Error-bounded probabilistic computations between MA and AM

Elmar Böhler, Christian Glaßer,
Daniel Meister

Report No. 299 Oktober 2002

Preprint-Reihe
Fachbereich Mathematik und Informatik
Universität Würzburg

Error-bounded probabilistic computations between MA and AM

Elmar Böhler Christian Glaßer Daniel Meister

Theoretische Informatik
Julius–Maximilians–Universität Würzburg
97074 Würzburg, Germany

{boehler, glasser, meister}@informatik.uni-wuerzburg.de

7th October 2002

Abstract

We introduce the probabilistic class SBP which is defined in a BPP-like manner. This class emerges from BPP by keeping the promise of a probability gap but decreasing the probability limit from $1/2$ to exponentially small values.

We show that SBP is in the polynomial-time hierarchy, exactly between the Arthur-Merlin classes MA and AM on one the hand and between BPP and the threshold class $\mathrm{BPP}_{\mathrm{path}}$ on the other hand. We provide evidence that SBP does not coincide with these and other known complexity classes. In particular, in a suitable relativized world, SBP is not contained in Σ_2^{P}. As a consequence we get the same for $\mathrm{BPP}_{\mathrm{path}}$. This solves an open question raised by Han *et al.* [HHT97].

1 Introduction

The use of randomness is one possible extension of conventional deterministic Turing machines. The origins of this idea go back to the work of de Leeuw, Moore, Shannon, and Shapiro [dLMSS56]. In 1972 Gill started the investigation of probabilistic *polynomial-time bounded* machines [Gil72, Gil77]. Such machines can be considered as an extension of usual polynomial-time Turing machines. An (even practically) important class of languages decidable by such machines is BPP (bounded-error probabilistic polynomial-time) [Gil72, Gil77]. For each language L in this class there exists a $\rho > 1/2$ and a probabilistic polynomial-time decision procedure which finds the correct answer to arbitrary queries "$x \in L$?" with probability $> \rho$. With help of an amplification technique one can even increase this success probability to values arbitrarily close to 1. So in spite of the fact that BPP is an extension of P (which is normally considered as the class of practically solvable problems) the decision problems in BPP can still be considered as feasible in practice. A prominent example of a problem in BPP is the primality test, i.e., with high probability one can decide whether a given number is a prime.

The Topic. When looking at the definition of BPP there are two things that make this class different from P. On one hand there is a *probability limit* of $1/2$, i.e., an input is accepted if and only if the machine accepts with probability $> 1/2$. On the other hand for a suitable $\varepsilon > 0$ there is a *probability gap*, i.e., the machine promises that its acceptance probability is never in the interval $[1/2 - \varepsilon, 1/2 + \varepsilon]$. This paper studies what happens when one lowers the probability limit. It is known that nothing happens when the probability limit is decreased by a polynomial factor. However, this changes when we decrease it by an exponential factor. So the focus of this paper is on probabilistic polynomial-time machines that have an exponentially small probability limit and still keep the promise of a probability gap. The class of languages accepted by such machines is denoted by SBP (*small bounded-error probability*).

Motivation and Related Work. The class SBP emerges in different contexts. So far we looked at SBP as a *generalization of* BPP. In order to motivate our investigations and to explain why we think this class is interesting we present the following alternative ways that SBP can be looked at.

When one abstains from the probability gap in the definition of BPP this yields the class PP (probabilistic polynomial-time). Since PP can be defined via GapP functions and since these functions have different characterizations [FFK94] the following statements are equivalent to saying that $L \in \mathrm{PP}$.

1. There is a nondeterministic polynomial-time machine M with $x \in L \iff \mathrm{acc}_M(x) > \mathrm{rej}_M(x)$.
2. There exist $f \in \#\mathrm{P}$ and $g \in \mathrm{FP}$ such that $x \in L \iff f(x) > g(x)$.
3. There exist $f, g \in \#\mathrm{P}$ such that $x \in L \iff f(x) > g(x)$.

Interestingly, this equivalence completely disappears when we return to the demand of a probability gap. By this we mean that there must be some $\varepsilon > 0$ such that either $\mathrm{acc}_M(x) > (1+\varepsilon) \cdot \mathrm{rej}_M(x)$ or $\mathrm{acc}_M(x) < (1-\varepsilon) \cdot \mathrm{rej}_M(x)$; the probability gaps for the statements 3 and 2 are defined analogously. It is not difficult to see that with this modification, statement 1 describes just BPP. Moreover, we will see that statement 3 meets exactly the threshold class $\mathrm{BPP_{path}}$ which was introduced by Han *et al.* [HHT97]. But what about statement 2 when demanding a probability gap?

We will see that apart from the original definition of SBP one can allow any polynomial-time computable probability limit. This means that SBP can be characterized by the following equivalence: $L \in \mathrm{SBP}$ if and only if there exist a probability gap $\varepsilon > 0$, a probability limit $g \in \mathrm{FP}$ and an $f \in \#\mathrm{P}$ such that

$$
\begin{aligned}
x \in L &\implies f(x) > (1+\varepsilon) \cdot g(x) \quad \text{and} \\
x \notin L &\implies f(x) < (1-\varepsilon) \cdot g(x).
\end{aligned}
$$

This shows that statement 2 with a probability gap yields our new class SBP. So when starting from three equivalent characterizations of PP and when introducing a probability gap then the equivalence disappears and one meets the three classes BPP, SBP and $\mathrm{BPP_{path}}$. In particular this shows that SBP can be thought of as a *restriction of* $\mathrm{BPP_{path}}$ and therefore $\mathrm{BPP} \subseteq \mathrm{SBP} \subseteq \mathrm{BPP_{path}}$.

Another context where SBP raises interesting questions aims at SBP's *relationship to gap-definable counting classes*, in particular with the class AWPP introduced by Fenner *et al.* [FFKL93, FFKL]. Starting from a new characterization of AWPP [Fen02] we show that the $\#\mathrm{P}$ counterpart of AWPP is contained in SBP.

Our Contribution. After showing that SBP admits a certain kind of amplification we turn to investigate SBP with respect to other complexity classes. When looking at SBP's definition one notices a similarity to the definitions of strong counting classes. So at first glance it is not obvious that SBP is in the polynomial-time hierarchy. We show that SBP is located exactly between Babai's [Bab85] Arthur-Merlin classes MA and AM. In particular, it is contained in the class Π_2^{P} of the polynomial-time hierarchy. In the proof we use similar arguments on linear hash functions as in the proof for $\mathrm{BPP} \subseteq \mathrm{PH}$ [Lau83, Sip83]. Furthermore, we show that $\mathrm{BPP} \subseteq \mathrm{SBP} \subseteq \mathrm{BPP_{path}}$ (cf. Figure 1).

On the basis of collapse consequences for the polynomial-time hierarchy and on the basis of oracle constructions we give evidence that SBP does not coincide with known complexity classes like BPP, $\mathrm{BPP_{path}}$, MA, AM and AWPP. A summary of all oracle separations can be found in Figure 2.

When looking at the inclusion $\mathrm{SBP} \subseteq \Pi_2^{\mathrm{P}}$ one might hope that also $\mathrm{SBP} \subseteq \Sigma_2^{\mathrm{P}}$. We show that this is not true in a suitable relativized world. Since $\mathrm{SBP} \subseteq \mathrm{BPP_{path}}$ holds relativizable this oracle shows that $\mathrm{BPP_{path}} \not\subseteq \Sigma_2^{\mathrm{P}}$ in some relativized world. This solves an open question of Han *et al.* [HHT97] which aimed at the relation of $\mathrm{BPP_{path}}$ to $\mathrm{R^{NP}}$ and Σ_2^{P}. Moreover, with respect to this oracle, SBP is not closed under complementation.

Paper Outline. After this introduction we start with some preliminaries. Then in section 3 we introduce SBP, show different characterizations for this class and prove that it admits amplification and that it is closed under union. Furthermore, we show that $\mathrm{BP}\cdot\mathrm{UP} \cup \mathrm{MA} \subseteq \mathrm{SBP} \subseteq \mathrm{BPP}_{\mathrm{path}} \cap \mathrm{AM}$ and we give a picture (cf. Figure 1) that compares SBP with other complexity classes. In section 4 we go into other complexity classes that are interesting with respect to SBP. In particular we show that the $\#\mathrm{P}$ counterpart of AWPP is in SBP. In section 5 we provide evidence (by means of collapse consequences and oracle constructions) that all inclusions we discussed in the previous sections are strict. In particular we construct a relativized world where SBP is not contained in Σ_2^{P}. As a consequence we obtain that $\mathrm{BPP}_{\mathrm{path}} \not\subseteq \Sigma_2^{\mathrm{P}}$ with respect to this oracle

2 Preliminaries

We fix the finite alphabet $\Sigma \stackrel{df}{=} \{0,1\}$. For the definition of P, NP, PP, the classes of the polynomial-time hierarchy and standard notions of complexity theory see any text book, e.g., [Pap94, BDG95]. For a nondeterministic polynomial-time Turing machine M, let $\mathrm{acc}_M(x)$ (resp., $\mathrm{rej}_M(x)$) denote the number of accepting (resp., rejecting) paths of M on input x. Moreover, let $\mathrm{total}_M(x) \stackrel{df}{=} \mathrm{acc}_M(x) + \mathrm{rej}_M(x)$ denote the total number of paths. Throughout the paper, if not stated otherwise, variables are natural numbers and polynomials have natural coefficients. The characteristic function of a set B is denoted by c_B.

Since we will have a very close look at classes that are defined via probabilistic machines, we will introduce them here. A *probabilistic machine* works like a deterministic machine that has the additional ability to make randomized operations. So, for example, a program of a probabilistic machine could in one step assign to a variable x the value 3 with probability $\frac{1}{3}$ and the value 17 with probability $\frac{2}{3}$. In consequence, the result of a computation of such a machine, since it may depend on several random decisions, is randomized. For instance, a probabilistic machine may return 0 with probability $\frac{1}{10}$ and 1 with probability $\frac{9}{10}$. We will only regard a special type of probabilistic machines, namely those that make a random decision between two alternatives every step, and each alternative is chosen with a probability of $\frac{1}{2}$. Besides that, our main interest is in *balanced* machines, i.e. machines that for an input of length n always make the same number of random decisions. Henceforth, if we talk of probabilistic machines, we mean balanced machines, unless we explicitly announce them to be unbalanced (as needed in the definition of $\mathrm{BPP}_{\mathrm{path}}$). If such a machine stops after n steps, the probability that it has made one concrete series of random choices is exactly 2^{-n}. Hence, the probability of a specific result x of such a machine is $k \cdot 2^{-p}$, where k is the number of series of random choices, that lead to the output x. Let M be a probabilistic machine (maybe unbalanced). For an $x \in \Sigma^*$, we write $\mathrm{prob}_M(x)$ to express the probability that M accepts x .

Another equivalent model of probabilistic machines is that of machines where the path of computation is nondeterministically split in two in each step. We require each path of computation of the machine to have the same length and say that the probability that the machine outputs x is the number of paths that output x divided by all paths of the machine. We express the correspondence between the probability of acceptance and the number of accepting paths in the proposition below. Before that, to avoid cumbersome notation, we define for every set $B \subseteq (\Sigma^*)^{n+1}$, every function $f : \mathbb{N} \to \mathbb{N}$, and all $x \in \Sigma^*$: $\mathrm{count}_B^{=f}(x_1, \ldots, x_n) =_{\mathrm{def}} \#\{y : |y| = f(|x_1 \cdots x_n|) \text{ and } (x_1, \ldots, x_n, y) \in B\}$. As a rule we will use this notation for $n = 1$. Obviously, if $B \in \mathrm{P}$ and f is a polynomial then $\mathrm{count}_B^{=f} \in \#\mathrm{P}$.

Proposition 2.1 *For every function $h : \Sigma^* \to [0,1]$ the following statements are equivalent:*

1. *There exist a polynomial q and a probabilistic machine M that runs exactly $q(|x|)$ steps on input x such that $\mathrm{prob}_M(x) = h(x)$ for all $x \in \Sigma^*$.*
2. *There exist a polynomial q and a $B \in \mathrm{P}$ such that $\mathrm{count}_B^{=q}(x) = h(x) \cdot 2^{q(|x|)}$ for all $x \in \Sigma^*$.*

Proof: The claim follows immediately from the definition of probabilistic machines. □

Starting from existing complexity classes one can define new one with the help of so-called *operators*. We will introduce here some of these operators:

Definition 2.2 *Let $\mathcal{C}$ be a complexity class.*

- *We say $A \in \exists{\cdot}\mathcal{C}$ if there is a $B \in \mathcal{C}$ and a polynomial p such that for all $x \in \Sigma^*$:*

$$x \in A \Leftrightarrow \mathrm{count}_B^{=p}(x) \geq 1$$

- *We say $A \in \forall{\cdot}\mathcal{C}$ if there is a $B \in \mathcal{C}$ and a polynomial p such that for all $x \in \Sigma^*$:*

$$x \in A \Leftrightarrow \mathrm{count}_B^{=p}(x) = 2^{p(|x|)}$$

- *We say $A \in \mathrm{BP}{\cdot}\mathcal{C}$ if there is a $B \in \mathcal{C}$, a polynomial p, and an $\varepsilon > 0$ such that for all $x \in \Sigma^*$ the following holds:*

$$\begin{aligned} x \in A &\implies \mathrm{count}_B^{=p} > (\frac{1}{2} + \varepsilon) \cdot 2^{p(|x|)} \\ x \notin A &\implies \mathrm{count}_B^{=p} < (\frac{1}{2} - \varepsilon) \cdot 2^{p(|x|)} \end{aligned}$$

- *We say $A \in \mathrm{U}{\cdot}\mathcal{C}$ if there is a $B \in \mathcal{C}$ and a polynomial p such that for all $x \in \Sigma^*$ the following holds:*

$$\begin{aligned} x \in A &\implies \mathrm{count}_B^{=p} = 1 \\ x \notin A &\implies \mathrm{count}_B^{=p} = 0 \end{aligned}$$

By considering Proposition 2.1 and inserting a $B \in \mathrm{P}$ in the above definition, it is obvious that $\exists{\cdot}\mathrm{P} = \mathrm{NP}$ and $\forall{\cdot}\mathrm{P} = co\mathrm{NP}$. The BP$\cdot$ operator was introduced by Schöning [Sch89] generalizing the idea of Gill's class $\mathrm{BPP} = \mathrm{BP}{\cdot}\mathrm{P}$ [Gil77]. In the definition of $\mathrm{BP}{\cdot}\mathcal{C}$ there is the gap $[(\frac{1}{2} - \varepsilon) \cdot 2^{p(|x|)}, (\frac{1}{2} + \varepsilon) \cdot 2^{p(|x|)}]$, the value of the function $\mathrm{count}_B^{=p}$ must never belong to. We have already seen, that there is a strong correspondence between the count function and the probability of acceptance of a probabilistic machine. From this correspondence it is evident, that a set that can be defined with a large gap can be decided by a machine that works very accurate, i.e. that yields a correct result with high probability. Therefore we are interested in ways to widen this gap; this is possible if the class $\mathcal{C}$ satisfies a certain closure property.

Definition 2.3 *Let A and B be two sets.*

- *We say that A is* conjunctive truth-table reducible *to B (in notation $A \leq_{ctt}^{P} B$) if there is a function $f \in \mathrm{FP}$ such that for every $x \in \Sigma^*$ it holds $x \in A$ if and only if $f(x) = \langle x_1, \ldots, x_k \rangle$ and $c_B(x_1) = \cdots = c_B(x_k) = 1$.*
- *We say that A is* majority-reducible *to B (in notation $A \leq_{maj}^{P} B$) if there is a function $f \in \mathrm{FP}$ such that for every $x \in \Sigma^*$ it holds $x \in A$ if and only if $f = \langle x_1, \ldots, x_k \rangle$ and there is an $I \subseteq \{1, \ldots, k\}$ with $|I| > \frac{k}{2}$ and $c_B(x_i) = 1$ for all $i \in I$.*

Proposition 2.4 *(Amplification [Sch89]) If $\mathcal{C}$ is closed under $\leq_{maj}^{P}$ then for all $A \in \mathrm{BP}{\cdot}\mathcal{C}$ and all polynomials p there is a $B \in \mathcal{C}$ and a polynomial q such that*

$$\begin{aligned} x \in A &\implies \mathrm{count}_B^{=q} > (1 - 2^{-p(|x|)}) \cdot 2^{q(|x|)} \textit{ and} \\ x \notin A &\implies \mathrm{count}_B^{=q} < 2^{-p(|x|)} \cdot 2^{q(|x|)}. \end{aligned}$$

Since P is obviously closed under $\leq_{maj}^{P}$ we can give the following definition of $\mathrm{BP}{\cdot}\mathrm{P}$ that coincides with Gill's class BPP.

Definition 2.5 *A set A is in* $\mathrm{BPP} = \mathrm{BP}{\cdot}\mathrm{P}$ *if there is a $B \in \mathrm{P}$, a polynomial p, and an $\varepsilon > 0$ such that*

$$\begin{aligned} x \in A &\implies \mathrm{count}_B^{=p} > (\frac{1}{2} + \varepsilon) \cdot 2^{p(|x|)} \text{ and} \\ x \notin A &\implies \mathrm{count}_B^{=p} < (\frac{1}{2} - \varepsilon) \cdot 2^{p(|x|)}. \end{aligned}$$

We already mentioned the equivalence between the number of paths of balanced machines and their probability of acceptance. In a balanced machine each path of computation has the same probability so that we can determine whether or not an input x is accepted by counting the number of paths and dividing the result by the total number of paths of the machine. In an unbalanced machine we have shorter paths and longer paths and the shorter a path is, the more probable the machine will choose this path. It is easy to see that the above definition of BPP could be given equivalently using unbalanced probabilistic machines as follows: A set A is in BPP if there is an unbalanced probabilistic machine M that runs for at most p steps, where p is a polynomial, and an $\varepsilon > 0$ such that

$$\begin{aligned} x \in A &\implies \mathrm{prob}_M(x) > \frac{1}{2} + \varepsilon \text{ and} \\ x \notin A &\implies \mathrm{prob}_M(x) < \frac{1}{2} - \varepsilon. \end{aligned}$$

Since we talk about probability, in this definition we implicitly weight the paths of the machine in such a way that short paths have higher probabilities. If we do not weight the paths and just count their number we meet the following threshold class which was introduced by Han *et al.*

Definition 2.6 ([HHT97]) *A set A is in $\mathrm{BPP}_{\mathrm{path}}$ if there exists a nondeterministic polynomial-time Turing machine M and an $\varepsilon > 0$ such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in A &\implies \mathrm{acc}_M(x) > (\frac{1}{2} + \varepsilon) \cdot \mathrm{total}_M(x) \\ x \notin A &\implies \mathrm{acc}_M(x) < (\frac{1}{2} - \varepsilon) \cdot \mathrm{total}_M(x) \end{aligned}$$

Theorem 2.7 ([HHT97]) $\mathrm{P}^{\mathrm{NP[log]}} \subseteq \mathrm{BPP}_{\mathrm{path}}$

In 1985 Babai [Bab85] introduced the so-called Arthur-Merlin hierarchy that. The classes of the hierarchy consist of sets that can be decided by an Arthur-Merlin game that works as follows: The "board" the game takes place on is a set B from P that is known to both, Arthur and Merlin. On an input x, Arthur and Merlin alternately make moves, where move i consists of outputting a string y_i of polynomial length in x. Each player can remember all of the moves that were already made. The game ends after n moves and Merlin wins if and only if $(x, y_1, \ldots, y_n) \in B$. Besides that, we require Merlin to always make optimal moves and Arthur to always make totally arbitrary moves. We say a set L can be decided by an Arthur-Merlin game if there is a $B \in \mathrm{P}$ such that for all inputs x: If x belongs to L then the probability that Merlin wins is greater than $\frac{1}{2}$ plus some constant. If x is not from L then the probability that Arthur wins has to be greater than $\frac{1}{2}$ plus some constant. Depending on who of the two makes the first move, and how many moves the game lasts, we can sort sets in M, A, MA, AM, MAM and so on, thus forming the Arthur-Merlin hierarchy. It is easy to see, that $\mathrm{A} = \mathrm{BPP}$ and that $\mathrm{M} = \mathrm{NP}$. Besides that, Babai showed that $\mathrm{MA} \subseteq \mathrm{AM}$ and that the Arthur-Merlin hierarchy collapses to AM. We will now give a formal definition of the classes MA and AM.

Definition 2.8 *[Bab85] The set L is in* MA *if there is a set $B \in \mathrm{P}$, polynomials p, q, and an $\varepsilon > 0$ such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in L &\implies \exists y(|y| = p(|x|) \wedge \mathrm{count}_B^{=q}(x,y) > (\frac{1}{2} + \varepsilon) \cdot 2^{q(|xy|)}) \\ x \notin L &\implies \forall y(|y| = p(|x|) \rightarrow \mathrm{count}_B^{=q}(x,y) < (\frac{1}{2} - \varepsilon) \cdot 2^{q(|xy|)}) \end{aligned}$$

The set L is in AM if there is a set $B \in \mathrm{P}$, polynomials p, q and an $\varepsilon > 0$ such that for all $x \in \Sigma^*$ the following holds:

$$x \in L \implies \#\{y : |y| = q(|x|) \wedge \exists z(|z| = p(|x|) \wedge (x, y, z) \in B)\} > (\frac{1}{2} + \varepsilon) \cdot 2^{q(|x|)}$$
$$x \notin L \implies \#\{y : |y| = q(|x|) \wedge \exists z(|z| = p(|x|) \wedge (x, y, z) \in B)\} < (\frac{1}{2} - \varepsilon) \cdot 2^{q(|x|)}$$

It is obvious that AM coincides with $\mathrm{BP}{\cdot}\mathrm{NP}$ but MA does not seem to be the same as $\exists{\cdot}\mathrm{BPP}$: There exists an oracle A with $\mathrm{MA}^A \neq \exists{\cdot}\mathrm{BPP}^A$ [FFKL93]. As well as BPP, both AM and MA can be amplified:

Proposition 2.9 $L \in \mathrm{MA}$ *if and only if there exists a polynomial p such that for every polynomial $r > 1$ there exists a set $B \in \mathrm{P}$ and a polynomial q with*

$$x \in L \implies \exists y(|y| = p(|x|) \wedge \mathrm{count}_B^{=q}(x, y) > (1 - 2^{-r(|xy|)}) \cdot 2^{q(|xy|)}) \textit{ and}$$
$$x \notin L \implies \forall y(|y| = p(|x|) \rightarrow \mathrm{count}_B^{=q}(x, y) < 2^{-r(|xy|)} \cdot 2^{q(|xy|)}).$$

$L \in \mathrm{AM}$ *if and only if for every polynomial $r > 1$ there is a set $B \in \mathrm{P}$ and polynomials p, q such that for all $x \in \Sigma^*$:*

$$x \in L \implies \#\{y : |y| = p(|x|) \wedge \exists z(|z| = q(|x|) \wedge (x, y, z) \in B)\} > (1 - 2^{-r(|x|)}) \cdot 2^{p(|x|)}$$
$$x \notin L \implies \#\{y : |y| = p(|x|) \wedge \exists z(|z| = q(|x|) \wedge (x, y, z) \in B)\} < 2^{-r(|x|)} \cdot 2^{p(|x|)}$$

3 The Class SBP

The class SBP is similar to BPP. Again the idea is that of a probabilistic machine with a probability gap, i.e., a machine whose acceptance probability never falls into a certain forbidden interval. We want such a machine to accept an input either with probability less than some a or with probability greater than some b, where $0 \leq a < b \leq 1$. But whereas in the definition of BPP the probability gap defined by a and b forms a constant interval around $\frac{1}{2}$, an SBP machine has a probability gap around some smaller limit which is negatively exponential in the length of the input.

Definition 3.1 *The set A is in SBP if there exists an $\varepsilon > 0$, a $B \in \mathrm{P}$ and polynomials p, q such that for all $x \in \Sigma^*$:*

$$x \in A \implies \mathrm{count}_B^{=q} > (1 + \varepsilon) \cdot \frac{2^{q(|x|)}}{2^{p(|x|)}}$$
$$x \notin A \implies \mathrm{count}_B^{=q} < (1 - \varepsilon) \cdot \frac{2^{q(|x|)}}{2^{p(|x|)}}$$

This definition leads to a class that seems to be considerably more powerful than BPP.

3.1 Properties of SBP

In this chapter we will have a look at the classes around SBP and we will integrate it into known hierarchies. Before that we will discuss basic properties of SBP and alternative characterizations.

Just as BPP, we can amplify SBP but in comparison with the amplification lemmas we saw until now, this proposition does not increase the absolute size of the probability gap. It just diminishes the probability of failure of an SBP machine for the negative case. As a consequence the *relative size* of the probability gap w.r.t. this failure probability increases. So we can replace every SBP machine by another one that has a very low probability of failure on inputs that should be rejected.

Proposition 3.2 (Amplification) *$A \in \mathrm{SBP}$ if and only if for every polynomial $r > 0$ there exist a $B \in \mathrm{P}$ and polynomials q, s such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in A &\implies \mathrm{count}_B^{=q}(x) > 2^{r(|x|)} \cdot \frac{2^{q(|x|)}}{2^{s(|x|)}} \\ x \notin A &\implies \mathrm{count}_B^{=q}(x) < \frac{1}{2^{r(|x|)}} \cdot \frac{2^{q(|x|)}}{2^{s(|x|)}} \end{aligned}$$

Proof: We start with the implication from right to left. Choose the constant polynomial $r(n) \stackrel{df}{=} 1$ and let B, q, s be the witnesses of the right-hand side. For $\varepsilon \stackrel{df}{=} 1/2$ we obtain $L \in \mathrm{SBP}$ since for all $x \in \Sigma^*$:

$$\begin{aligned} x \in A &\implies \mathrm{count}_B^{=q}(x) > (1+\varepsilon) \cdot \frac{2^{q(|x|)}}{2^{s(|x|)}} \\ x \notin A &\implies \mathrm{count}_B^{=q}(x) < (1-\varepsilon) \cdot \frac{2^{q(|x|)}}{2^{s(|x|)}} \end{aligned}$$

For the other direction let $r > 0$ be a polynomial, $A \in \mathrm{SBP}$ and B, p, q, ε as in the definition of SBP. Surely there is a $k > 0$ such that $(1+\varepsilon)^k \geq 2$ and $(1-\varepsilon)^k \leq \frac{1}{2}$. Let

$$B' =_{\mathrm{def}} \{(x,y) : y = y_1 \cdots y_{k \cdot r(|x|)} \text{ with } |y_i| = q(|x|) \text{ and } (x, y_i) \in B \text{ for } 1 \leq i \leq k \cdot r(|x|)\}$$

and observe that $B \in \mathrm{P}$. Moreover, with $q'(n) \stackrel{df}{=} k \cdot q(n) \cdot r(n)$ and $p'(n) \stackrel{df}{=} k \cdot p(n) \cdot r(n)$ we get:

$$\begin{aligned} x \in A &\implies \mathrm{count}_{B'}^{=q'}(x) = \left(\mathrm{count}_B^{=q}(x)\right)^{k \cdot r(|x|)} > \left((1+\varepsilon) \cdot \frac{2^{q(|x|)}}{2^{p(|x|)}}\right)^{k \cdot r(|x|)} \geq 2^{r(|x|)} \cdot \frac{2^{q'(|x|)}}{2^{p'(|x|)}} \\ x \notin A &\implies \mathrm{count}_{B'}^{=q'}(x) = \left(\mathrm{count}_B^{=q}(x)\right)^{k \cdot r(|x|)} < \left((1-\varepsilon) \cdot \frac{2^{q(|x|)}}{2^{p(|x|)}}\right)^{k \cdot r(|x|)} \leq \frac{1}{2^{r(|x|)}} \cdot \frac{2^{q'(|x|)}}{2^{p'(|x|)}} \end{aligned}$$

□

Proposition 3.3 *The following statements are equivalent for every $A \subseteq \Sigma^*$.*

1. $A \in \mathrm{SBP}$
2. *There exist polynomials p, q, some $\varepsilon > 0$ and a probabilistic machine M running exactly $q(|x|)$ steps on input x, such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in A &\implies \mathrm{prob}_M(x) > (1+\varepsilon) \cdot 2^{-p(|x|)} \\ x \notin A &\implies \mathrm{prob}_M(x) < (1-\varepsilon) \cdot 2^{-p(|x|)} \end{aligned}$$

3. *There exists an $f \in \#\mathrm{P}$ and a polynomial q such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in A &\implies f(x) > (1+\varepsilon) \cdot 2^{q(|x|)} \\ x \notin A &\implies f(x) < (1-\varepsilon) \cdot 2^{q(|x|)} \end{aligned}$$

4. *There exist $f \in \#\mathrm{P}$, $g \in \mathrm{FP}$ and $\varepsilon > 0$ such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in A &\implies f(x) > (1+\varepsilon) \cdot g(x) \\ x \notin A &\implies f(x) < (1-\varepsilon) \cdot g(x) \end{aligned}$$

5. *For every polynomial $r > 0$ there exist $B \in \mathrm{P}$ and polynomials s, t such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in A &\implies \mathrm{count}_B^{=t}(x) > 2^{r(|x|)} \cdot \frac{2^{t(|x|)}}{2^{s(|x|)}} \\ x \notin A &\implies \mathrm{count}_B^{=t}(x) < \frac{2^{t(|x|)}}{2^{s(|x|)}} \end{aligned}$$

6. *For every* $h \in \mathrm{FP}$ *with* $h > 1$ *there exist* $f \in \#\mathrm{P}$, $g \in \mathrm{FP}$ *such that for all* $x \in \Sigma^*$*:*

$$\begin{aligned} x \in A &\implies f(x) > h(x) \cdot g(x) \\ x \notin A &\implies f(x) < g(x) \end{aligned}$$

Proof:

The equivalence of the points 1, 2, and 3 is evident from the definition of SBP, probabilistic machines, $\#\mathrm{P}$, and Proposition 2.1. Obviously, if point 3 holds, then point 4 does so.

Assume now that A satisfies point 4 of the proposition; we will show that this implies point 3. Note that we can assume $\varepsilon < 1$. Since $f \in \#\mathrm{P}$ there exists a polynomial $p > 0$ and some $B \in \mathrm{P}$ such that $f(x) = \mathrm{count}_B^{=p}(x)$ for all $x \in \Sigma^*$. In order to prevent that the value of g vanishes we define the following normalizations for $x \in \Sigma^*$.

$$g'(x) \stackrel{df}{=} \begin{cases} 1 & : \text{ if } g(x) = 0 \\ g(x) & : \text{ otherwise} \end{cases}$$

$$f'(x) \stackrel{df}{=} \begin{cases} 2 & : \text{ if } g(x) = 0 \\ f(x) & : \text{ otherwise} \end{cases}$$

Note that $g' \in \mathrm{FP}$ and $f' \in \#\mathrm{P}$. Moreover, observe that if $g(x) = 0$ then $x \in A$. Therefore, we obtain that f' and g' achieve the same as f and g in the following sense:

$$\begin{aligned} x \in A &\implies f'(x) > (1+\varepsilon) \cdot g'(x) \\ x \notin A &\implies f'(x) < (1-\varepsilon) \cdot g'(x) \end{aligned}$$

Choose a polynomial q such that $2^{q(n)} \cdot \varepsilon/2 > 2^{p(n)}$ for all $n \geq 0$. Let $h(x) \stackrel{df}{=} \lfloor 2^{q(|x|)}/g'(x) \rfloor \cdot f'(x)$ and note that $h \in \#\mathrm{P}$. Now observe the following implications.

$$\begin{aligned} x \in A \implies \frac{f'(x)}{g'(x)} > (1+\varepsilon) &\implies h(x) > 2^{q(|x|)} \cdot \frac{f'(x)}{g'(x)} - f'(x) > 2^{q(|x|)} \cdot (1+\varepsilon) - 2^{p(|x|)} \\ &\qquad > (1+\varepsilon/2) \cdot 2^{q(|x|)} \\ x \notin A \implies \frac{f'(x)}{g'(x)} < (1-\varepsilon) &\implies h(x) \leq 2^{q(|x|)} \cdot \frac{f'(x)}{g'(x)} < (1-\varepsilon) \cdot 2^{q(|x|)} < (1-\varepsilon/2) \cdot 2^{q(|x|)} \end{aligned}$$

It follows that A satisfies point 3. So we proved that the points 1–4 are equivalent.

By Proposition 3.2, point 1 implies point 5. Note that for every $h \in \mathrm{FP}$ there is a polynomial r such that $2^{r(n)} \geq h(n)$ so point 6 follows directly from point 5. It remains to show that point 6 implies point 4. For this we choose $h(x) \stackrel{df}{=} 3$ and get:

$$\begin{aligned} x \in A &\implies f(x) > 3 \cdot g(x) = (1 + \frac{1}{2}) \cdot 2g(x) \\ x \notin A &\implies f(x) < g(x) = (1 - \frac{1}{2}) \cdot 2g(x) \end{aligned}$$

□

If we generalize the characterization of SBP that is given in Proposition 3.3.4 by using not a $\#\mathrm{P}$ and an FP function but two $\#\mathrm{P}$ functions we get a larger class that, as we will see later, coincides with the threshold class $\mathrm{BPP}_{\mathrm{path}}$.

Closure properties of complexity classes are another point of interest. It is known that BPP is closed under union, intersection, and complement. We cannot show SBP to be likewise robust: We will see that there is an oracle where $\mathrm{SBP} \neq \mathrm{coSBP}$. Besides that it remains open whether or not SBP is closed under intersection (we even do not know whether there is an oracle where SBP is not closed under intersection). However, we can prove that it is closed under union:

Proposition 3.4 SBP *is closed under* $\cup$.

Proof: By Proposition 3.3.6, for $A_1, A_2 \in \mathrm{SBP}$ there exist $f_1, f_2 \in \#\mathrm{P}$ and $g_1, g_2 \in \mathrm{FP}$ such that

$$\begin{aligned} x \in A_1 &\implies f_1(x) > 4 \cdot g_1(x), && (1)\\ x \notin A_1 &\implies f_1(x) < g_1(x), \\ x \in A_2 &\implies f_2(x) > 4 \cdot g_2(x), \text{ and} && (2)\\ x \notin A_2 &\implies f_2(x) < g_2(x). \end{aligned}$$

Multiplying equation (1) with g_2 and equation (2) with g_1 leads to

$$\begin{aligned} x \in A_1 &\implies f_1(x)g_2(x) > 4 \cdot g_1(x)g_2(x), \\ x \notin A_1 &\implies f_1(x)g_2(x) < g_1(x)g_2(x), \\ x \in A_2 &\implies f_2(x)g_1(x) > 4 \cdot g_1(x)g_2(x), \text{ and} \\ x \notin A_2 &\implies f_2(x)g_1(x) < g_1(x)g_2(x). \end{aligned}$$

Hence we can conclude for $\varepsilon = 1/3$, $f(x) = f_1(x) \cdot g_2(x) + f_2(x) \cdot g_1(x)$ and $g(x) = 3 \cdot g_1(x) \cdot g_2(x)$:

$$\begin{aligned} x \in A_1 \cup A_2 &\implies f(x) > (1+\varepsilon) \cdot g(x) \\ x \notin A_1 \cup A_2 &\implies f(x) < (1-\varepsilon) \cdot g(x) \end{aligned}$$

Obviously $f \in \#\mathrm{P}$ and $g \in \mathrm{FP}$ and therefore $A_1 \cup A_2 \in \mathrm{SBP}$ by Proposition 3.3.4. □

3.2 SBP is between MA and AM

In this subsection we will fit SBP in already known hierarchies. In particular we will show that it fits in Babai's Arthur-Merlin hierarchy between MA and AM.

Theorem 3.5 $\mathrm{MA} \subseteq \mathrm{SBP}$

Proof: Let $L \in \mathrm{MA}$. By Proposition 2.9, there exist a polynomial p such that for $s(n) \stackrel{df}{=} n + 2$ there exist a set $B \in \mathrm{P}$ and a polynomial q with:

$$\begin{aligned} x \in L &\implies \exists y(|y| = p(|x|) \text{ and } \mathrm{count}_B^{=q}(x,y) > (1 - 2^{-s(|xy|)}) \cdot 2^{q(|xy|)}) \\ x \notin L &\implies \forall y(|y| = p(|x|) \to \mathrm{count}_B^{=q}(x,y) < 2^{-s(|xy|)} \cdot 2^{q(|xy|)}) \end{aligned}$$

Let $\varepsilon \stackrel{df}{=} 1/2$, $q'(n) \stackrel{df}{=} p(n) + q(n + p(n))$, $p'(n) \stackrel{df}{=} p(n) + 1$ and

$$B' \stackrel{df}{=} \{(x,y) : y = y_1 y_2 \wedge y_1 \in \Sigma^{p(|x|)} \wedge y_2 \in \Sigma^{q(|xy_1|)} \wedge (x, y_1, y_2) \in B\}.$$

Then the following holds.

$$\begin{aligned} x \in L \implies \mathrm{count}_{B'}^{=q'}(x) &> (1 - 2^{-s(|x|+p(|x|))}) \cdot 2^{q(|x|+p(|x|))} \\ &= (1 - 2^{-2-|x|-p(|x|)}) \cdot 2^{q'(|x|)-p(|x|)} \\ &\geq \frac{3}{4} \cdot 2^{q'(|x|)-p(|x|)} = (1+\varepsilon) \cdot 2^{q'(|x|)-p'(|x|)} \\ x \notin L \implies \mathrm{count}_{B'}^{=q'}(x) &< 2^{p(|x|)} \cdot 2^{-s(|x|+p(|x|))} \cdot 2^{q(|x|+p(|x|))} \\ &= 2^{-2-|x|-p(|x|)} \cdot 2^{q'(|x|)} \\ &\leq \frac{1}{4} \cdot 2^{q'(|x|)-p(|x|)} = (1-\varepsilon) \cdot 2^{q'(|x|)-p'(|x|)} \end{aligned}$$

Since $B' \in \mathrm{P}$ this shows $L \in \mathrm{SBP}$. □

To show that SBP is a subset of AM we need the following definitions: A *linear hash function* $h : \Sigma^m \longrightarrow \Sigma^k$ is given by a Boolean (k, m)-matrix M. A string $x = x_1 \cdots x_m$ is mapped to a string $y = y_1 \cdots y_k$ if and only if $y = M \cdot x^T$ (here we mean the inner product modulo 2). We adopt notations from [KW94] and define for a set $X \subseteq \Sigma^m$ and a family of hash functions $H = \{h_1, \ldots, h_l\}$ the predicate Collision as

$$\mathrm{Collision}(X, H) \overset{df}{\Longleftrightarrow} (\exists x \in X)(\exists y_1, \ldots, y_l \in X \setminus \{x\})(\forall i : 1 \leq i \leq l)[h_i(x) = h_i(y_i)].$$

If $\mathrm{Collision}(X, H)$ then we say that X has a collision w.r.t. H. The set of all families $H = \{h_1, \ldots, h_l\}$ of l linear hash functions from Σ^m to Σ^k is denoted by $\mathcal{H}(l, m, k)$. In 1983 Sipser proved the following useful theorems about linear hash functions.

Theorem 3.6 ([Sip83, Coding Lemma]) *Let $X \subseteq \Sigma^m$ be a set of cardinality at most 2^{k-1}. If we choose a hash family H uniformly at random from $\mathcal{H}(k, m, k)$, then the probability that X has a collision w.r.t. H is at most $1/2$.*

So if the set X is not to big then collision does not occur to often. An easy pigeon-hole argument shows that if X contains slightly more elements then collisions occur with probability 1.

Theorem 3.7 ([Sip83]) *For any hash family $H \in \mathcal{H}(k, m, k)$ and any set $X \subseteq \Sigma^m$ of cardinality $|X| > k \cdot 2^k$, X must have a collision w.r.t. H.*

Theorem 3.8 $\mathrm{SBP} \subseteq \mathrm{BP}{\cdot}\mathrm{NP} = \mathrm{AM}$

Proof: Let $L \in \mathrm{SBP}$. By Proposition 3.2 there exist some $B \in \mathrm{P}$ and polynomials p, q such that for all $x \in \Sigma^*$:

$$\begin{aligned} x \in L &\implies \mathrm{count}_B^{=q}(x) > 2^{|x|+1} \cdot \frac{2^{q(|x|)}}{2^{p(|x|)}} \\ x \notin L &\implies \mathrm{count}_B^{=q}(x) < \frac{2^{q(|x|)}}{2^{p(|x|)}} \end{aligned}$$

With the following set D we can test whether a given family of hash functions H can hash all witnesses of a given x (i.e., all y with $(x, y) \in B$) without collisions.

$$D \overset{df}{=} \Big\{(x, H) \,:\, x \in \Sigma^*,\, H \in \mathcal{H}(k, m, k) \text{ and } \mathrm{Collision}(X, H) \text{ for} \tag{3}$$

- $k \overset{df}{=} q(|x|) - p(|x|) + 1$
- $m \overset{df}{=} q(|x|)$
- $X \overset{df}{=} \{y \,:\, |y| = q(|x|) \text{ and } (x, y) \in B\}\Big\}$

From the definition of $\mathrm{Collision}(X, H)$ it is easy to see that $D \in \mathrm{NP}$. Now we consider an arbitrary word x that is sufficiently long, i.e., long enough such that $2^{|x|} > q(|x|) - p(|x|) + 1$. Define k, m and X as in equation (3). We consider two cases:

- $x \in L$: Then $|X| \geq 2^{q(|x|) - p(|x|) + |x| + 1} = 2^{k + |x|} > k \cdot 2^k$ since x was chosen long enough. From Theorem 3.7 it follows that $\mathrm{Collision}(X, H)$ for all $H \in \mathcal{H}(k, m, k)$.
- $x \notin L$: Then $|X| \leq 2^{q(|x|) - p(|x|)} = 2^{k-1}$ and from Theorem 3.6 it follows that

$$\frac{\#\{H \,:\, H \in \mathcal{H}(k, m, k) \text{ and } \mathrm{Collision}(X, H)\}}{\#\{H \in \mathcal{H}(k, m, k)\}} \leq \frac{1}{2}.$$

So we obtain:

$$
\begin{array}{rcl}
x \in L & \Longrightarrow & \dfrac{\#\{H \in \mathcal{H}(k,m,k) \,:\, (x,H) \in D\}}{\#\{H \in \mathcal{H}(k,m,k)\}} = 1 \\
x \notin L & \Longrightarrow & \dfrac{\#\{H \in \mathcal{H}(k,m,k) \,:\, (x,H) \in D\}}{\#\{H \in \mathcal{H}(k,m,k)\}} \leq \dfrac{1}{2}
\end{array}
$$

Since $D \in \mathrm{NP}$ this shows $L \in \mathrm{BP}\cdot\mathrm{NP}$. □

We are now able to fix the position of SBP:

Corollary 3.9 $\exists\cdot\mathrm{BPP} = \mathrm{NP}^{\mathrm{BPP}} \subseteq \mathrm{MA} \subseteq \mathrm{SBP} \subseteq \mathrm{AM} = \mathrm{BP}\cdot\mathrm{NP}$.

Proof: $\exists\cdot\mathrm{BPP} = \mathrm{NP}^{\mathrm{BPP}}$ follows immediately by the selflowness of BPP [Ko82, Zac82], and the remaining claims follow from the definitions of MA and AM and from theorems 3.5 and 3.8. □

In Proposition 3.3.4 we characterized SBP using a #P function and an FP function. The natural question arises what would happen if we defined a class in a very similar way but using two #P functions this time. We show now that this leads exactly to the threshold class $\mathrm{BPP}_{\mathrm{path}}$.

Proposition 3.10 *$L \in \mathrm{BPP}_{\mathrm{path}}$ if and only if there exist $f, g \in \#\mathrm{P}$ and $\varepsilon > 0$ such that for all $x \in \Sigma^*$:*

$$
\begin{array}{rcl}
x \in L & \Longrightarrow & f(x) > (1+\varepsilon) \cdot g(x) \\
x \notin L & \Longrightarrow & f(x) < (1-\varepsilon) \cdot g(x)
\end{array}
$$

Proof: Let $L \in \mathrm{BPP}_{\mathrm{path}}$ and choose M and ε as in Definition 2.6. Then the following is easy to see.

$$
\begin{array}{l}
x \in L \implies 2 \cdot \mathrm{acc}_M(x) > (1+2\varepsilon)(\mathrm{acc}_M(x) + \mathrm{rej}_M(x)) \implies \mathrm{acc}_M(x) > (1+2\varepsilon) \cdot \mathrm{rej}_M(x) \\
x \notin L \implies 2 \cdot \mathrm{acc}_M(x) < (1-2\varepsilon)(\mathrm{acc}_M(x) + \mathrm{rej}_M(x)) \implies \mathrm{acc}_M(x) < (1-2\varepsilon) \cdot \mathrm{rej}_M(x)
\end{array}
$$

Since the functions acc_M and rej_M are in #P this shows that the proposition holds from left to right.

Now assume that we are given a language L satisfying the right-hand side of the proposition. Note that without loss of generality we may assume $\varepsilon < 1$. Since $f, g \in \#\mathrm{P}$ there exist nondeterministic polynomial-time Turing machines N_1, N_2 with $\mathrm{acc}_{N_1}(x) = f(x)$ and $\mathrm{acc}_{N_2}(x) = g(x)$ for all $x \in \Sigma^*$. Let p be a polynomial bounding the computation time of both machines N_1 and N_2. Choose a polynomial q large enough such that $2^{q(n)} \cdot \varepsilon/4 > 2^{p(n)+1}$ for all $n \geq 0$.

Let M denote a nondeterministic polynomial-time Turing machine working as follows on input x: First of all, M produces two paths while making one nondeterministic step. On the first (resp., second) path M simulates N_1 (resp., N_2) on input x. Each time this simulation ends with a rejecting path, M makes one more nondeterministic step in order to produce one accepting and one rejecting path. If the simulation of N_1 (resp., N_2) ends with an accepting path then M makes $q(|x|)$ additional nondeterministic steps in order to produce $2^{q(|x|)}$ accepting (resp., rejecting) paths.

In the remaining part of the proof we will show that M accepts L in the sense of $\mathrm{BPP}_{\mathrm{path}}$. From the definition of M we get the following estimations for acc_M and rej_M.

$$
2^{q(|x|)} \cdot f(x) \leq \mathrm{acc}_M(x) \leq 2^{q(|x|)} \cdot f(x) + 2^{p(|x|)+1} \tag{4}
$$

$$
2^{q(|x|)} \cdot g(x) \leq \mathrm{rej}_M(x) \leq 2^{q(|x|)} \cdot g(x) + 2^{p(|x|)+1} \tag{5}
$$

If $x \in L$ then $f(x) > g(x) \cdot (1+\varepsilon)$ and therefore $f(x) \geq 1$. Since $\varepsilon < 1$ we have $1/(1+\varepsilon) \leq 1 - \varepsilon/2$ and $f(x) \cdot (1-\varepsilon/2) \geq g(x)$. So we obtain:

$$
\begin{array}{rcll}
\mathrm{rej}_M(x) & \leq & 2^{q(|x|)} \cdot g(x) + 2^{p(|x|)+1} & \\
 & \leq & 2^{q(|x|)} \cdot f(x) \cdot (1-\varepsilon/2) + 2^{p(|x|)+1} & \\
 & \leq & 2^{q(|x|)} \cdot f(x) \cdot (1-\varepsilon/4) - 2^{q(|x|)} \cdot \varepsilon/4 + 2^{p(|x|)+1} & \text{(since } f(x) \geq 1) \\
 & < & 2^{q(|x|)} \cdot f(x) \cdot (1-\varepsilon/4) & \text{(by the choice of } q) \\
 & \leq & (1-\varepsilon/4) \cdot \mathrm{acc}_M(x) & \text{(by equation (4))} \qquad (6)
\end{array}
$$

If $x \notin L$ then $f(x) < (1-\varepsilon) \cdot g(x)$ and therefore $g(x) \geq 1$. In this case we get:

$$
\begin{array}{rcll}
\mathrm{acc}_M(x) & \leq & 2^{q(|x|)} \cdot f(x) + 2^{p(|x|)+1} & \\
 & < & 2^{q(|x|)} \cdot g(x) \cdot (1-\varepsilon) + 2^{p(|x|)+1} & \\
 & \leq & 2^{q(|x|)} \cdot g(x) \cdot (1-\varepsilon/2) - 2^{q(|x|)} \cdot \varepsilon/2 + 2^{p(|x|)+1} & \text{(since } g(x) \geq 1\text{)} \\
 & < & 2^{q(|x|)} \cdot g(x) \cdot (1-\varepsilon/2) & \text{(by the choice of } q\text{)} \\
 & \leq & (1-\varepsilon/4) \cdot \mathrm{rej}_M(x) & \text{(by equation (5))} \qquad (7)
\end{array}
$$

Observe that inequality (6) implies $\mathrm{rej}_M(x) \leq \mathrm{total}_M(x)/2$ and inequality (7) implies $\mathrm{acc}_M(x) \leq \mathrm{total}_M(x)/2$. Therefore, if we add $(1-\varepsilon/4) \cdot \mathrm{rej}_M(x)$ (resp., $(1-\varepsilon/4) \cdot \mathrm{acc}_M(x)$) to both sides of inequality (6) (resp., inequality (7)) we get:

$$
\begin{array}{rcl}
x \in L \implies (2-\frac{\varepsilon}{4}) \cdot \mathrm{rej}_M(x) < (1-\frac{\varepsilon}{4}) \cdot \mathrm{total}_M(x) & \implies & 2 \cdot \mathrm{rej}_M(x) < (1-\frac{\varepsilon}{8}) \cdot \mathrm{total}_M(x) \\
 & \implies & \mathrm{rej}_M(x) < (\frac{1}{2} - \frac{\varepsilon}{16}) \cdot \mathrm{total}_M(x) \\
 & \implies & \mathrm{acc}_M(x) > (\frac{1}{2} + \frac{\varepsilon}{16}) \cdot \mathrm{total}_M(x) \\
x \notin L \implies (2-\frac{\varepsilon}{4}) \cdot \mathrm{acc}_M(x) < (1-\frac{\varepsilon}{4}) \cdot \mathrm{total}_M(x) & \implies & 2 \cdot \mathrm{acc}_M(x) < (1-\frac{\varepsilon}{8}) \cdot \mathrm{total}_M(x) \\
 & \implies & \mathrm{acc}_M(x) < (\frac{1}{2} - \frac{\varepsilon}{16}) \cdot \mathrm{total}_M(x)
\end{array}
$$

This shows $L \in \mathrm{BPP}_{\mathrm{path}}$ and it follows that the implication from right to left holds. □

This result enables us to precise the position of SBP.

Corollary 3.11 $\mathrm{BPP} \subseteq \mathrm{SBP} \subseteq \mathrm{BPP}_{\mathrm{path}}$

Proof: This is an immediate consequence from Corollary 3.9 and the Propositions 3.3.4 and 3.10. □

We provide a picture of the mentioned classes'sinclusion structure, that is established when we take the above results into account, at the end of section 4.

4 Relations to Other Classes

In 1976 Valiant [Val76] introduced the class UP of languages that are decidable in unambiguous polynomial-time. This means that UP consists of all languages that can be accepted in polynomial-time by a nondeterministic machine satisfying the promise that each computation has at most one accepting path. Equivalently, $L \in \mathrm{UP}$ if and only if there exists some $f \in \#\mathrm{P}$ such that for all $x \in \Sigma^*$:

$$
\begin{array}{rcl}
x \in L & \implies & f(x) = 1 \\
x \notin L & \implies & f(x) = 0
\end{array}
$$

If one weakens this definition and asks for some $f \in \mathrm{GapP}$ one meets the GapP counterpart of UP, the class SPP (*stoic* PP because the machine doesn't change its behavior much between accept and reject). It was introduced in 1991 independently by Fenner *et al.* [FFK94], Gupta [Gup91] (under the name ZUP), and Ogiwara and Hemachandra [OH93] (under the name XP).

Definition 4.1 ([FFK94, Gup91, OH93]) *The class* SPP *consists of all languages* $L \subseteq \Sigma^*$ *for which there exists an* $f \in \mathrm{GapP}$ *such that for all* $x \in \Sigma^*$*:*

$$
\begin{array}{rcl}
x \in L & \implies & f(x) = 1 \\
x \notin L & \implies & f(x) = 0
\end{array}
$$

Theorem 4.2 ([FFK94]) $\mathrm{Few} \subseteq \mathrm{SPP}$

In [FFK94] it is shown that SPP is exactly the class of languages that are low for GapP. Moreover, SPP is closed under polynomial-time Turing reductions (i.e., is closed in particular under union, intersection and complementation) [FFK94].

A relaxation of the definition above leads to WPP (*wide*-PP), a class which was introduced in 1991 by Fenner *et al.* [FFK94].

Definition 4.3 ([FFK94]) *The class* WPP *consists of all languages* $L \subseteq \Sigma^*$ *for which there exist an* $f \in \mathrm{GapP}$ *and a* $g \in \mathrm{FP}$ *with* $g > 0$ *such that for all* $x \in \Sigma^*$*:*

$$\begin{array}{lcl} x \in L & \Longrightarrow & f(x) = g(x) \\ x \notin L & \Longrightarrow & f(x) = 0 \end{array}$$

Proposition 4.4 $\mathrm{SPP} \subseteq \mathrm{WPP}$

Neither it is known whether this inclusion is strict nor it is known whether WPP is closed under polynomial-time Turing reductions [FFK94].

Another class that came up in the context of PP-lowness is AWPP (*almost-wide* PP) introduced by Fenner *et al.* [FFKL93, FFKL]. Li [Li93a, Li93b] showed that AWPP is closed under union, intersection and complementation, and all languages from AWPP are low for PP.

The original definition of AWPP is such that the class admits amplification by definition. Recently, Fenner showed [Fen02] that a weaker definition can be used equivalently. Here we use the characterization of Fenner for the definition of AWPP. Theorem 4.7 below establishes the connection to the original definition.

Definition 4.5 *The class* AWPP *consists of all languages* $L \subseteq \Sigma^*$ *for which there exist an* $f \in \mathrm{GapP}$*, a polynomial* p *and* $\varepsilon > 0$ *such that for all* $x \in \Sigma^*$*:*

$$\begin{array}{lcl} x \in L & \Longrightarrow & (1+\varepsilon) \cdot \dfrac{2^{p(|x|)}}{2} < f(x) \leq 2^{p(|x|)} \\ x \notin L & \Longrightarrow & 0 \leq f(x) < (1-\varepsilon) \cdot \dfrac{2^{p(|x|)}}{2} \end{array}$$

Theorem 4.6 ([FFKL93, FFKL]) $\mathrm{WPP} \subseteq \mathrm{AWPP}$

Theorem 4.7 ([Fen02], amplification for AWPP**)** *The following is equivalent for* $L \subseteq \Sigma^*$.

1. $L \in \mathrm{AWPP}$

2. *There exist an* $f \in \mathrm{GapP}$ *and polynomials* p, q *with* $q > 0$ *such that for all* $x \in \Sigma^*$*:*

$$\begin{array}{lcl} x \in L & \Longrightarrow & (1+\dfrac{1}{q(|x|)}) \cdot \dfrac{2^{p(|x|)}}{2} < f(x) \leq 2^{p(|x|)} \\ x \notin L & \Longrightarrow & 0 \leq f(x) < (1-\dfrac{1}{q(|x|)}) \cdot \dfrac{2^{p(|x|)}}{2} \end{array}$$

3. *For every polynomial* $r > 0$ *there exist an* $f \in \mathrm{GapP}$ *and a polynomial* p *such that for all* $x \in \Sigma^*$*:*

$$\begin{array}{lcl} x \in L & \Longrightarrow & (1-\dfrac{1}{2^{r(|x|)}}) \cdot 2^{p(|x|)} < f(x) \leq 2^{p(|x|)} \\ x \notin L & \Longrightarrow & 0 \leq f(x) < \dfrac{1}{2^{r(|x|)}} \cdot 2^{p(|x|)} \end{array}$$

It turned out that AWPP has also interesting connections to quantum computing: The quantum class BQP (*bounded-error quantum polynomial-time*; think of this as the class of problems that can be solved efficiently by quantum computers) is contained in AWPP [FR99] and is therefore low for PP. In [BV97] it is shown that BPP is a lower bound for BQP, i.e., we have $\mathrm{BPP} \subseteq \mathrm{BQP} \subseteq \mathrm{AWPP}$. Up to now this is the best classification of BQP w.r.t. traditional complexity classes. In particular we have no evidence whether BQP is in the polynomial-time hierarchy. In this connection [GP01] constructs a relativized world where EQP (exact quantum polynomial-time) is not contained in P^{NP}. So in this world, $\mathrm{BQP} \not\subseteq \mathrm{P}^{\mathrm{NP}}$ since $\mathrm{EQP} \subseteq \mathrm{BQP}$ holds relativizable.

With the definition of APP (*amplified* PP), Li introduced another class of problems that are low for PP [Li93a].

Definition 4.8 ([Li93a, Li93b]) *The class* APP *consists of all languages $L \subseteq \Sigma^*$ such that for all polynomials r there exist $f, g \in \mathrm{GapP}$ with $f(1^n) > 0$ for $n \geq 0$ such that for all n, x with $n \geq |x|$:*

$$x \in L \implies (1 - \frac{1}{2^{r(n)}}) \cdot f(1^n) < g(x, 1^n) \leq f(1^n)$$

$$x \notin L \implies 0 \leq g(x, 1^n) < \frac{1}{2^{r(n)}} \cdot f(1^n)$$

It is known that $\mathrm{APP} \subseteq \mathrm{PP}$ and that APP is closed under polynomial-time Turing reductions (in particular it is closed under union, intersection and complementation) [Li93b]. APP and AWPP were introduced independendly and for both was independendly shown that they are low for PP. However, Fenner [Fen02] showed that $\mathrm{AWPP} \subseteq \mathrm{APP}$ thus giving another proof of the lowness of APP for PP.

Theorem 4.9 ([Fen02]) $\mathrm{AWPP} \subseteq \mathrm{APP}$

Remember that SPP can be considered as the GapP analog of UP. With the following definition we start from AWPP and define its #P analog.

Definition 4.10 *The class* WAPP *(*weak almost-wide PP*) consists of all languages $L \subseteq \Sigma^*$ for which there exist an $f \in \#\mathrm{P}$, a polynomial p and $\varepsilon > 0$ such that for all $x \in \Sigma^*$:*

$$x \in L \implies (1 + \varepsilon) \cdot \frac{2^{p(|x|)}}{2} < f(x) \leq 2^{p(|x|)}$$

$$x \notin L \implies 0 \leq f(x) < (1 - \varepsilon) \cdot \frac{2^{p(|x|)}}{2}$$

Proposition 4.11 $\mathrm{WAPP} \subseteq \mathrm{AWPP}$

Proof: This is an immediate consequence of Definition 4.5. □

It is not known whether AWPP is in the polynomial-time hierarchy and we will see in section 5 that there is a relativized world where $\mathrm{AWPP} \not\subseteq \mathrm{PH}$. However, in spite of the very similar definitions of AWPP and WAPP we can show that $\mathrm{WAPP} \subseteq \mathrm{PH}$. More precisely, WAPP is located between the classes BP·UP and SBP.

Proposition 4.12 $\mathrm{BP{\cdot}UP} \subseteq \mathrm{WAPP} \subseteq \mathrm{SBP}$.

Proof: Let $L \in \mathrm{BP{\cdot}UP}$, i.e., there exists an $f \in \#\mathrm{P}$, a polynomial p, and $\varepsilon > 0$ such that for all $x, y \in \Sigma^*$, $f(x, y) \leq 1$ and for all $x \in \Sigma^*$,

$$x \in L \implies \# \left\{ y \in \Sigma^{p(|x|)} : f(x,y) = 1 \right\} > \left(\frac{1}{2} + \varepsilon \right) \cdot 2^{p(|x|)}$$

$$x \notin L \implies \# \left\{ y \in \Sigma^{p(|x|)} : f(x,y) = 1 \right\} < \left(\frac{1}{2} - \varepsilon \right) \cdot 2^{p(|x|)}.$$

Let $g(x) \stackrel{df}{=} \#\{y \in \Sigma^{p(|x|)} : f(x,y) = 1\}$ and note that $g \in \#\mathrm{P}$ since $f(x,y) \leq 1$. It follows that $L \in \mathrm{WAPP}$.

Let $A \in \mathrm{WAPP}$ and consider definition 4.10. Multiply the right-hand sides of the implications with 2. Since $2 \cdot f(n) \in \#\mathrm{P}$ and $2^{p(n)} \in \mathrm{FP}$ we can apply proposition 3.3.4 and obtain $A \in \mathrm{SBP}$. □

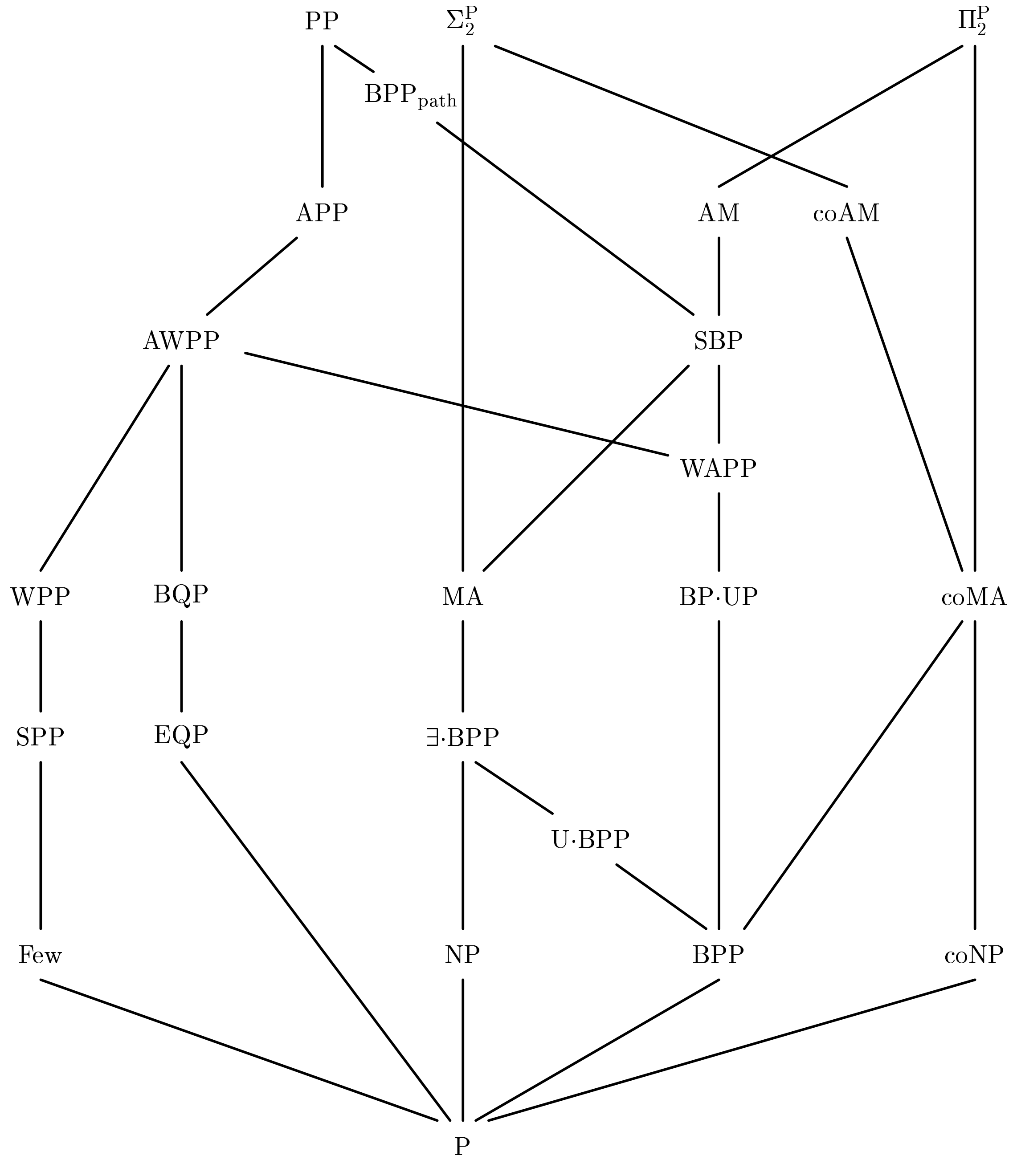

Figure 1: Relationships between SBP and known complexity classes
(inclusions hold from bottom to top)

5 Separation Results

In the previous sections our observations aimed at the localization of SBP with respect to known complexity classes. In particular this yielded $\mathrm{BPP} \subseteq \mathrm{SBP} \subseteq \mathrm{BPP}_{\mathrm{path}}$ and $\mathrm{MA} \subseteq \mathrm{SBP} \subseteq \mathrm{AM}$. However, up to now we have not provided any evidence of the strictness of these inclusions. So, the objective of this section is to give hints that make us strongly believe that SBP is different form the known classes BPP, $\mathrm{BPP}_{\mathrm{path}}$, MA, and AM. Furthermore, we will prove separation results with respect to Σ_2^{P} and classes defined in section 4.

As usual in complexity theory we cannot expect to find absolute separations since these would imply $\mathrm{P} \neq \mathrm{NP}$. Instead, we show either that the equivalence of SBP with other classes implies an unlikely complexity-theoretic consequence (such as a collapse of the polynomial-time hierarchy) or that SBP differs from other classes in suitable relativized worlds. We start with the separation of SBP from BPP and $\mathrm{BPP}_{\mathrm{path}}$.

Theorem 5.1 *If* $\mathrm{BPP} = \mathrm{SBP}$ *or* $\mathrm{SBP} = \mathrm{BPP}_{\mathrm{path}}$ *then the polynomial-time hierarchy collapses to its second level.*

Proof: If $\mathrm{SBP} \subseteq \mathrm{BPP}$ then $\mathrm{NP} \subseteq \mathrm{BPP}$. Sipser [Sip83] showed that this implies NP to have small circuits. By Karp and Lipton [KL82] it follows that the polynomial-time hierarchy collapses to its second level.

If $\mathrm{BPP}_{\mathrm{path}} \subseteq \mathrm{SBP}$ then we get $\mathrm{coNP} \subseteq \mathrm{BPP}_{\mathrm{path}} \subseteq \mathrm{SBP} \subseteq \mathrm{AM}$ from the Theorems 2.7 and 3.8. The result of Boppana, Håstad, and Zachos [BHZ87] shows that $\mathrm{coNP} \subseteq \mathrm{AM}$ implies a collapse of the polynomial-time hierarchy to its second level. $\square$

Corollary 5.2 *There exists an oracle* A *such that* $\mathrm{BPP}^A \neq \mathrm{SBP}^A$ *and* $\mathrm{SBP}^A \neq {\mathrm{BPP}_{\mathrm{path}}}^A$.

Proof: This holds since theorem 5.1 is relativizable and since there exists a relativized world where the polynomial-time hierarchy is infinite [Yao85]. $\square$

In contrast to Theorem 5.1, concerning the separation of SBP from MA and AM we could not prove similar unlikely consequences. Therefore, we approach this question with the construction of suitable relativized worlds where the conjectured separations hold. On one hand this gives some evidence that the separations could still hold in the nonrelativized case. On the other hand the oracles show that even if equalities like $\mathrm{MA} = \mathrm{SBP}$ and $\mathrm{SBP} = \mathrm{AM}$ hold then they can only be proved with nonrelativizable proof techniques. Since these techniques are known to be rare and difficult it is most likely that we are still a long way off from the final solution of these separation questions.

The separation results below will be derived on one hand from known oracle constructions [Yao85, Ver92, Bei94, For99] and on the other hand from a new construction that is described in the proof of Theorem 5.15. In particular, in this new relativized world, SBP is not contained in Σ_2^{P}. Since $\mathrm{SBP} \subseteq \mathrm{BPP}_{\mathrm{path}}$ holds relativizable we will see that our oracle shows that $\mathrm{BPP}_{\mathrm{path}} \not\subseteq \mathrm{R}^{\mathrm{NP}}$ and $\mathrm{BPP}_{\mathrm{path}} \not\subseteq \Sigma_2^{\mathrm{P}}$ in some relativized world. This solves an open question of Han *et al.* [HHT97]. We start our considerations with an oracle from Vereshchagin.

Theorem 5.3 ([Ver92]) *There exists an oracle* A *such that* $\mathrm{AM}^A \cap \mathrm{coAM}^A \not\subseteq \mathrm{PP}^A$.

Corollary 5.4 *There exists an oracle* A *such that* $\mathrm{AM}^A \not\subseteq \mathrm{SBP}^A$ *and* $\mathrm{coAM}^A \not\subseteq \mathrm{SBP}^A$.

Proof: Define A to be the oracle from Theorem 5.3. The corollary follows since $\mathrm{SBP} \subseteq \mathrm{PP}$ in all relativized worlds. $\square$

The following oracle goes back to a construction of Beigel.

Theorem 5.5 ([Bei94]) *There exists an oracle* A *such that* $\mathrm{P}^{\mathrm{NP}^A} \not\subseteq \mathrm{PP}^A$.

Corollary 5.6 *There exists an oracle A such that the following holds for every complexity class $\mathcal{C} \in \{\mathrm{APP}, \mathrm{AWPP}, \mathrm{WAPP}, \mathrm{BP{\cdot}UP}, \mathrm{BPP}, \mathrm{P}, \mathrm{WPP}, \mathrm{SPP}, \mathrm{Few}, \mathrm{BQP}, \mathrm{EQP}\}$.*

1. $\mathrm{NP}^A \not\subseteq \mathcal{C}^A$
2. $\exists{\cdot}\mathrm{BPP}^A \not\subseteq \mathcal{C}^A$
3. $\mathrm{MA}^A \not\subseteq \mathcal{C}^A$
4. $\mathrm{SBP}^A \not\subseteq \mathcal{C}^A$
5. $\mathrm{AM}^A \not\subseteq \mathcal{C}^A$
6. ${\Sigma_2^{\mathrm{P}}}^A \not\subseteq \mathcal{C}^A$

Proof: Define A to be the oracle from Theorem 5.5 and assume that $\mathrm{NP}^A \subseteq \mathrm{APP}^A$. In [Li93a, Li93b] Li proved that APP is low for PP. Since the proof is relativizable, APP^A is low for PP^A. In particular, this means $\mathrm{PP}^{\mathrm{NP}^A} \subseteq \mathrm{PP}^A$ and therefore $\mathrm{P}^{\mathrm{NP}^A} \subseteq \mathrm{PP}^A$. This contradicts the assumption on A and we get $\mathrm{NP}^A \not\subseteq \mathrm{APP}^A$. This proves statement 1 for $\mathcal{C} = \mathrm{APP}$. The remaining statements for $\mathcal{C} = \mathrm{APP}$ follow since NP is relativizably contained in $\exists{\cdot}\mathrm{BPP}$, MA, SBP, AM and Σ_2^{P}. The statements for the remaining classes $\mathcal{C}$ hold because these classes are subsets of APP in all relativized worlds. □

Corollary 5.7 *There exists an oracle A such that ${\Sigma_2^{\mathrm{P}}}^A \not\subseteq {\mathrm{BPP}_{\mathrm{path}}}^A$.*

Proof: This follows from Theorem 5.5 since $\mathrm{P}^{\mathrm{NP}} \subseteq \Sigma_2^{\mathrm{P}}$ and $\mathrm{BPP}_{\mathrm{path}} \subseteq \mathrm{PP}$ holds in all relativized worlds. □

At this point we want to mention another oracle that is interesting when looking at AWPP. In [FFKL93, FFKL] Fenner, Fortnow, Kurtz, and Li study the notion of $\mathcal{SP}$-genericy. In particular, it is shown that under any $\mathcal{SP}$-generic oracle it holds that the polynomial-time hierarchy is infinite and $\mathrm{P} = \mathrm{UP} = \mathrm{AWPP}$ (see [FFKL93] for definitions of and discussions on various genericy notions).

The next oracle we want to make use of is due to Fortnow.

Theorem 5.8 ([For99]) *There exists a relativized world where SPP strictly contains an infinite polynomial-time hierarchy.*

Corollary 5.9 *There exists an oracle A such that for every $\mathcal{C} \in \{\mathrm{SPP}, \mathrm{WPP}, \mathrm{AWPP}, \mathrm{APP}, \mathrm{PP}\}$:*

1. $\mathcal{C}^A \not\subseteq \mathrm{AM}^A$
2. $\mathcal{C}^A \not\subseteq \mathrm{SBP}^A$
3. $\mathcal{C}^A \not\subseteq \mathrm{MA}^A$
4. $\mathcal{C}^A \not\subseteq \exists{\cdot}\mathrm{BPP}^A$
5. $\mathcal{C}^A \not\subseteq \mathrm{NP}^A$
6. $\mathcal{C}^A \not\subseteq \mathrm{WAPP}^A$
7. $\mathcal{C}^A \not\subseteq \mathrm{BP{\cdot}UP}^A$
8. $\mathcal{C}^A \not\subseteq \mathrm{BPP}^A$
9. $\mathcal{C}^A \not\subseteq \mathrm{U{\cdot}BPP}^A$

Proof: Let A be the oracle from Theorem 5.8. Since $\mathrm{AM} \subseteq \mathrm{PH}$ holds relativizable we have $\mathrm{SPP}^A \not\subseteq \mathrm{AM}^A$ which shows statement 1 for $\mathcal{C} = \mathrm{SPP}$. The remaining statements for $\mathcal{C} = \mathrm{SPP}$ follow from the fact that the classes SBP, MA, $\exists{\cdot}$BPP, NP, WAPP, BP·UP, BPP, and U·BPP are subsets of AM in all relativized worlds. Finally, we obtain the statements corresponding to $\mathcal{C} \in \{\mathrm{WPP}, \mathrm{AWPP}, \mathrm{APP}, \mathrm{PP}\}$ since $\mathrm{SPP} \subseteq \mathrm{WPP} \subseteq \mathrm{AWPP} \subseteq \mathrm{APP} \subseteq \mathrm{PP}$ holds relativizable [Fen02]. □

Corollary 5.10 *There exists an oracle A such that ${\Sigma_2^{\mathrm{P}}}^A \not\subseteq \mathrm{SBP}^A$ and ${\Sigma_2^{\mathrm{P}}}^A \not\subseteq \mathrm{AM}^A$.*

Proof: This follows from Theorem 5.8 since in all relativized worlds, $\mathrm{SBP} \subseteq \mathrm{AM} \subseteq \Pi_2^{\mathrm{P}}$, and a collapse of the polynomial-time hierarchy is implied by $\Sigma_2^{\mathrm{P}} \subseteq \Pi_2^{\mathrm{P}}$. □

Remember that AM contains classes like NP, BPP, MA, and it is unlikely that AM is contained in Σ_2^{P}. So in this light AM seems to be quite powerful. However, Boppana, Håstad and Zachos [BHZ87] showed that unless the polynomial-time hierarchy collapses AM (and therefore also SBP) is not powerful enough to contain coNP. Together with Yao's oracle this has the following consequence.

Theorem 5.11 ([Yao85, BHZ87]) *There exists an oracle A such that* $\mathrm{coNP}^A \not\subseteq \mathrm{AM}^A$.

Proof: Yao [Yao85] constructed a relativized world A where the polynomial-time hierarchy is infinite. Boppana, Håstad, and Zachos [BHZ87] showed with a relativizable proof that $\mathrm{coNP} \subseteq \mathrm{AM}$ implies a collapse of the polynomial-time hierarchy to its second level. So we obtain $\mathrm{coNP}^A \not\subseteq \mathrm{AM}^A$. □

Corollary 5.12 *There exists an oracle A such that* $\mathrm{coNP}^A \not\subseteq \mathrm{SBP}^A$.

Proof: This follows since $\mathrm{SBP} \subseteq \mathrm{AM}$ and $\mathrm{coNP} \subseteq \Sigma_2^{\mathrm{P}}$ hold relativizably. □

We come now to a new oracle construction showing that even a certain subclass of $\mathrm{BP}{\cdot}\mathrm{UP}$ is probably not contained in Σ_2^{P}. In order to specify this subclass we define the following operator.

Definition 5.13 *For a complexity class $\mathcal{C}$ let $\overline{\mathrm{R}}{\cdot}\mathcal{C}$ be the class that consists of all languages L such that there exist an $B \in \mathcal{C}$, a polynomial p, and $\varepsilon > 0$ such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in L &\implies \#\{y \in \Sigma^{p(|x|)} : (x,y) \in B\} = 2^{p(|x|)} \\ x \notin L &\implies \#\{y \in \Sigma^{p(|x|)} : (x,y) \in B\} < (1-\varepsilon) \cdot 2^{p(|x|)}. \end{aligned}$$

The idea of this operator bases on coR, i.e., the complement of the probabilistic class R [Gil72, Gil77, the class VPP]. Note that in order to describe some class $\overline{\mathrm{R}}{\cdot}\mathcal{C}$ we cannot go back to some operator $\mathrm{R}\cdot$ which is derived from the class R, since $\overline{\mathrm{R}}{\cdot}\mathcal{C} = \mathrm{co}(\mathrm{R}{\cdot}\mathcal{C})$ holds only if $\mathcal{C}$ is closed under complementation.

Proposition 5.14 (Amplification for $\overline{\mathbf{R}}\cdot$) *If $\mathcal{C}$ is closed under $\leq_{ctt}^{P}$ then for every $L \in \overline{\mathrm{R}}{\cdot}\mathcal{C}$ and every polynomial q there exist a $B' \in \mathcal{C}$ and a polynomial p' such that for all $x \in \Sigma^*$:*

$$\begin{aligned} x \in L &\implies \#\{y \in \Sigma^{p'(|x|)} : (x,y) \in B'\} = 2^{p'(|x|)} \\ x \notin L &\implies \#\{y \in \Sigma^{p'(|x|)} : (x,y) \in B'\} < \frac{1}{2^{q(|x|)}} \cdot 2^{p'(|x|)}. \end{aligned}$$

Proof: Since $L \in \overline{\mathrm{R}}{\cdot}\mathcal{C}$ there exists a $B \in \mathcal{C}$, a polynomial p, and $\varepsilon > 0$ such that for all $x \in \Sigma^*$:

$$\begin{aligned} x \in L &\implies \#\{y \in \Sigma^{p(|x|)} : (x,y) \in B\} = 2^{p(|x|)} \\ x \notin L &\implies \#\{y \in \Sigma^{p(|x|)} : (x,y) \in B\} < (1-\varepsilon) \cdot 2^{p(|x|)}. \end{aligned}$$

Without loss of generality we may assume that $\varepsilon < 1$. Let $c \stackrel{df}{=} \lceil -1/\log_2(1-\varepsilon)\rceil$, $p'(n) \stackrel{df}{=} c \cdot p(n) \cdot q(n)$, and observe that $c > 0$ and $(1-\varepsilon)^c \leq (1-\varepsilon)^{-(\log_2 2)/\log_2(1-\varepsilon)} = (1-\varepsilon)^{-\log_{(1-\varepsilon)} 2} = \frac{1}{2}$.

$$B' \stackrel{df}{=} \left\{(x,y) \ : \ x \in \Sigma^*, y = y_1 y_2 \cdots y_{c\cdot q(|x|)} \text{ for suitable } y_i \in \Sigma^{p(|x|)} \text{ and } \bigwedge_{1\leq i\leq c\cdot q(|x|)} (x,y_i) \in B\right\}.$$

Obviously, $B' \leq_{ctt}^{P} B$ and therefore $B' \in \mathcal{C}$. Now consider an arbitrary $x \in \Sigma^*$. If $x \in L$ then $(x,y) \in B$ for all $y \in \Sigma^{p(|x|)}$ which in turn implies $\#\{y \in \Sigma^{p'(|x|)} : (x,y) \in B'\} = 2^{p'(|x|)}$. If $x \notin L$ then

$$\begin{aligned} \#\{y \in \Sigma^{p'(|x|)} : (x,y) \in B'\} &= \left(\#\{y \in \Sigma^{p(|x|)} : (x,y) \in B\}\right)^{c\cdot q(|x|)} \\ &< (1-\varepsilon)^{c\cdot q(|x|)} \cdot 2^{c\cdot q(|x|)\cdot p(|x|)} \\ &\leq \frac{1}{2^{q(|x|)}} \cdot 2^{p'(|x|)}. \end{aligned}$$

□

When we apply this proposition to UP we see that the class $\overline{\mathrm{R}}{\cdot}\mathrm{UP}$ admits amplification. In contrast, we cannot show the same for $\mathrm{BP}{\cdot}\mathrm{UP}$.

We turn to the construction of an oracle A with $\mathrm{SBP}^A \not\subseteq \mathrm{MA}^A$. We will prove a result which is stronger, namely that there exists a relativized world where $\overline{\mathrm{R}}{\cdot}\mathrm{UP} \not\subseteq \Sigma_2^{\mathrm{P}}$. Since $\overline{\mathrm{R}}{\cdot}\mathrm{UP} \subseteq \mathrm{BP}{\cdot}\mathrm{UP} \subseteq \mathrm{SBP}$ and $\mathrm{MA} \subseteq \Sigma_2^{\mathrm{P}}$ in all relativized worlds, we will finally get $\mathrm{BP}{\cdot}\mathrm{UP}^A \not\subseteq \mathrm{MA}^A$ and $\mathrm{SBP}^A \not\subseteq \mathrm{MA}^A$.

Theorem 5.15 *There exists an oracle A such that $\overline{\mathrm{R}}{\cdot}\mathrm{UP}^A \not\subseteq \exists{\cdot}\forall{\cdot}\mathrm{P}^A$.*

Before we prove this theorem let us summarize some immediate consequences.

Corollary 5.16 *There exists an oracle A such that the following holds for every complexity class $\mathcal{C} \in \{\Sigma_2^{\mathrm{P}^{\mathrm{BPP}}}, \Sigma_2^{\mathrm{P}}, \mathrm{MA}, \exists{\cdot}\mathrm{BPP}, \mathrm{NP}, \mathrm{BPP}, \mathrm{coAM}, \mathrm{coMA}, \mathrm{coNP}\}$.*

1. $\overline{\mathrm{R}}{\cdot}\mathrm{UP}^A \not\subseteq \mathcal{C}^A$
2. $\mathrm{BP}{\cdot}\mathrm{UP}^A \not\subseteq \mathcal{C}^A$
3. $\mathrm{WAPP}^A \not\subseteq \mathcal{C}^A$
4. $\mathrm{AWPP}^A \not\subseteq \mathcal{C}^A$
5. $\mathrm{SBP}^A \not\subseteq \mathcal{C}^A$
6. $\mathrm{AM}^A \not\subseteq \mathcal{C}^A$

Proof: Let A be the oracle from Theorem 5.15 and note that $\exists{\cdot}\forall{\cdot}\mathrm{P}^A = \Sigma_2^{\mathrm{P}^A}$. In [Sch89] Schöning showed that BPP is low for Σ_2^{P}. Since this theorem is relativizable we obtain the statement 1 for $\mathcal{C} = \Sigma_2^{\mathrm{P}^{\mathrm{BPP}}}$. The remaining statements for $\mathcal{C} = \Sigma_2^{\mathrm{P}^{\mathrm{BPP}}}$ hold since $\overline{\mathrm{R}}{\cdot}\mathrm{UP}$ is relativizable contained in the classes $\mathrm{BP}{\cdot}\mathrm{UP}$, WAPP, AWPP, SBP, and AM. The statements for the remaining classes $\mathcal{C}$ follow since these classes are subclasses of $\Sigma_2^{\mathrm{P}^{\mathrm{BPP}}}$ in all relativized worlds. □

Corollary 5.17 *There exists an oracle A such that SBP^A is not closed under complementation. In particular, SBP^A is closed neither under Turing-reductions nor under truth-table reductions.*

Proof: By Theorem 3.8, $\mathrm{SBP} \subseteq \mathrm{AM}$. Since this proof and the proof for $\mathrm{AM} \subseteq \Pi_2^{\mathrm{P}}$ are relativizable we have $\mathrm{SBP} \subseteq \Pi_2^{\mathrm{P}}$ in all relativized worlds. But by Corollary 5.16 there exists an oracle A such that $\mathrm{SBP}^A \not\subseteq \Sigma_2^{\mathrm{P}^A}$. Hence SBP^A is not closed under complementation. □

In [HHT97] Han *et al.* introduce and investigate the threshold class $\mathrm{BPP_{path}}$. We have seen (cf. Proposition 3.10) that $\mathrm{BPP_{path}}$ is closely related to SBP, i.e., if we start from SBP's characterization in Proposition 3.3.4 and if we allow g to be a $\#\mathrm{P}$ function then we meet $\mathrm{BPP_{path}}$. [HHT97] compares in particular the classes $\mathrm{BPP_{path}}$ and BPP, and poses as an open question whether Sipser's [Sip83] result $\mathrm{BPP} \subseteq \mathrm{R}^{\mathrm{NP}} \subseteq \Sigma_2^{\mathrm{P}}$ can be transferred to $\mathrm{BPP_{path}}$. With the oracle from Theorem 5.15 we have found a relativized world where this question has a negative answer.

Corollary 5.18 *There exists an oracle A such that $\mathrm{BPP_{path}}^A \not\subseteq \Sigma_2^{\mathrm{P}^A}$.*

Proof: Since the proof of Corollary 3.11 is relativizable, we have $\mathrm{SBP} \subseteq \mathrm{BPP_{path}}$ in all relativized worlds. So from Corollary 5.16 it follows that there is an oracle A with $\mathrm{BPP_{path}}^A \not\subseteq \Sigma_2^{\mathrm{P}^A}$. □

The corollaries above show that if SBP coincides with known complexity classes then the corresponding proofs cannot be relativizable. Moreover, we have seen that SBP and APP (resp., AWPP) are incomparable under relativizing proof techniques. These oracle results give evidence that also in the real world SBP does not coincide with known complexity classes. A summary of inclusions and separations concerning SBP is given in Figure 2 below.

We turn now to the remaining proof of Theorem 5.15. In this oracle construction we will need the following estimation.

Lemma 5.19 *Let $a_1 \stackrel{df}{=} 2^{12}$ and $a_{i+1} \stackrel{df}{=} 2^{a_i}$ for $i \geq 1$. Then $2^{a_i/4} \geq (a_i)^i$ for $i \geq 1$.*

Proof: This can be seen as follows.

$$\begin{array}{rcllcl} a_i \geq 16 \cdot i^2 & \Longrightarrow & \frac{a_i}{\log_2 a_i} \cdot \log_2 2^{1/4} & \geq & \frac{1}{4} \cdot \sqrt{a_i} \geq i, & \text{since } \sqrt{x} \geq \log_2 x \text{ for } x \geq 4 \\ & \Longrightarrow & a_i \cdot \log_2 2^{1/4} & \geq & i \cdot \log_2 a_i & \\ & \Longrightarrow & \log_2 2^{a_i/4} & \geq & \log_2 (a_i)^i & \\ & \Longrightarrow & 2^{a_i/4} & \geq & (a_i)^i & \end{array}$$

□

Proof of Theorem 5.15: We will construct oracle stages $A_1, A_2, \ldots$ and at the end we will define $A \overset{df}{=} \bigcup_{i\geq 1} A_i$. As an abbreviation for intervals of stages A_i we use $A[k,j] \overset{df}{=} \bigcup_{k\leq i\leq j} A_i$. Let $a_1 \overset{df}{=} 2^{12}$ and $a_{i+1} \overset{df}{=} 2^{a_i}$ for $i \geq 1$. Moreover, for every $B \subseteq \Sigma^*$ and every $i \geq 1$ we define the following conditions:

$$\mathrm{C1}(B,i) \overset{df}{=} \text{for every } x \in \Sigma^{a_i/4} \text{ there exists at most one } y \in \Sigma^{a_i\cdot 3/4} \text{ with } xy \in B$$

$$\mathrm{C2}(B,i) \overset{df}{=} |B \cap \Sigma^{a_i}| = 2^{a_i/4} \vee |B \cap \Sigma^{a_i}| \leq \frac{1}{2} \cdot 2^{a_i/4}$$

The oracle construction will be such that $A_i \subseteq \Sigma^{a_i} \wedge \mathrm{C1}(A[1,i],i) \wedge \mathrm{C2}(A[1,i],i)$ for each $i \geq 1$ (note that these conditions are equivalent to $A_i \subseteq \Sigma^{a_i} \wedge \mathrm{C1}(A_i,i) \wedge \mathrm{C2}(A_i,i)$). For $B \subseteq \Sigma^*$ let

$$W(B) \overset{df}{=} \{0^{a_i} \ : \ i \geq 1 \text{ and for all } x \in \Sigma^{a_i/4} \text{ there exists exactly one } y \in \Sigma^{a_i\cdot 3/4} \text{ with } xy \in B\}.$$

We will use $W(A)$ as a witness language: Assume that $A_i \subseteq \Sigma^{a_i} \wedge \mathrm{C1}(A[1,i],i) \wedge \mathrm{C2}(A[1,i],i)$ holds for all $i \geq 1$, and let $A \overset{df}{=} \bigcup_{i\geq 1} A_i$. Then, since $\mathrm{C1}(A[1,i],i)$ holds, the set $W'(A) \overset{df}{=} \{(0^{a_i}, x) :$ $x \in \Sigma^{a_i/4}$ and there is exactly one $y \in \Sigma^{a_i\cdot 3/4}$ such that $xy \in A\}$ is in UP^A. So if $0^{a_i} \in W(A)$ then $\#\{x \in \Sigma^{a_i/4} : (0^{a_i}, x) \in W'(A)\} = 2^{a_i/4}$. If $0^{a_i} \notin W(A)$ then there is an $x \in \Sigma^{a_i/4}$ such that there is no $y \in \Sigma^{a_i\cdot 3/4}$ with $xy \in A$. Since $\mathrm{C2}(A[1,i],i)$ holds, this implies $\#\{x \in \Sigma^{a_i/4} : (0^{a_i}, x) \in W'(A)\} \leq \frac{1}{2} \cdot 2^{a_i/4}$. Therefore, we have $W(A) \in \overline{\mathrm{R}}\cdot\mathrm{UP}^A$. Additionally, A will be constructed such that $W(A) \notin \exists\cdot\forall\cdot\mathrm{P}^A$.

Let $T_1, T_2, \ldots$ be an enumeration of all triples of the form $T = (M, r, s)$ where M is a deterministic polynomial-time oracle machine and r, s are polynomials. Without loss of generality we may assume that if $T_i = (M_i, r_i, s_i)$ then $r_i(n) \leq n^i$ and there exists a polynomial $t_i(n) \leq n^i$ such that the computation $M_i^B(x,y,z)$ halts in $t_i(|x|)$ steps for any oracle B and any $x \in \Sigma^+$, $y \in \Sigma^{r(|x|)}$, $z \in \Sigma^{s(|x|)}$.

In order to achieve $W(A) \notin \exists\cdot\forall\cdot\mathrm{P}^A$, during the construction of stage A_i we diagonalize against the triple T_i in the following sense: We interpret T_i as a possible "$\exists\cdot\forall\cdot\mathrm{P}$-machine" for $W(A)$ and we construct A_i such that the machine fails to give the right answer with respect to the question $0^{a_i} \in W(A)$. More precisely, if $T_i = (M_i, r_i, s_i)$ then with the construction of A_i we will *prevent* the following equivalence.

$$0^{a_i} \in W(A[1,i]) \iff \big(\exists y \in \Sigma^{r_i(a_i)}\big)\big(\forall z \in \Sigma^{s_i(a_i)}\big)\big[(0^{a_i}, y, z) \in \mathrm{L}(M_i^{A[1,i]})\big]$$

So our construction will additionally satisfy the conditions $\mathrm{C3}(A[1,i],i)$ for $i \geq 1$ which are defined as follows: For $B \subseteq \Sigma^*$ and $i \geq 1$ let

$$\mathrm{C3}(B,i) \overset{df}{=} \neg\Big(0^{a_i} \in W(B) \iff \big(\exists y \in \Sigma^{r_i(a_i)}\big)\big(\forall z \in \Sigma^{s_i(a_i)}\big)\big[(0^{a_i}, y, z) \in \mathrm{L}(M_i^B)\big]\Big).$$

As an abbreviation for the conditions defined so far we use $\mathrm{C}(B,i) \overset{df}{=} \mathrm{C1}(B,i) \wedge \mathrm{C2}(B,i) \wedge \mathrm{C3}(B,i)$.

Claim 5.20 *There exist oracle stages $A_1, A_2, \ldots$ such that $A_i \subseteq \Sigma^{a_i}$ and $\mathrm{C}(A[1,i],i)$ for all $i \geq 1$.*

Before we prove this claim let us see that it implies the correctness of the theorem. We have already seen that with $A \overset{df}{=} \bigcup_{i\geq 1} A_i$ it holds that $W(A) \in \overline{\mathrm{R}}\cdot\mathrm{UP}^A$. Assume that $W(A) \in \exists\cdot\forall\cdot\mathrm{P}^A$, i.e., there exist a deterministic polynomial-time oracle machine M and polynomials r, s such that for all $x \in \Sigma^*$:

$$x \in W(A) \iff \big(\exists y \in \Sigma^{r(|x|)}\big)\big(\forall z \in \Sigma^{s(|x|)}\big)\big[(x,y,z) \in \mathrm{L}(M^A)\big] \tag{8}$$

Hence there exists some $i \geq 1$ such that $T_i = (M_i, r_i, s_i) = (M, r, s)$. We consider equation (8) for $x \overset{df}{=} 0^{a_i}$. Note that $0^{a_i} \in W(A) \iff 0^{a_i} \in W(A[1,i])$. Moreover, by Lemma 5.19 the sequence of a_i's grows fast enough such that for every oracle $B \subseteq \Sigma^*$ the computations $M^B(0^{a_i}, y, z)$ cannot ask for words of length $\geq a_{i+1}$ (remember our assumption on the enumeration of the triples T_i). Therefore, for these computations it is equivalent to use oracle $A[1,i]$ instead of A. So from equation (8) we obtain

$$0^{a_i} \in W(A[1,i]) \iff \big(\exists y \in \Sigma^{r_i(a_i)}\big)\big(\forall z \in \Sigma^{s_i(a_i)}\big)\big[(0^{a_i}, y, z) \in \mathrm{L}(M_i^{A[1,i]})\big]. \tag{9}$$

By Claim 5.20, $\mathrm{C}(A[1,i],i)$ holds. In particular this implies $\mathrm{C3}(A[1,i],i)$ which in turn contradicts equation (9). So we get $W(A) \notin \exists\cdot\forall\cdot\mathrm{P}^A$; this proves the theorem.

So, it remains to show the correctness of Claim 5.20. We will prove this by contradiction, i.e., we will derive a contradiction from the following assumption.

A1 $\stackrel{\text{df}}{=}$ there exists some $n \geq 1$ and oracle stages $A_1, A_2, \ldots, A_{n-1}$ such that $A_i \subseteq \Sigma^{a_i} \wedge \mathrm{C}(A[1,i],i)$ for $1 \leq i < n$, and there does not exist an $A' \subseteq \Sigma^{a_n}$ with $\mathrm{C}(A[1,n-1] \cup A', n)$.

So assume A1. Let $\alpha = \frac{3}{4} \cdot a_n$, $\beta = 2^{\frac{\alpha}{3}} = 2^{\frac{a_n}{4}}$, and $\psi = \frac{\alpha \cdot \beta}{4}$. We will show that under this assumption we could encode an arbitrary number $\mathcal{N} \in [0, 2^\psi)$ with less than ψ bits. For simplicity, we will write M, r, s instead of M_n, r_n, s_n, respectively. Choose a prime number $p \in (2^{\alpha-1}, 2^\alpha]$; this is possible by Bertrand's postulate[1] which says that for every $k \geq 1$ there is some prime number p with $k < p \leq 2k$. Each $\mathcal{N} \in [0, 2^\psi)$ can be represented as a $\frac{\beta}{2}$-digit number with digits from $[0, 2^{\alpha-1})$ since $2^\psi = 2^{\frac{\alpha}{2}\cdot\frac{\beta}{2}} \leq (2^{\alpha-1})^{\frac{\beta}{2}}$. These digits can be considered as elements of the finite field $\mathrm{GF}(p)$. So each such $\mathcal{N}$ can be thought of as a $\frac{\beta}{2}$-dimensional vector $\vec{z}_\mathcal{N} \in \mathrm{GF}(p)^{\frac{\beta}{2}}$.

Now, we make the vectors $\vec{z}_\mathcal{N}$ redundant, i.e., we double their dimension and transform them into vectors $\vec{y}_\mathcal{N} \in \mathrm{GF}(p)^\beta$ in such a way that $\vec{z}_\mathcal{N}$ can be reconstructed when knowing an arbitrary half of the components of $\vec{y}_\mathcal{N}$. For this, we define the following matrix over $\mathrm{GF}(p)$ which can be considered as a generalization of a Vandermonde matrix.

$$\mathcal{M} \stackrel{\text{df}}{=} \begin{pmatrix} 1^1 & 1^2 & 1^3 & \cdots & 1^{\frac{\beta}{2}} \\ 2^1 & 2^2 & 2^3 & \cdots & 2^{\frac{\beta}{2}} \\ 3^1 & 3^2 & 3^3 & \cdots & 3^{\frac{\beta}{2}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \beta^1 & \beta^2 & \beta^3 & \cdots & \beta^{\frac{\beta}{2}} \end{pmatrix}.$$

It is known that if one deletes $\frac{\beta}{2}$ arbitrary rows in this matrix then one obtains a quadratic matrix which is invertible in $\mathrm{GF}(p)$. Therefore, if one knows $\frac{\beta}{2}$ components from the β-dimensional vector $\vec{y}_\mathcal{N} \stackrel{\text{df}}{=} \mathcal{M} \cdot \vec{z}_\mathcal{N}^{\,T}$ (i.e., the inner product modulo p), then one can reconstruct the vector $\vec{z}_\mathcal{N}$; or equivalently, if $\vec{y}_{\mathcal{N}_1} = \mathcal{M} \cdot (\vec{z}_{\mathcal{N}_1})^T$, $\vec{y}_{\mathcal{N}_2} = \mathcal{M} \cdot (\vec{z}_{\mathcal{N}_2})^T$ and the vectors $\vec{y}_{\mathcal{N}_1}$ and $\vec{y}_{\mathcal{N}_2}$ coincide in at least $\frac{\beta}{2}$ many components, then $\vec{z}_{\mathcal{N}_1} = \vec{z}_{\mathcal{N}_2}$ and $\vec{y}_{\mathcal{N}_1} = \vec{y}_{\mathcal{N}_2}$. Since $p \leq 2^\alpha$, the vector $\vec{y}_\mathcal{N}$ can be represented as the oracle stage $A_\mathcal{N}$,

$$\begin{aligned} A_\mathcal{N} \stackrel{\text{df}}{=} \{w \in \Sigma^{a_n} \,:\, & w = w_1 w_2, |w_1| = a_n/4 \text{ and } w_2 \text{ is the binary} \\ & \text{representation of the } w_1\text{-th component of } \vec{y}_\mathcal{N}\}. \end{aligned}$$

So, in this sense, each $\mathcal{N} \in [0, 2^\psi)$ induces a vector $\vec{z}_\mathcal{N}$ which induces a redundant vector $\vec{y}_\mathcal{N}$ which in turn induces an oracle stage $A_\mathcal{N}$. We get the following claim from our above observations.

Claim 5.21 *Each $\mathcal{N} \in [0, 2^\psi)$ can be reconstructed from an arbitrary half of the words in $A_\mathcal{N}$. Formally, if $\mathcal{N}_1, \mathcal{N}_2 \in [0, 2^\psi)$ and $|A_{\mathcal{N}_1} \cap A_{\mathcal{N}_2}| \geq \frac{\beta}{2}$, then $\mathcal{N}_1 = \mathcal{N}_2$.*

Note that $0^{a_n} \in W(A[1, n-1] \cup A_\mathcal{N})$ for each $\mathcal{N} \in [0, 2^\psi)$. Moreover, it holds that $A_\mathcal{N} \subseteq \Sigma^{a_n}$ and $\mathrm{C1}(A[1,n-1] \cup A_\mathcal{N}, n) \wedge \mathrm{C2}(A[1,n-1] \cup A_\mathcal{N}, n)$. But by assumption A1, we have $\neg\mathrm{C}(A[1,n-1] \cup A_\mathcal{N}, n)$ and therefore $\neg\mathrm{C3}(A[1,n-1] \cup A_\mathcal{N}, n)$. Together with $0^{a_n} \in W(A[1,n-1] \cup A_\mathcal{N})$ it follows

$$\big(\exists y \in \Sigma^{r(a_n)}\big)\big(\forall z \in \Sigma^{s(a_n)}\big)\big[(0^{a_n}, y, z) \in \mathrm{L}(M^{A[1,n-1] \cup A_\mathcal{N}})\big]. \tag{10}$$

[1]This was first conjectured by J. Bertrand and in 1850 proved by P. Chebychev. In 1937, A. E. Ingham [Ing37] showed that there is at least one prime number between neighboured cubic numbers. It is still an open question whether the same holds for neighboured squares.

For each $\mathcal{N} \in [0, 2^{\psi})$, let $y_{\mathcal{N}}$ be the lexicographically smallest witness of this condition. Although the length of $y_{\mathcal{N}}$ is polynomial in a_n, it contains much information about $\mathcal{N}$; we will use this information to reconstruct $\mathcal{N}$. Informally, our further way is as follows: we use certain subsets $B \subseteq A[1, n-1] \cup A_{\mathcal{N}}$ as oracle and look for words z such that the computation $M^B(0^{a_n}, y_{\mathcal{N}}, z)$ rejects. Each of these computations asks for at least one word in $A_{\mathcal{N}}$. If we repeat these considerations for several z then this reveals many different words from $A_{\mathcal{N}}$. A single such word is characterized by its position in the computation $M^B(0^{a_n}, y_{\mathcal{N}}, z)$ which can be described in $\mathrm{O}(\log_2 a_n)$ bits. So, only a few bits are needed to encode the words z, and with these words at hand we are able to reconstruct $A_{\mathcal{N}}$ and therefore also $\mathcal{N}$.

For every $\mathcal{N} \in [0, 2^{\psi})$, $Q \subseteq \Sigma^{a_n}$ and $z \in \Sigma^{s(a_n)}$ we define $q^Q_{\mathcal{N},z}$ as the sequence $(w_0, w_1, \ldots, w_j)$ of oracle queries that are asked in the computation $M^{A[1,n-1]\cup Q}(0^{a_n}, y_{\mathcal{N}}, z)$. For convenience we will use this query sequence also in the sense of a set. Consider the following algorithm `ApproxA` for every $\mathcal{N} \in [0, 2^{\psi})$.

1. $Q := \emptyset$
2. for $i := 1$ to $\frac{\beta}{2}$
3. choose the smallest $z \in \Sigma^{s(a_n)}$ such that $M^{A[1,n-1]\cup Q}(0^{a_n}, y_{\mathcal{N}}, z)$ rejects
4. choose the smallest element from $q^Q_{\mathcal{N},z} \cap (A_{\mathcal{N}} \setminus Q)$ and add it to the set Q
5. next i
6. return Q

This algorithm looks for words from $A_{\mathcal{N}}$ and it collects these words in the set Q. So, `ApproxA` can be considered as an approximation procedure for $A_{\mathcal{N}}$. However, it is not immediately clear that the steps 3 and 4 can always be carried out. The following two claims make sure that this is possible.

Claim 5.22 *Let $\mathcal{N} \in [0, 2^{\psi})$ and consider the computation of* `ApproxA`*. The choice of z in step 3 is always possible, i.e., $\big(\exists z \in \Sigma^{s(a_n)}\big)\big[(0^{a_n}, y_{\mathcal{N}}, z) \notin \mathrm{L}(M^{A[1,n-1]\cup Q})\big]$.*

Assume that there exists a moment where the choice in step 3 is not possible. Of course it holds that $Q \subseteq A_{\mathcal{N}}$ and $|Q| \leq \frac{\beta}{2}$. So we obtain $Q \subseteq \Sigma^{a_n} \wedge \mathrm{C1}(A[1, n-1] \cup Q, n) \wedge \mathrm{C2}(A[1, n-1] \cup Q, n)$. Additionally we have $0^{a_n} \notin W(A[1, n-1] \cup Q)$. From the assumption of this claim it follows that $\big(\exists y \in \Sigma^{r(a_n)}\big)\big(\forall z \in \Sigma^{s(a_n)}\big)[(0^{a_n}, y, z) \in \mathrm{L}(M^{A[1,n-1]\cup Q})]$ and therefore $\mathrm{C3}(A[1, n-1] \cup Q, n)$. So we get $\mathrm{C}(A[1, n-1] \cup Q, n)$ which contradicts the assumption A1. This proves claim 5.22.

Claim 5.23 *Let $\mathcal{N} \in [0, 2^{\psi})$ and consider the computation of* `ApproxA`*. The choice in step 4 is always possible, i.e., $q^Q_{\mathcal{N},z} \cap (A_{\mathcal{N}} \setminus Q) \neq \emptyset$.*

By the choice of $y_{\mathcal{N}}$ and by claim 5.22 the following holds for each value of Q that is possible in step 3.

- $\big(\forall z \in \Sigma^{s(a_n)}\big)\big[(0^{a_n}, y_{\mathcal{N}}, z) \in \mathrm{L}(M^{A[1,n-1]\cup A_{\mathcal{N}}})\big]$
- $\big(\exists z \in \Sigma^{s(a_n)}\big)\big[(0^{a_n}, y_{\mathcal{N}}, z) \notin \mathrm{L}(M^{A[1,n-1]\cup Q})\big]$
- $Q \subseteq A_{\mathcal{N}}$

If z is a witness of the second condition then $(0^{a_n}, y_{\mathcal{N}}, z) \in \mathrm{L}(M^{A[1,n-1]\cup A_{\mathcal{N}}}) \setminus \mathrm{L}(M^{A[1,n-1]\cup Q})$. This means that there is at least one oracle query q such that $q \in A_{\mathcal{N}} \setminus Q$ and q is asked during the computation $M^{A[1,n-1]\cup Q}(0^{a_n}, y_{\mathcal{N}}, z)$, i.e., $q^Q_{\mathcal{N},z} \cap (A_{\mathcal{N}} \setminus Q) \neq \emptyset$. This proves claim 5.23.

By the previous claims, each step of `ApproxA` can be carried out. So it is easy to see that `ApproxA` returns a set Q with $|Q| = \frac{\beta}{2}$ and $Q \subseteq A_{\mathcal{N}}$. But we still have the problem that `ApproxA` on input $\mathcal{N}$

makes use of the oracle stage $A_{\mathcal{N}}$. However, we will see that with help of a few bits of information one can abstain from $A_{\mathcal{N}}$. We just need to know $y_{\mathcal{N}}$ and the information which word from $q^Q_{\mathcal{N},z}$ was chosen in step 4. The latter can be described with $\mathrm{O}(\log_2 a_n)$ bits since the cardinality of $q^Q_{\mathcal{N},z}$ is polynomial in a_n.

By our assumption, there exists a polynomial $t(n) \leq n^i$ such that for all oracles B and all $x \in \Sigma^+$, $y \in \Sigma^{r(|x|)}$, $z \in \Sigma^{s(|x|)}$ the computation $M^B_n(x, y, z)$ halts in $t(|x|) \leq |x|^n$ steps. Let $m \overset{df}{=} \lceil \log_2 t(a_n) \rceil$. Consider the computation of `ApproxA` for an arbitrary $\mathcal{N} \in [0, 2^\psi)$ and assume that we are in the i-th pass of the loop in step 4. Here we choose a certain word from the query sequence $q^Q_{\mathcal{N},z} = (w_0, w_1, \ldots, w_j)$ and we add this word to Q. Note that $j < t(a_n)$ holds by the definition of $q^Q_{\mathcal{N},z}$. If we choose the word w_k with $0 \leq k \leq j$ then define $w_{\mathcal{N},i}$ to be the m-digit binary representation of k. For every $\mathcal{N} \in [0, 2^\psi)$ we define the packed encoding of $\mathcal{N}$ as $\mathrm{Code}(\mathcal{N}) \overset{df}{=} y_{\mathcal{N}} \cdot w_{\mathcal{N},1} \cdot w_{\mathcal{N},2} \cdot w_{\mathcal{N},3} \cdots w_{\mathcal{N},\frac{\beta}{2}}$.

Claim 5.24 *Each $\mathcal{N} \in [0, 2^\psi)$ can be reconstructed from $\mathrm{Code}(\mathcal{N})$. Formally, if $\mathcal{N}_1, \mathcal{N}_2 \in [0, 2^\psi)$ and $\mathrm{Code}(\mathcal{N}_1) = \mathrm{Code}(\mathcal{N}_2)$ then $\mathcal{N}_1 = \mathcal{N}_2$.*

Let $\mathcal{N} \in [0, 2^\psi)$ and assume that we are given $\mathrm{Code}(\mathcal{N}) = y_{\mathcal{N}} \cdot w_{\mathcal{N},1} \cdot w_{\mathcal{N},2} \cdot w_{\mathcal{N},3} \cdots w_{\mathcal{N},\frac{\beta}{2}}$. First of all we see that we can simulate the computation of `ApproxA` (without the knowledge of $\mathcal{N}$ and $A_{\mathcal{N}}$) because

1. step 3 can be simulated with help of $y_{\mathcal{N}}$, and
2. in step 4 with help of the words $w_{\mathcal{N},j}$ we chose the right word from $q^Q_{\mathcal{N},z}$.

We know that this simulation yields a set Q with $Q \subseteq A_{\mathcal{N}}$ and $|Q| = \frac{\beta}{2}$. Therefore, if $\mathcal{N}_1, \mathcal{N}_2 \in [0, 2^\psi)$ and if we use $\mathrm{Code}(\mathcal{N}_1) = \mathrm{Code}(\mathcal{N}_2)$ for the simulation then we get a set Q with $Q \subseteq A_{\mathcal{N}_1} \cap A_{\mathcal{N}_1}$ and $|Q| = \frac{\beta}{2}$. From claim 5.21 it follows that $\mathcal{N}_1 = \mathcal{N}_2$. This proves claim 5.24.

In order to determine $|\mathrm{Code}(\mathcal{N})|$ we make the following estimation: For $x \geq 6$ it holds that $2x < \frac{2^x}{4}$. If we let $y \overset{df}{=} 2^{2x}$ then we obtain $\log_2 y < \sqrt{y}/4$ for $y \geq 2^{12}$. It follows that $(\log_2 y)^2 < \frac{3}{16} y$ for $y \geq 2^{12}$. Since $a_i \geq 2^{12}$ we get for $i \geq 1$,

$$(\log_2 a_i)^2 < \frac{3}{16} a_i. \tag{11}$$

For every $\mathcal{N} \in [0, 2^\psi)$ the length of $\mathrm{Code}(\mathcal{N})$ can be estimated as follows.

$$\begin{aligned}
|\mathrm{Code}(\mathcal{N})| &= r(a_n) + \lceil \log_2 t(a_n) \rceil \cdot \frac{\beta}{2} \\
&\leq (a_n)^n + \lceil \log_2((a_n)^n) \rceil \cdot \frac{\beta}{2} \qquad \text{(by the assumptions about } r \text{ and } t\text{)} \\
&\leq (a_n)^n + 2n \cdot \log_2(a_n) \cdot \beta \\
&\leq 3n \cdot \log_2(a_n) \cdot \beta \qquad \text{(by proposition 5.19)} \\
&\leq (\log_2 a_n)^2 \cdot \beta \qquad \text{(since } a_n ge 2^{3n}\text{)} \\
&< \psi \qquad \text{(by equation (11))}
\end{aligned}$$

This means that the number of code words is less than 2^ψ. Hence there exist two different numbers $\mathcal{N}_1, \mathcal{N}_2 \in [0, 2^\psi)$ such that $\mathrm{Code}(\mathcal{N}_1) = \mathrm{Code}(\mathcal{N}_2)$. This contradicts claim 5.24. Therefore, our assumption A1 is false. This proves claim 5.20 and completes the proof of the theorem. □

Following definition 2.2, for a complexity class $\mathcal{C}$ we say that a language L belongs to $\exists!\cdot\mathcal{C}$ if and only if there exist a set $B \in \mathcal{C}$ and a polynomial p such that the equivalence $x \in L \Leftrightarrow \mathrm{count}^{=p}_B(x) = 1$ holds for all $x \in \Sigma^*$. Note that the oracle construction above also shows that $W(A) \in \forall\cdot\exists!\cdot\mathrm{P}^A$ and $W(A) \notin \exists\cdot\forall\cdot\mathrm{P}^A$. This yields the following oracle which could be of interest in connection with leaf languages.

Corollary 5.25 *There exists an oracle A such that $\forall\cdot\exists!\cdot\mathrm{P}^A \not\subseteq \Sigma_2^{\mathrm{P}^A}$.*

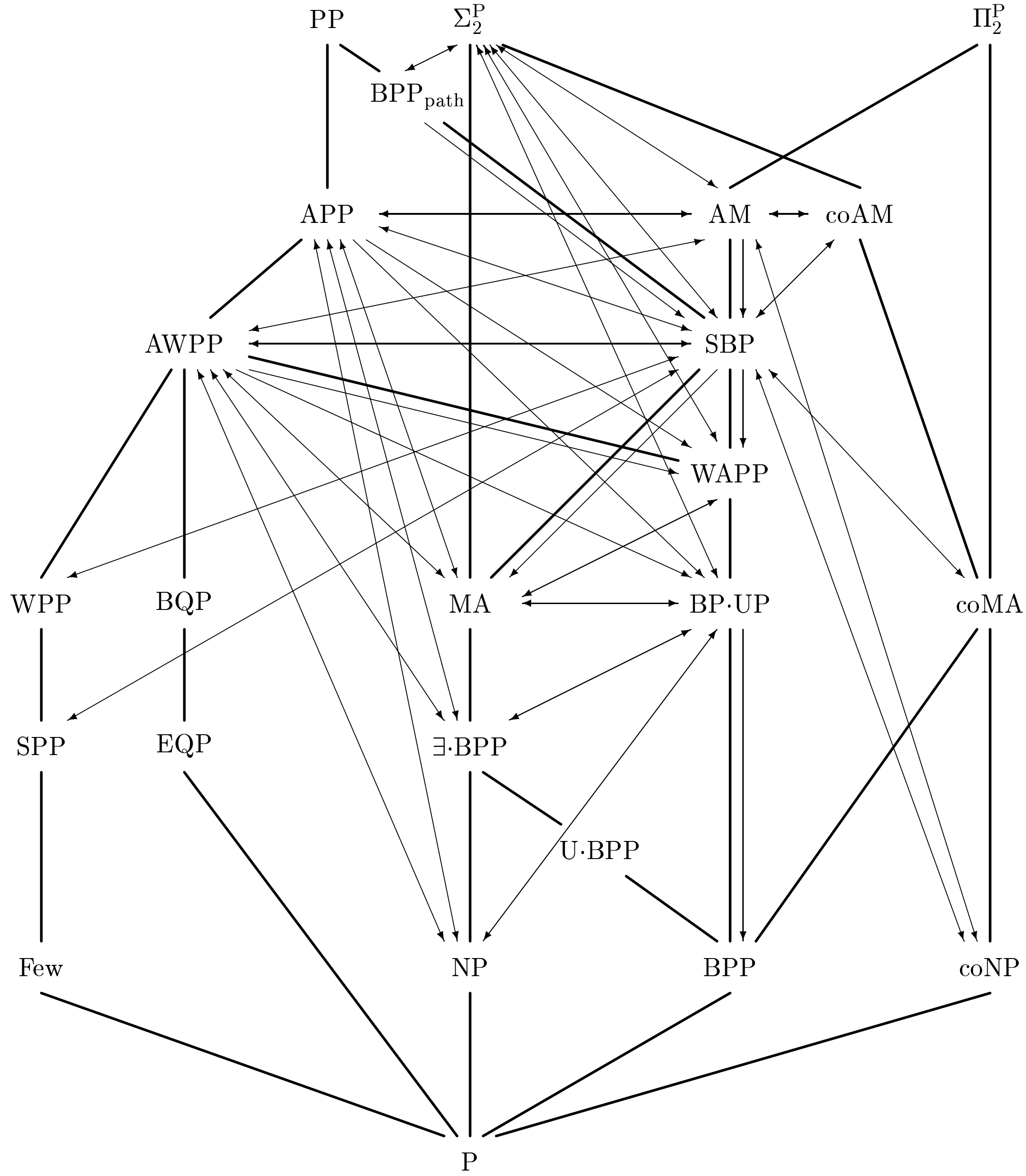

Figure 2: Inclusions and Oracle Separations in the Context of SBP

- inclusions hold from bottom to top
- $\mathcal{C} \longrightarrow \mathcal{D}$ means $\mathcal{C} \not\subseteq \mathcal{D}$ in some relativized world
- $\mathcal{C} \longleftrightarrow \mathcal{D}$ is an abbreviation for $\mathcal{C} \longrightarrow \mathcal{D}$ and $\mathcal{D} \longrightarrow \mathcal{C}$

6 Conclusions and Open Questions

We have seen that with the definition of SBP one meets an interesting complexity class which is located between MA and AM on one hand, and between BPP and $\mathrm{BPP}_{\mathrm{path}}$ on the other hand. By means of collapse consequences and oracle separations we obtained evidence that SBP does not coincide with these classes. In particular we know that SBP is closed under union and in some relativized world it is not closed under complementation. For intersection this question is open, i.e., neither we can prove that SBP is closed under intersection, nor we can construct an oracle where this does not hold. Note that in contrast to GapP, it is not known whether $\#\mathrm{P}$ is closed under subtraction. So the methods showing that PP is closed under intersection [BRS95] cannot be transferred directly to SBP.

Other open questions address the separation of SBP from MA and AM. Can one extend the oracle separations to collapse consequences? Note that Theorem 5.1 shows that such an extension is possible for the separations from BPP and $\mathrm{BPP}_{\mathrm{path}}$. In addition it would be nice to find an unlikely consequence of the assumption $\mathrm{SBP} \subseteq \Sigma_2^{\mathrm{P}}$ (cf. Corollary 5.16 for the respective oracle separation).

In [HHT97] the authors ask whether $\mathrm{BPP}_{\mathrm{path}}$ has complete sets. The same question is also interesting with respect to SBP. Since we expect a negative answer, one should ask whether there is a relativized world where SBP does not have complete sets. Note that there exists an oracle [HH88] where this holds for BPP.

It seems (at least when looking at the definitions) that the classes $\mathrm{BPP}_{\mathrm{path}}$ and AM do not have much in common. However, SBP is contained in both classes. So it would be desirable to know more about the intersection $\mathrm{BPP}_{\mathrm{path}} \cap \mathrm{AM}$. Is it equal to SBP? If so, since $\mathrm{BPP}_{\mathrm{path}}$ and AM are closed under intersection, this would imply that also SBP is closed under intersection. If $\mathrm{BPP}_{\mathrm{path}} \cap \mathrm{AM}$ does not coincide with SBP it would be possible that it coincides at least with SBP's closure under intersection. Definitely, this would be a very nice characterization of the common features of $\mathrm{BPP}_{\mathrm{path}}$ and AM.

In section 4 we considered complexity classes that are defined via GapP and $\#\mathrm{P}$ functions. We have seen that UP is the $\#\mathrm{P}$ counterpart of SPP. Moreover, with the definition of WAPP we introduced the $\#\mathrm{P}$ counterpart of AWPP. Correspondingly, when we restrict Definition 4.3 such that $f \in \#\mathrm{P}$ we meet the $\#\mathrm{P}$ counterpart of the class WPP. What can one say about this class?

Acknowledgements. We thank Klaus W. Wagner for initiating this work and for many helpful discussions. In particular, the idea of the class SBP is due to him. Furthermore, we thank Stephen A. Fenner, Frederic Green, Lane A. Hemaspaandra, Sven Kosub, and Heribert Vollmer for helpful hints.

References

[Bab85] L. Babai. Trading group theory for randomness. In *Proceedings 17th Symposium on Theory of Computing*, pages 421–429. ACM Press, 1985.

[BDG95] J. L. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity I*. Texts in Theoretical Computer Science. Springer Verlag, Berlin Heidelberg, 2nd edition, 1995.

[Bei94] R. Beigel. Perceptrons, PP, and the polynomial hierarchy. *Computational Complexity*, 4:339–349, 1994.

[BHZ87] R. B. Boppana, J. Håstad, and S. Zachos. Does co-NP have short interactive proofs? *Information Processing Letters*, 25(2):127–132, 1987.

[BRS95] R. Beigel, N. Reingold, and D. Spielman. PP is closed under intersection. *Journal of Computer and System Sciences*, 50:191–202, 1995.

[BV97] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

[dLMSS56] K. de Leeuw, E. F. Moore, C. E. Shannon, and N. Shapiro. Computability by probabilistic machines. In C. E. Shannon, editor, *Automata Studies*, volume 34 of *Annals of Mathematical Studies*, pages 183–198. Rhode Island, 1956.

[Fen02] S. Fenner. PP-lowness and a simple definition of AWPP. 2002. To appear.

[FFK94] S. Fenner, L. Fortnow, and S. Kurtz. Gap-definable counting classes. *Journal of Computer and System Sciences*, 48:116–148, 1994.

[FFKL] S. Fenner, L. Fortnow, S. Kurtz, and L. Li. An oracle builder's toolkit. Journal submission. An earlier version appeared in Proceedings 8th Structure in Complexity Theory, pages 120–131, 1993.

[FFKL93] S. Fenner, L. Fortnow, S. Kurtz, and L. Li. An oracle builder's toolkit. In *Proceedings 8th Structure in Complexity Theory*, pages 120–131, 1993.

[For99] L. Fortnow. Relativized worlds with an infinite hierarchy. *Information Processing Letters*, 69(4):309–313, 1999.

[FR99] L. Fortnow and J. Rogers. Complexity limitations on quantum computation. *Journal of Computer and System Sciences*, 59(2):240–252, 1999.

[Gil72] J. Gill. *Probabilistic Turing Machines and Complexity of Computations*. PhD thesis, University of California Berkeley, 1972.

[Gil77] J. Gill. Computational complexity of probabilistic turing machines. *SIAM Journal on Computing*, 6:675–695, 1977.

[GP01] F. Green and R. Pruim. Relativized separation of EQP from P(NP). *Information Processing Letters*, 80(5):257–260, 2001.

[Gup91] S. Gupta. The power of witness reduction. In *Proceedings 6th Structure in Complexity Theory*, pages 43–59. IEEE Computer Society Press, 1991.

[HH88] J. Hartmanis and L. A. Hemachandra. Complexity classes without machines: On complete languages for UP. *Theoretical Computer Science*, 58:129–142, 1988.

[HHT97] Y. Han, L. A. Hemaspaandra, and T. Thierauf. Threshold computation and cryptographic security. *SIAM Journal on Computing*, 26(1):59–78, 1997.

[Ing37] A. E. Ingham. On the difference between consecutive primes. *Quarterly Journal of Mathematics, Oxford Series 8*, pages 255–266, 1937.

[KL82] R. Karp and R. Lipton. Turing machines that take advice. *L'enseignement mathématique*, 28:191–209, 1982.

[Ko82] K.-I. Ko. Some observations on the probabilistic algorithms and NP-hard problems. *Information Processing Letters*, 14:39–43, 1982.

[KW94] J. Köbler and O. Watanabe. New collapse consequences of NP having small circuits. Technical Report 94-11, Fakultät für Mathematik, Universität Ulm, 1994.

[Lau83] C. Lautemann. BPP and the polynomial hierarchy. *Information Processing Letters*, 17:215–217, 1983.

[Li93a] L. Li. On PP-low classes. Technical Report 3, University of Chicago, 1993. available at http://www.cs.uchicago.edu/research/publications/techreports/TR-93-03.

[Li93b] L. Li. *On the Counting Functions*. PhD thesis, University of Chicago, 1993. available at http://www.cs.uchicago.edu/research/publications/techreports/TR-93-12.

[OH93] M. Ogiwara and L. Hemachandra. A complexity theory of feasible closure properties. *Journal of Computer and System Sciences*, 46:295–325, 1993.

[Pap94] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, Reading, MA, 1994.

[Sch89] U. Schöning. Probabilistic complexity classes and lowness. *Journal of Computer and System Sciences*, 39:84–100, 1989.

[Sip83] M. Sipser. A complexity theoretic approach to randomness. In *Proceedings of the 15th Symposium on Theory of Computing*, pages 330–335, 1983.

[Val76] L. G. Valiant. Relative complexity of checking and evaluation. *Information Processing Letters*, 5:20–23, 1976.

[Ver92] N. K. Vereshchagin. On the power of PP. In *Proceedings 7th Structure in Complexity Theory*, pages 138–143. IEEE Computer Society Press, 1992.

[Yao85] A. C. C. Yao. Separating the polynomial-time hierarchy by oracles. In *Proceedings 26th Foundations of Computer Science*, pages 1–10. IEEE Computer Society Press, 1985.

[Zac82] S. Zachos. Robustness of probabilistic computational complexity classes under definitional perturbations. *Information & Control*, 54:143–154, 1982.