemeredi. When we try to amplify a language in $BP\cdot W[P]$, (or in $BP\cdot\mathcal{C}$, where $\mathcal{C}$ is some nondeterministic parameterized class), the difficulties arise during the voting stage, because in this stage we are demanded to certify long sequences, and long sequences seem to require long certificates, (we could say that conditions 1 and 2 do not matter because there exist good pseudorandom generators).

Note that in the definition of sampling algorithm there are not constraints concerning the running time. It is clear that if we want to use a sampling algorithm to amplify probabilities in *fpt* time, we have to choose, to this end, an *fpt* sampling algorithm. We have not explicitly including, in the definition of sampling algorithm, a running time condition, because the main result of this chapter rules out the possibility of using a sampling algorithm to amplify $gap\text{-}p\text{-}CLOGSAT$, even if the sampling algorithm is time consuming.

First at all, we consider the following question: Which are the sequences that we can use to amplify $gap\text{-}p\text{-}CLOGSAT$?

Consider the following situation. Suppose that M is a sampling algorithm and suppose that $\{\sigma_n\}_{n\geq 1}$ is a sequence of permutations such that, for any $i\geq 1$, we

have that $\sigma_i \in \mathcal{S}_i$. Consider the sampling algorithm M^σ defined in the following way.

On input (y,k)

(1) M^σ simulates the computation of M on input (y,k).
(2) Given $y_1,...,y_{N_2 k\log(|C|)}$ the output sequence of M, the algorithm M^σ outputs

$$y_{\sigma_{N_2 k\log(|C|)}(1)},...,y_{\sigma_{N_2 k\log(|C|)}(N_2 k\log(|C|))}.$$

It is natural to impose on $\mathcal{F}$, the following condition: The sequence $\mathcal{F}$ can not distinguish the algorithms M and M^σ. It implies that, for any $n\geq 1$ the output value of f_n does not change if we permute the inputs, i.e. we are demanding that, for any $n\geq 1$, the function f_n is a symmetric boolean function. We can argue, along the similar lines, that it is natural and convenient to impose a monotonicity condition on the sequences. So, in the following, we will restrict our attention to sequences of symmetric-monotone boolean functions. Remember that given $f_n : \{0,1\}^n \to \{0,1\}$ a symmetric-monotone boolean function, there exists $T(f_n)\leq n$ such that

$f_n(X_1,...,X_n)=1$ if and only if $|\{i\leq n : X_i=1\}|\geq T(f_n)$.

The number $T(f_n)$ is called the threshold of f_n. Given $\mathcal{F}=\{f_n\}_{n\geq 1}$, a sequence of symmetric functions, we can associate to $\mathcal{F}$ the threshold function $T(\mathcal{F}):\mathbb{N}\to\mathbb{N}$ defined by: $T(\mathcal{F})(n)=T(f_n)$. We consider two cases

(1) The function $T(\mathcal{F})$ is computable and unbounded.
(2) The function $T(\mathcal{F})$ is bounded.

First, we show that if $T(\mathcal{F})$ is bounded, the language *gap-p-CLOGSAT* can not be amplified using an $\mathcal{F}$-amplification algorithm. Suppose that for all $n\geq 1$, we have that $T(\mathcal{F})(n)\leq D$. Let M be a (N_1,N_2)-sampling algorithm. We show that we can no amplify *gap-p-CLOGSAT* using the pair $(M,\mathcal{F})$.

Theorem 9. *If $T(\mathcal{F})$ is bounded, the language gap-p-CLOGSAT can not be amplified using the pair $(M,\mathcal{F})$.*

Proof. We suppose that $D\geq 2$, the case $D=1$ is easy to handle. Suppose that $(C(X),k)$ is a positive instance of $p-CLOGSAT$. Given $\overrightarrow{a}\in\{0,1\}^D$ we define a circuit $G_{\overrightarrow{a}}(X,Y)$ in the following way

$G_{\overrightarrow{a}}(X,Y) := C(X)\wedge F_{\overrightarrow{a}}(Y)$

where $Y=\{Y_1,...,Y_{k\log(|C|)}\}$ is a set of input gates such that $X\cap Y=\varnothing$ and $F_{\overrightarrow{a}}(Y)$ is the circuit $\bigwedge_{i\leq D} Y_i = \overrightarrow{a}(i)$. Note that for any $\overrightarrow{a}\in\{0,1\}^D$ the pair $(G_{\overrightarrow{a}}(X,Y),k)$ satisfies

(1) $|X|,|Y|\leq k\log(|G_{\overrightarrow{a}}|)$.
(2) $\Pr_{y\in\{0,1\}^{k\log(|C|)}}[(G_{\overrightarrow{a}}(X,y),k)\in p\text{-}CLOGSAT] =$
$\Pr_{y\in\{0,1\}^{k\log(|C|)}}[F_{\overrightarrow{a}}(y)=1] = 2^{-D}\leq \frac{1}{4}$.

From 1 and 2 we have that $(G_{\overrightarrow{a}}(X,Y),k)$ is a No-instance of *gap-p-CLOGSAT*. Hence, we have

$$\Pr_{y\in\{0,1\}^{N_1 k\log(|C|)}}[|\{i\leq N_2 k\log(|C|) : F_{\overrightarrow{a}}(y_i)=1\}|\geq D]\leq 2^{-\log(|C|)} \qquad [eq.1]$$

Note that, for any $y \in \{0,1\}^{N_1 k \log(|C|)}$ and for any $i \leq N_2 k \log(|C|)$ we have that $\bigvee_{\overrightarrow{a} \in \{0,1\}^D} F_{\overrightarrow{a}}(y_i)$. If we suppose that $\log(|C|) \gneqq D2^D$, we have

$$\Pr_{y \in \{0,1\}^{N_1 k \log(|C|)}} \left[\bigvee_{\overrightarrow{a} \in \{0,1\}^D} |\{i \leq N_2 k \log(|C|) : F_{\overrightarrow{a}}(y_i) = 1\}| \geq D \right] = 1$$

But, from [*eq*.1] we have

$$\Pr_{y \in \{0,1\}^{N_1 k \log(|C|)}} \left[\bigvee_{\overrightarrow{a} \in \{0,1\}^D} |\{i \leq N_2 k \log(|C|) : F_{\overrightarrow{a}}(y_i) = 1\}| \geq D \right] \leq$$

$$\sum_{\overrightarrow{a} \in \{0,1\}^D} \Pr_{y \in \{0,1\}^{N_1 k \log(|C|)}} [|\{i \leq N_2 k \log(|C|) : F_{\overrightarrow{a}}(y_i) = 1\}| \geq D] \leq$$

$2^D 2^{-\log(|C|)} \lneqq \frac{2^D}{D2^D} \lneqq 1$. □

Thus, we have proven that a sequence of bounded threshold can not be used to amplify *gap-p-CLOGSAT*. Now, we consider the second case, that is we suppose that the threshold function $T(\mathcal{F})$ is computable and unbounded. Given $f : \mathbb{N} \to \mathbb{N}$ a computable and unbounded function and given $D \geq 1$, we define a parameterized problem $p\text{-}CLOGSAT_{f,D}$ in the following way

Definition 25. *($p\text{-}CLOGSAT_{f,D}$)*

- *Instances:* $\left(C_1, ..., C_{f(Dk\log(|C_1|))}, k, M\right)$*, where* $k, M \in \mathbb{N}$ *and for all* $i \leq f(Dk\log(|C_1|))$ *we have that* C_i *is a circuit of size* M *whose number of input gates is less than or equal to* $k \log M$*.*
- *Parameter:* $k \in \mathbb{N}$*.*
- *Problem: Decide if for all* $i \leq f(Dk\log(|C_1|))$*, the pair* (C_i, k) *belongs to p-CLOGSAT.*

Note that, if $\left(C_1, ..., C_{f(Dk\log(|C_1|))}, k, M\right) \in p\text{-}CLOGSAT_{f,D}$, the size of their natural certificates, i.e. sequences $\left(v_1, ..., v_{f(Dk\log(M))}\right)$ of satisfying assignments, is equal to $kf(Dk\log(M))\log(M)$, which is very big because of the term $f(Dk\log(M))\log(M)$. Remember that, if a parameterized language L belongs to $W[P]$, then there exists a computable function g such that for all instance (x,k) of L, if $(x,k) \in L$, the instance (x,k) can be certified using $g(k)\log(|x|)$ nondeterministic bits. For any unbounded function f and for any $D \geq 1$ it is very unlikely that $p\text{-}CLOGSAT_{f,D}$ belongs to $W[P]$, since it is very unlikely that there exists a computable function g such that we can certify a positive random instance $\left(C_1, ..., C_{f(Dk\log(|M|))}, k, M\right)$ of $p\text{-}CLOGSAT_{f,D}$ using $g(k)\log(M)$ nondeterministic bits.

Theorem 10. *If gap-p-CLOGSAT can be amplified using the pair* $(M, \mathcal{F})$*. Then, there exists* $D \geq 1$ *such that* $p\text{-}CLOGSAT_{T(\mathcal{F}),D} \in W[P]$*.*

Proof. We suppose that *gap-p-CLOGSAT* can be amplified using the pair $(M, \mathcal{F})$, where M is a (N_1, N_2)-sampling algorithm. We will prove that $p\text{-}CLOGSAT_{T(\mathcal{F}),N_2}$ belongs to $W[P]$, (i.e. we set $D = N_2$).

Let $S := (C_1, k), ..., \left(C_{T(\mathcal{F})(N_2 k\log(|C_1|))}, k\right)$ be a sequence, such that for all $i \leq N_2 k\log(|C_1|)$, C_i is a circuit of size $|C_1|$ whose number of input gates is less than or equal to $k\log(|C_1|)$, that is the tuple $\left(C_1, ..., C_{T(\mathcal{F})(N_2 k\log(|C_1|))}, k, |C_1|\right)$ is an instance of $p\text{-}CLOGSAT_{T(\mathcal{F}),N_2}$. Note that, for any sequence $\mathcal{F}$ and for any $n \geq 1$ we have $T(\mathcal{F})(n) \leq n$, (we can suppose, without loss of generality, that for

any $n \geq 1$ we have $T(\mathcal{F})(n) \lneqq n$). Given $y \in \{0,1\}^{N_1 k \log(|C_1|)}$ the algorithm M, on input (y,k), computes a sequence $y_1, ..., y_{N_2 k \log(|C_1|)}$. From the pair (S,y) we can define a No-instance of *gap-p-CLOGSAT* in the following way:

Let $C_{y,S}(X,Y) = \bigvee_{i \leq N_2 k \log(|C_1|)} (G_i(X) \wedge Y = y_i)$

With:

- $G_i(X) := C_i(X)$ if $i \leq T(\mathcal{F})(N_2 k \log(|C_1|))$.
- $G_i(X) := (X \neq X)$ if $i \gneqq T(\mathcal{F})(N_2 k \log(|C_1|))$.

It is clear that $|C_{y,S}(X,Y)| \geq |C_1|$. Note that

$|X|, |Y| \leq k \log(|C_1|) \leq k \log(|C_{y,S}(X,Y)|)$

that is, the pair $(C_{y,S}(X,Y), k)$ is an instance of *gap-p-CLOGSAT*. We will prove that $(C_{y,S}(X,Y), k)$ is a No-instance

We have that, if $|C_1|^k \geq 4N_2 k \log(|C_1|)$,then

$\Pr_{v \in \{0,1\}^{k \log(|C_1|)}} [(C_{y,S}(X,v), k) \in p\text{-}CLOGSAT] \leq$

$\Pr_{v \in \{0,1\}^{k \log(|C_1|)}} \left[v \in \left\{y_1, ..., y_{N_2 k \log(|C_1|)}\right\}\right] \leq \frac{1}{4}$.

Thus, if $|C_1|^k \geq 4N_2 k \log(|C_1|)$, the pair $(C_{y,S}(X,Y), k)$ is a No-instance of the language *gap-p-CLOGSAT*.

Now we note that given $y \in \{0,1\}^{N_1 k \log(|C_1|)}$

$f_{N_2 k \log(|C|)}\left(z_1^y, ..., z_{N_2 k \log(|C_1|)}^y\right) = 1$

if and only if

$$\bigwedge_{i \leq T(\mathcal{F})(N_2 k \log(|C_1|))} (C_i, k) \in p\text{-}CLOGSAT$$

if and only if

$\left(C_1, ..., C_{T(\mathcal{F})(N_2 k \log(|C_1|))}, k, |C_1|\right) \in p\text{-}CLOGSAT_{T(\mathcal{F}),N_2}$.

Remember that, if *gap-p-CLOGSAT* is amplified by the pair $(M, \mathcal{F})$, we can decide if

$f_{N_2 k \log(|C|)}\left(z_1^y, ..., z_{N_2 k \log(|C_1|)}^y\right) = 1$

using a $W[P]$ restricted Turing machine. Hence, to verify if

$\left(C_1, ..., C_{T(\mathcal{F})(N_2 k \log(|C_1|))}, k, |C_1|\right) \in p\text{-}CLOGSAT_{T(\mathcal{F}),N_2}$

we can use a $W[P]$ restricted Turing machine that verifies if

$f_{N_2 k \log(|C|)}\left(z_1^y, ..., z_{N_2 k \log(|C_1|)}^y\right) = 1$

and uses this information to give us the correct answer. We can conclude that, if we could amplify *gap-p-CLOGSAT* using the pair $(M, \mathcal{F})$, then $p\text{-}CLOGSAT_{T(\mathcal{F}),N_2}$ belongs to $W[P]$ □

We have proven that *gap-p-CLOGSAT* can not be amplified using a pair $(M, \mathcal{F})$ unless $W[P]$ is closed under some special type of unbounded conjunctive reductions. This result is a partial converse of the theorem claiming that, if $W[P]$ is $\wedge$-closed, then $W[P]$ has the *pam* property.

What is the meaning of our results? We have proven that, if one wants to amplify probabilities, of languages in $BP \cdot W[P]$, using a sampling algorithm, there are three possibilities.

(1) One can choose as sampling algorithm a good pseudorandom generator M. In this case, if $W[P]$ is not $\wedge$-closed, one will be unable to fulfill condition 3 in our definition of probability amplification using a pair $(M, \mathcal{F})$, for any $\mathcal{F}$ which is *threshold-unbounded.*

(2) One can choose a sampling algorithm, which only outputs unlikely sequences that can be certified using few bits. In this case, one will be unable to cheat *gap-p-CLOGSAT*, i.e. one will be unable to amplify *gap-p-CLOGSAT*.
(3) One can choose a *threshold-bounded* sequence, in this case, given M a sampling algorithm, one will be unable to amplify *gap-p-CLOGSAT* using the pair $(M, \mathcal{F})$.

Our results rule out the possibility of amplifying the language *gap-p-CLOGSAT* using a sample and (symmetric-monotone) voting algorithm. Can someone figure out an alternative amplification procedure? We believe that we are providing strong evidence against that $W[P]$ has the *pam* property.

Acknowledgement Thanks to VIE-UIS, research project 5153-07.

References

[AR] M. Alekhnovich, A. Razborov. Resolution is not automatizable unless $W[P]$ is tractable. Proceedings 42nd IEEE FOCS, 2001, pages 210-219.

[AvR] V. Arvind and V. Raman. Approximation algorithms for some parameterized counting problems. In P. Bose and P. Morin, editors, *Proceedings of the 13th Annual International Symposium on Algorithms and Computation,* Volume 2518 of *Lecture Notes in Computer Science,* pages 453-464. Springer-Verlag, 2002.

[DF] R.G. Downey and M.R. Fellows. *Parameterized Complexity.* Springer-Verlag, 1999.

[DFR] R.G. Downey, M. Fellows, K. Regan. Parameterized circuit complexity and the W hierarchy. *Theoretical computer sciences,* 191: 97-115, 1998.

[FG] J. Flum and M. Grohe. *Parameterized Complexity Theory.* Springer-Verlag, 2006.

[FG1] J. Flum and M. Grohe. The parameterized complexity of counting problems. *SIAM Journal of Computing,* 33(4):892-922, 2004.

[G] O. Goldreich. Randomized methods in Computation. Manuscript, 2001. http//www.wisdom.weizmann.ac.il/~oded/rnd.html.

[KST] J. Kobler, U. Schoning, J. Toran. *The graph isomorphism problem: Its structural complexity.* Birkhauser, Basel, 1993.

[M] C. McCartin. Parameterized counting problems. In K. Diks and W. Rytter, editors, *Proceedings of the 27th International Symposium on Mathematical Foundations of Computer Science Logic,* Volume 2420 of *Lecture Notes in Computer Science,* pages 556-567. Springer-Verlag, 2002.

[Mo] J. Montoya. *On parameterized counting.* Ph.D. thesis, Freiburg University, August 2008.

[Mo2] J. Montoya. The parameterized complexity of probability amplification. Information processing letters (accepted).

[Mu] M. Muller. Forthcoming Ph.D. thesis, Freiburg university.

[Mu2] M. Muller. Randomized approximations of parameterized counting problems. *Proceedings 2nd IWPEC, Lecture Notes in Computer Science* (4169), pages 50-59. Springer Verlag, 2006.

[S] L. Stockmeyer. On approximation Algorithms for $\#P$. *SIAM Journal on Computing,* 14(4): 849-861, 1985.

[T] S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing,* 20(5):865-877, 1991.

[V1] L.G. Valiant. The complexity of computing the permanent. *Theoretical computer Science,* 8:189-201, 1979.

[V2] L.G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing,* 8(3):410-421, 1979.

Escuela de Matemáticas, Universidad industrial de Santander
E-mail address: amontoyaa@googlemail.com, juamonto@uis.edu.co