JOURNAL OF COMPUTER AND SYSTEM SCIENCES **52**, 308–327 (1996)
ARTICLE NO. 0024

RAINER SCHULER[†]

Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, D-89069 Ulm, Germany

AND

TOMOYUKI YAMAKAMI[‡]

Department of Computer Science, University of Toronto, Toronto, Ontario, Canada M5S 1A4

Levin introduced an average-case complexity measure, based on a notion of ''polynomial on average,'' and defined ''average-case polynomial-time many-one reducibility'' among randomized decision problems. We generalize his notions of average-case complexity classes, Random-NP and Average-P. Ben-David *et al.* use the notation of $\langle \mathcal{C}, \mathcal{F} \rangle$ to denote the set of randomized decision problems (L, μ) such that L is a set in $\mathcal{C}$ and μ is a probability density function in $\mathcal{F}$. This paper introduces Aver$\langle \mathcal{C}, \mathcal{F} \rangle$ as the class of randomized decision problems (L, μ) such that L is computed by a type-$\mathcal{C}$ machine on μ-average and μ is a density function in $\mathcal{F}$. These notations capture all known average-case complexity classes as, for example, Random-NP $= \langle$NP, P-comp$\rangle$ and Average-P $=$ Aver$\langle$P, $*\rangle$, where P-comp denotes the set of density functions whose distributions are computable in polynomial time, and $*$ denotes the set of all density functions. Mainly studied are polynomial-time reductions between randomized decision problems: many–one, deterministic Turing and nondeterministic Turing reductions and the average-case versions of them. Based on these reducibilities, structural properties of average-case complexity classes are discussed. We give average-case analogues of concepts in worst-case complexity theory; in particular, the polynomial time hierarchy and Turing self-reducibility, and we show that all known complete sets for Random-NP are Turing self-reducible. A new notion of ''real polynomial-time computations'' is introduced based on average polynomial-time computations for arbitrary distributions from a fixed set, and it is used to characterize the worst-case complexity classes Δ_k^p and Σ_k^p of the polynomial-time hierarchy. © 1996 Academic Press, Inc.

1. INTRODUCTION

The classical complexity theory of NP-completeness is based on the worst-case analysis of algorithms. A probabilistic analysis has been applied so far only to specific algorithms typically with respect to the uniform distribution for each length of inputs. Levin [21] gave a general framework to perform *average-case analysis* in a way that allows us to discuss many questions of worst-case complexity theory in a more general setting. The average-case analysis considers *randomized problems*, namely, pairs of a decision problem and a probability density (or distribution) function which assigns probabilities to instances.

In [21] Levin implicitly defined the average-case complexity classes "Average-P" and "Random-NP" as analogues of the worst-case complexity classes P and NP, respectively. Roughly speaking, Average-P (called AP in [16, 37]) is the class of randomized problems which are, on the average, solvable in polynomial time for a given distribution on the input, and Random-NP (called DistNP in [13, 4, 29], RNP in [16], and DNP in [37]) is the class of problems in NP, together with polynomial time computable distribution on the inputs. A stimulating question of whether Average-P contains the class Random-NP, was raised by Levin in his very terse papers [21, 22]. Up to now, this question has not been solved.

In recent literature [13, 16, 21, 22, 34–36], several randomized decision problems have been shown to be polynomial-time complete for Random-NP with respect to several different reducibilities. Average-case analysis is very sensitive to the choice of distributions on the instances, since, for example, if a density function μ decreases faster than $2^{-|x|}$ with the length of the instances x, then all NP-complete problems are solvable in time polynomial on μ-average. Even if we stick to "reasonable" distributions, "fast on average" algorithms have been found even for natural NP-complete problems. Fox example, the *satisfiability problem*, SAT with a natural distribution [9], the *graph* 3-*colorability problem* with a natural distribution [40], and the *Hamiltonian circuit problem* with a natural distribution and edge probability $\frac{1}{2}$ [7] are all in Average-P. In the decade since Levin's fundamental papers, several results have revealed the significant role of distributions on the average-case complexity (see, e.g., [37]). However, it is still open whether every NP-complete problem is complete

[*] A preliminary version of this paper appeared in the Proceedings of the 12th Conference on Foundations of Software Technology and Theoretical Computer Science, New Delhi, December 1992, Lecture Notes in Computer Science, Vol. 652 (1992), pp. 128–139.

[†] Supported by the Deutsche Forschungsgemeinschaft, Grant Scho 302/2-1.

[‡] This work was done while the author was a guest at the Universität Ulm from April to August in 1991.

0022-0000/96 $18.00

for Random-NP for some reasonable distribution. In this paper, our main interest lies in the analysis of structural properties of average-case complexity classes. We extend well-known notions of worst-case complexity theory such as time- and space-bounded computation, nondeterministic Turing reducibility, self-reducibility, and relativization to average-case analysis. Taking a set of distributions, we further discuss the (maximal) average-case complexity of problems under every distribution chosen from the given set.

Ben-David *et al.* [4] introduced the notation $\langle \mathscr{C}, \mathscr{F} \rangle$ to discuss the average-case complexity of decision problems in a general setting. The class $\langle \mathscr{C}, \mathscr{F} \rangle$ contains all randomized problems (L, μ) (recall that a randomized problem here is always a pair of a decision problem and a density function on the instances) such that L is in the complexity class $\mathscr{C}$ and μ is in a class of density functions $\mathscr{F}$. Random-NP can simply be denoted as $\langle$NP, P-comp$\rangle$, where P-comp (polynomial-time computable) denotes the set of density functions whose distributions can be approximated by a deterministic Turing machine in polynomial time.

Schapire [29] has given a different characterization of Levin's notion of "polynomial on μ-average," which will be used in this paper. He has shown that a function g from strings to nonnegative real numbers is *polynomial on μ average*, in Levin's sense, if and only if there exists a polynomial p such that $\text{Prob}_\mu[\{x \mid g(x) > p(|x| \cdot r)\}] < 1/r$ for any positive real number r. Schapire's characterization is intuitive and can be easily generalized to the notion of "f on μ-average" by replacing the polynomial p by a function f which is defined on nonnegative real numbers.

This paper introduces the notation Aver$\langle \mathscr{C}, \mathscr{F} \rangle$ to denote the set of all problems (L, μ) such that μ is in $\mathscr{F}$, and L is recognized by a "type-$\mathscr{C}$ on μ-average" machine. Here, $\mathscr{C}$ denotes a time- or space-bounded complexity class, and a *type-$\mathscr{C}$ on μ-average machine* is intuitively a Turing machine which respects, on μ-average, this time- or space-bound, respectively. By using this notation, for example, Average-P, the set of problems solvable in average polynomial time, can be denoted as Aver$\langle$P, $*\rangle$, where $*$ is the set of all density functions. Levin's most intriguing open question is rephrased as whether $\langle$NP, P-comp$\rangle \subseteq$ Aver$\langle$P, $*\rangle$ holds.

A central concept in this paper is different types of *polynomial-time reductions* which are generalizations of the polynomial-time many–one reducibility defined by Levin. Roughly speaking, a reduction function between two randomized problems reduces a set of strings to another set of strings and also satisfies the so-called *domination condition* between density functions, which ensures that likely instances are mapped to likely instances. Since Levin's work, random many–one reducibility [4, 34] and polynomial-time deterministic Turing reducibility [4, 18] have been defined and studied. All those reducibilities can be extended by allowing their many–one reduction functions or oracle Turing reduction machines to be polynomial-time bounded on μ-average. We study properties of many–one reduction, deterministic Turing reduction and their average-case extensions.

In this paper, a *polynomial-time nondeterministic Turing reduction* between randomized decision problems is defined in a way that captures both deterministic Turing reducibility and random many–one reducibility. Let M be a nondeterministic oracle Turing machine, E a set, and p a polynomial. By $Q(M, E, x, y)$, we denote the set of strings which are queried by M with oracle E on input x on computation path y, and $\text{Acc}(M, E, x)$ (resp. $\text{Rej}(M, E, x)$) denotes the set of (codes of) accepting (resp. rejecting) computation paths given by M with E on input x. We introduce the density function μ' that is induced from μ, M and E as: $\mu'(x, y) = \mu(x)/|\text{Acc}(M, E, x)|$ if y is an accepting path of M^E on input x; $\mu'(x, y) = \mu(x)/|\text{Rej}(M, E, x)|$ if there is no accepting paths and y is a rejecting path; otherwise, $\mu'(x, y) = 0$. The machine M polynomial-time nondeterministic (on μ-average) Turing reduces (D, μ) to (E, v) if $D = L(M^E)$, M^E is polynomial (on μ-average) time bounded, and there exists a density function υ which polynomially (on μ-average) dominates μ' such that $v(z) \geqslant \text{Prob}_\upsilon[\{(x, y) \mid z \in Q(M, E, x, y)\}]$ for all strings z.

A randomized decision problem (L, μ) is said to be *$\langle \mathscr{C}, \mathscr{F} \rangle$-complete* if (L, μ) is in $\langle \mathscr{C}, \mathscr{F} \rangle$, and every problem in $\langle \mathscr{C}, \mathscr{F} \rangle$ is polynomial-time many–one reducible to (L, μ). Levin has first proven that the *randomized tiling problem* is $\langle$NP, P-comp$\rangle$-complete [21]. Another important $\langle$NP, P-comp$\rangle$-complete problem is the *randomized bounded halting problem* [16]. Both problems are typical NP-complete problems, together with some natural distributions. Note that if (L, μ) is $\langle$NP, P-comp$\rangle$-complete, then L must be NP-complete. As we noted before, however, it is not known whether every NP-complete set L has an appropriate density function μ which forces (L, μ) to be $\langle$NP, P-comp$\rangle$-complete.

This paper extends the notion of average-case complexity further to discuss questions raised in structural complexity theory. An interesting property of complete sets is the (*Turing*) *self-reducibility* [24], and, in worst-case complexity theory, many known NP-complete sets are actually self-reducible. Analogously, we can introduce a *self-reducibility* into our average-case analysis. A randomized decision problem is called self-reducible if it is polynomial-time Turing reducible to itself, while querying only strings of length smaller than the input. We show the existence of $\langle \Sigma_k^{\mathrm{p}}$, P-comp$\rangle$-complete sets which are self-reducible by demonstrating that the kth level of the randomized bounded halting problem $(\text{RH}^k, \mu_{\text{RH}})$ is $\langle \Sigma_k^{\mathrm{P}}$, P-comp$\rangle$-complete and also self-reducible. From the fact that most of the known $\langle$NP, P-comp$\rangle$-complete problems are p-isomorphic [38], they turn out to be self-reducible.

We study a relativization of Aver$\langle$P, $\mathscr{F}\rangle$ and Aver$\langle$NP, $\mathscr{F}\rangle$ which is done in a similar fashion as we obtain relativized classes P^B and NP^B from P and NP, respectively. To be more precise, recall that P^B (resp. NP^B) is equivalent to the collection of sets which are polynomial-time deterministic (resp. nondeterministic) Turing reducible to B. Analogously, we define a class Aver$\langle \mathrm{P}, \mathscr{F}\rangle^{(E,\nu)}$ (resp. Aver$\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)}$) as a collection of randomized decision problems (D, μ), with $\mu \in \mathscr{F}$, which are deterministic (resp. nondeterministic) polynomial-time on μ-average Turing reducible to (E, ν).

We demonstrate that $\langle \mathrm{NP}, \mathscr{F}\rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, \mathscr{F}\rangle$ and $\langle \mathrm{NP}, \mathscr{F}\rangle \nsubseteq \mathrm{Aver}\langle \mathrm{P}, \mathscr{F}\rangle$ in some relativized worlds, using the technique of Baker *et al.* [1]. These contradictory results imply that any proof technique which can be relativized will not solve Levin's question.

Similar to the Meyer–Stockmeyer polynomial-time hierarchy [25] in worst-case complexity theory, we give a precise definition of its average-case analogue, the *average polynomial-time hierarchy*, founded on a relativization of Aver$\langle$P, $\mathscr{F}\rangle$ and Aver$\langle$NP, $\mathscr{F}\rangle$. Let Aver$\langle \Delta^{\mathrm{p}}_k, \mathscr{F}\rangle$ denote the kth level of the average polynomial-time hierarchy. This paper considers the general question of whether $\langle \Sigma^{\mathrm{p}}_k, \text{P-comp}\rangle \subseteq \mathrm{Aver}\langle \Delta^{\mathrm{p}}_k, *\rangle$. We show that this is not the case unless every tally set in Σ^{p}_k is in Δ^{p}_k. Hence, we conjecture that $\langle \Sigma^{\mathrm{p}}_k, \text{P-comp}\rangle \nsubseteq \mathrm{Aver}\langle \Delta^{\mathrm{p}}_k, *\rangle$ since it seems very likely that tally sets exist in $\Sigma^{\mathrm{p}}_k - \Delta^{\mathrm{p}}_k$.

Finally we discuss a tie between worst-case and average-case complexity classes. Note that every polynomial-time computation is polynomial on μ-average even if μ is chosen by some (powerful) adversary, that is, every set in P is computable by a deterministic Turing machine which is polynomial-time bounded on μ-average for every density function μ. In other words, if A is in P then $(A, \mu) \in \mathrm{Aver}\langle \mathrm{P}, *\rangle$ for every μ. On the other hand, P can be expected to be the largest class which satisfies this property. This is indeed true [23]. This observation can be generalized as follows. For a class $\mathscr{C}$, we define the *real $\mathscr{C}$ over $\mathscr{F}$*, denoted by $\mathscr{C}_{\mathscr{F}}$, to be the class of all sets A such that $(A, \mu) \in \mathrm{Aver}\langle \mathscr{C}, *\rangle$ for every density function μ in $\mathscr{F}$. For example, $\mathrm{P} \subseteq \mathrm{P}_{\text{P-comp}} \subseteq \mathrm{E}$, and Levin's question of whether $\langle \mathrm{NP}, \text{P-comp}\rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, *\rangle$ can be now stated as whether $\mathrm{NP} \subseteq \mathrm{P}_{\text{P-comp}}$. We show that $\Delta^{\mathrm{p}}_k = \Delta^{\mathrm{p}}_{k\,\text{REC-comp}}$ and $\Sigma^{\mathrm{p}}_k = \Sigma^{\mathrm{p}}_{k\,\text{REC-comp}}$ for all levels $k > 0$, where REC-comp denotes the collection of all recursive density functions. As a particular case, we can show that $\mathrm{P} = \mathrm{P}_{\text{E-comp}}$.

We show that, relative to random oracle, $\mathrm{NP}_{\text{P-comp}}$ is different from $\mathrm{P}_{\text{P-comp}}$ with probability 1.

2. PRELIMINARIES

In this paper, we follow the standard definitions and notations of complexity theory; see, e.g., [10, 2]. Here we briefly present necessary notations and notions.

Fix $\Sigma = \{0, 1\}$. By Σ^* we denote the set of all strings over Σ, and Σ^n denotes the set of all strings of length n. The set Σ^* admits the standard lexicographic order: $\varepsilon < 0 < 1 < 00 < 01 < \cdots$, where ε denotes the empty string. The successor and predecessor of x in this order are denoted by x^+ and x^-, respectively. A subset A of Σ^* is often identified with its *characteristic function*, i.e., $A(x) = 1$ if $x \in A$, and $A(x) = 0$ otherwise. The cardinality of a set A and the length of a string $x \in \Sigma^*$ are respectively denoted by $|A|$ and $|x|$. The *complement* of a set A is denoted by $\bar{A}$. For two sets A and B, let $A \triangle B$ denote the *symmetric difference* of A and B and let $A \oplus B$ be the *disjoint union* of A and B.

Let $\mathbb{N}$ be the set of all *nonnegative integers* and let $\mathbb{R}^+$ be the set of all *nonnegative real numbers*. A nonnegative integer can be identified with its binary representation, and we often refer to strings as nonnegative integers. Especially, Σ^* is identified with the set $\mathbb{D} = \{m/2^n \mid m < 2, m, n \in \mathbb{N}\}$ of *nonnegative dyadic rational numbers*, i.e., a string $d_1 d_2 \cdots d_n$ in Σ^* is identified with the number $d_1 2^{-1} + d_2 2^{-2} + \cdots + d_n 2^{-n}$ in $\mathbb{D}$.

A set is *tally* if it is a subset of $\{0\}^*$, and a set A is (*polynomially*) *sparse* if there is a polynomial p such that $|A \cap \Sigma^n| \leqslant p(n)$ for all $n \in \mathbb{N}$. Denote by TALLY the class of all tally sets and by SPARSE the class of all sparse sets.

A formal definition of *Turing machines* with semi-infinite tapes is given in, for example, [11, 10, 17, 2], and we assume the reader's familiarity with it. In this paper, we are interested in only *resource bounded* algorithms and assume that all Turing machines are designed in such a way that all computation paths have the same length. Therefore, for every oracle Turing machine M and a set A, the running *time* of the machine M with oracle A on input x, denoted by $\mathrm{Time}^A_M(x)$, is simply defined to be the length of some possible computation, and the space complexity, $\mathrm{Space}^A_M(x)$, is defined to be the maximum, over all configurations of M with oracle A on input x, of the number of tape-squares in use.

A (non-)deterministic oracle Turing machine *accepts* an input x if there is an accepting computation of M with oracle A on input x; otherwise, M *rejects* x. If M is probabilistic, then M *accepts* x if $\mathrm{Prob}_M[M$ on input x halts in an accepting state$] > \frac{1}{2}$; otherwise, M *rejects* x, where $\mathrm{Prob}_M[Q(M)]$ denotes the probability that $Q(M)$ holds.

Let $\mathrm{Acc}(M, A, x)$ denote the set of (codes of) *accepting* computations of M on input x with oracle A, and, similarly, $\mathrm{Rej}(M, A, x)$ denotes that of *rejecting* computation paths. Let $Q(M, A, x, y)$ be the set of strings queried by M with A on input x on computation path y. If M is deterministic, then we simply denote by $Q(M, A, x)$ the set of all strings queried by M on input x with oracle A.

As usual, $L(M, A)$ denotes the set of strings accepted by M with oracle A, and we simply say that M^A *computes* a set B if $B = L(M, A)$. For a machine M, $M^A(x)$ denotes the output of a computation of M on input x. For a deterministic Turing machine M with an output tape (also called a *transducer*), M *computes* a function f if $f(x) = M^A(x)$ for all $x \in \Sigma^*$.

For any function t on $\mathbb{N}$, a Turing machine M with oracle A is *t-time bounded* (resp. *t-space bounded*) if $\mathrm{Time}^A_M(x) \leqslant t(|x|)$ (resp. $\mathrm{Space}^A_M(x) \leqslant t(|x|)$) for all x. Let DTIME(t), NTIME(t), and DSPACE(t) denote the class of all sets computable by deterministic t-time bounded, nondeterministic t-time bounded, and deterministic t-space bounded Turing machines, respectively. Finally, let BPTIME(t) denote the class of sets B computable by t-time bounded probabilistic Turing machines with bounded error probability, i.e., there is a constant ε, $0 \leqslant \varepsilon < \frac{1}{2}$ such that $\mathrm{Prob}_M[M(x) = B(x)] < 1 - \varepsilon$ holds for all x.

We assume that the reader is familiar with standard notations of complexity classes, such as P, NP, Δ^{p}_k, Σ^{p}_k, PH (the polynomial-time hierarchy), BPP, PSPACE, E (linear exponential time), NE, EXP (exponential time), NEXP, and FP (polynomial-time computable functions). Moreover, we let $\mathrm{ESPACE} = \bigcup_{k>0} \mathrm{DSPACE}(2^{kn+k})$ and $\mathrm{BPE} = \bigcup_{k>0} \mathrm{BPTIME}(2^{kn+k})$.

A function f on Σ^* is *p-honest* if there exists a polynomial p such that $p(|f(x)|) \geqslant |x|$ for any string x, and f is *p-invertible* if there is a function g in FP such that $g \circ f(x) = x$ for all x. A function f from $\mathbb{N}$ to $\mathbb{R}^+$ is called *unbounded* if $\lim_{k \to \infty} \sup_{n>k} f(n) = \infty$.

A set D is *polynomial-time many-one reducible* to a set E if there exists a function f in FP such that, for all x, $x \in D$ if and only if $f(x) \in E$. A set D is *polynomial-time many–one complete* for a class $\mathscr{C}$ if $D \in \mathscr{C}$ and every set in $\mathscr{C}$ is polynomial-time many–one reducible to D. We simply say that D is *$\mathscr{C}$-complete* for a class $\mathscr{C}$ if it is polynomial-time many–one complete for $\mathscr{C}$.

In the following, we use the following pairing function $\langle\ ,\ \rangle$ [27], from $\Sigma^* \times \Sigma^*$ onto Σ^*, that is defined as follows: $\langle x, y \rangle = d(x)\, y$ if $|y| \leqslant 1$; otherwise $\langle x, y \rangle = d(x)\, i_2[(y^-)^-]$, where $d(\varepsilon) = \varepsilon$, $d(0x) = 00d(x)$, $d(1x) = 11d(x)$, $i_2[0x] = 01x$, and $i_2[1x] = 10x$ for all x. This pairing function is monotone, i.e., $x \leqslant x'$ and $y \leqslant y'$ imply $\langle x, y \rangle \leqslant \langle x', y' \rangle$, and it is computable in linear-time in the lengths of x and y. Moreover, it holds that $2\,|x| + |y| \leqslant |\langle x, y \rangle| \leqslant 2\,|x| + |y| + 1$ for all x and y. This paring function is recursively generalized to a bijection from $(\Sigma^*)^k$ onto Σ^* as $\langle x_1, x_2, ..., x_k \rangle = \langle x_1, \langle x_2, ..., x_k \rangle\rangle$.

A function $\mu\colon \Sigma^* \to [0, 1]$ is called a *density function* if $\sum_{x \in \Sigma^*} \mu(x) = 1$, and its corresponding *distribution* μ^* is given by $\mu^*(x) = \sum_{z \leqslant x} \mu(z)$. To avoid confusion, we remark here that density functions were also called "probability distributions" in [6, 37, 38] or "probability functions" in [16], and distributions were called "probability distributions" in [13, 16]. Let μ_n denote the *conditional* density function for strings of length n, i.e., $\mu_n(x) = \mu(x)/\sum_{y:|y|=n} \mu(y)$ whenever $\sum_{y:|y|=n} \mu(y) \neq 0$; otherwise 0. We use the notations $\mu(x, y)$ and $\mu(x, y, z)$ to denote $\mu(\langle x, y \rangle)$ and $\mu(\langle x, y, z \rangle)$, respectively.

For a finite domain D, the *uniform* density function on D is defined as $1/|D|$ for every x in D. The *standard* density function ν_{st} on Σ^* is defined as $\nu_{\mathrm{st}}(x) = (6/\pi^2) \cdot (|x| + 1)^{-2} \cdot 2^{-|x|}$ for all x. Although the standard distribution is called "uniform" in, e.g., [4]; actually only its conditional distribution is uniform for all lengths n. We note that there are other ways to define a "standard" density function; see Gurevich [16] for a discussion. A density function μ is called *flat* if $\mu(x) \leqslant 2^{-|x|^\varepsilon}$ for some constant $\varepsilon > 0$ [16], and μ is *positive* if $\mu(x) > 0$ for all x.

For a density function μ, we use the notation $\mathrm{Prob}_\mu[\{x \mid Q(x)\}]$ to denote the probability that property $Q(x)$ holds, where x is chosen randomly according to μ. For set A, let $\mu(A)$ denote $\sum_{x \in A} \mu(x)$.

In his papers [21, 22], Levin considers "polynomial-time computable" distributions as reasonable to discuss the average time-complexity of NP problems. Later a more generalized notion, i.e., "polynomial-time samplable" distributions, has been proposed [4]. This paper follows Gurevich [16] to define the notion of "polynomial-time computability" of distributions.

Definition 2.1 [20, 16]. Let g be a distribution from Σ^* to [0,1] and f be a function on $\mathbb{N}$:

1. g is *f-time computable* (resp. *f-space computable*) if there exists a deterministic f-time (resp. f-space) bounded transducer T such that, for all $x \in \Sigma^*$ and all $k \geqslant 0$,

$$|g(x) - T(x01^k)| \leqslant 2^{-k}.$$

2. g is *$\mathscr{F}$-time computable* (resp. *$\mathscr{F}$-space computable*) for a class $\mathscr{F}$ if there exists a function $f \in \mathscr{F}$ such that g is f-time (resp. f-space) computable.

3. Let L-comp, P-comp, PSPACE-comp, and EXP-comp denote the class of density functions whose *distributions* are logarithmic-space, polynomial-time, polynomial-space, and exponential-time computable, respectively. Let REC-comp denote the set of all *recursive* density functions.

We remark that if a distribution is $\mathscr{F}$-time computable, then the density function is also $\mathscr{F}$-time computable. The converse, however, may not always hold since it is shown in, e.g., [16] that if $\mathrm{P} \neq \mathrm{NP}$, then there is a polynomial-time computable density function such that its associated distribution can not be computed in polynomial time.

Note also that Ben-David *et al.* [4] use a stronger definition of polynomial-time computability; i.e., for all x, the value of $\mu^*(x)$ is exactly computed by some polynomial-time bounded transducer. Let SP-comp denote the class of these density functions. Naturally, if $g \in$ SP-comp, then $g(x)$ is either 0 or greater than $2^{-p(|x|)}$ for some polynomial p. However, for every $\mu \in$ P-comp, there exists a positive $\nu \in$ SP-comp such that $\nu(x)$ has at most $4 + 2\,|x|$ binary digits and $4\nu(x) > \mu(x)$ for all x [16]. Also, if $\mu(x) > 2^{-p(|x|)}$

for some polynomial p, then there exists a total, one–one, p-invertible function $f \in \mathrm{FP}$ such that, for all x, $4 \cdot 2^{-|f(x)|} \leqslant \mu(x) \leqslant 20 \cdot 2^{-|f(x)|}$ [38].

A central concept in average-case complexity is "a computation being time (space) bounded on the average for some distribution." For a discussion on the definition of "polynomial on μ-average," see [13, 16]. This paper uses a characterization of polynomial on μ-average given by Schapire [29], since it can be easily extended to the notion of "f on μ-average" for an arbitrary function f.

Definition 2.2 (cf. [29]). Let f be a function on $\mathbb{R}^+$ and let μ be a density function. A function $g\colon \Sigma^* \to \mathbb{R}^+$ is *f on μ-average* if $\mathrm{Prob}_\mu[\{x \mid g(x) > f(|x| \cdot r)\}] < 1/r$ for any real number $r > 0$. For a class $\mathscr{C}$ of functions, g is *$\mathscr{F}$ on μ-average* if there exists a function $f \in \mathscr{F}$, and g is f on μ-average.

It immediately follows from this definition that increasing the value of r also increases the probability weight of the set of strings x with the property that $g(x) \leqslant f(|x| \cdot r)$, which is $1 - 1/r$. One significant consequence of this fact is that if g is f on μ-average, then $g(x) \leqslant f(|x|/\mu(x))$ for all x with $\mu(x) > 0$. This fact can be seen as follows. Suppose that there exists an x_0 such that $g(x_0) > f(|x_0|/\mu(x_0))$ and $\mu(x_0) > 0$. Choose $r = 1/\mu(x_0)$. Then $\mu(x_0) \leqslant \mathrm{Prob}_\mu[\{x \mid g(x) > f(|x| \cdot r)\}] < 1/r = \mu(x_0)$, a contradiction.

Definition 2.2 allows us to discuss an arbitrary set of functions which are bounded on average. For instance, if $\mathscr{F} = \{\lambda x \cdot (x^k + k) \mid k > 0\}$ (the set of polynomials), then we obtain the notion of *polynomial on μ-average* as defined in [21, 22] and used in [13, 29, 16, 4, 37]. Similarly, if $\mathscr{F} = \{\lambda x \cdot (c \log x + d) \mid c, d \geqslant 0\}$ (the set of logarithmic functions), then the above definition yields the notion of *logarithmic on μ-average* as defined in [4] and also used in [14]. The following lemma gives the justification of the above definition.

Lemma 2.3. [29]. *Let g be a function from Σ^* to $\mathbb{R}^+$:*

1. *The function g is polynomial on μ-average if and only if it is polynomial on μ-average in the sense of Levin, i.e., for some $\delta > 0$,*

$$\sum_{x : |x| > 0} \frac{g(x)^\delta}{|x|}\, \mu(x) < \infty.$$

2. *The function g is logarithmic on μ-average if and only if, for some $\delta > 0$,*

$$\sum_{x : |x| > 0} \frac{(2^{g(x)})^\delta}{|x|}\, \mu(x) < \infty.$$

Proof. The proof of the claim (1) follows [29]. Without loss of generality, assume that $g(\varepsilon) = 0$ and that g is $\lambda n \cdot c n^k$ on μ-average for some k. By definition, for any real number $r > 0$, $\mathrm{Prob}_\mu[\{x \mid g(x) > c(r\,|x|)^k\}] < 1/r$. This indicates that $\mathrm{Prob}_\mu[\{x \mid g(x) > c(r\,|x|^2)^k\}] < 1/r$. In other words, $\mathrm{Prob}_\mu[\{x \mid g(x)^{1/2k}\,|x|^{-1} > c^{1/2k} r^{1/2}\}] < 1/r$. For every integer $t > 0$, let $r = t^2/c^{1/k}$, and thus $\mathrm{Prob}_\mu[\{x \mid g(x)^{1/2k}\,|x|^{-1} > t\}] < c^{1/k}/t^2$. Then,

$$\begin{aligned}
&\sum_{x : |x| > 0} \frac{g(x)^{1/2k}}{|x|}\, \mu(x) \\
&\quad \leqslant \sum_{t=1}^{\infty} \mathrm{Prob}_\mu\left[\left\{x \,\middle|\, t - 1 < \frac{g(x)^{1/2k}}{|x|} \leqslant t\right\}\right] \cdot t \\
&\quad = \sum_{t=0}^{\infty} \mathrm{Prob}_\mu\left[\left\{x \,\middle|\, \frac{g(x)^{1/2k}}{|x|} > t\right\}\right] < \infty.
\end{aligned}$$

Conversely, assume that $\sum_{x : |x| > 0} g(x)^\delta\,|x|^{-1}\,\mu(x) \leqslant N$ for some number $N \geqslant 1$ and choose a positive integer k such that $1/k < \delta$. Markov's inequality enables us to show that $\mathrm{Prob}_\mu[\{x \mid g(x)^{1/k}\,|x|^{-1} > r \cdot N\}] < 1/r$ for any real number $r > 0$. This yields $\mathrm{Prob}_\mu[\{x \mid g(x) > (rN\,|x|)^k\}] < 1/r$. Hence, g is $\lambda n \cdot (Nn)^k$ on μ-average.

To see the claim (2), note that the function g is logarithmic on μ-average if and only if $\lambda x \cdot 2^{g(x)}$ is polynomial on μ-average. ▌

The next definition follows the notion of "polynomial domination" introduced by Levin [21, 22]. The domination condition between density functions is crucial in the definition of reducibilities among randomized decision problems in Section 4. Intuitively, it ensures that if an algorithm is fast (e.g., polynomial) on average for a distribution μ, then this algorithm is also fast on average for all distributions which are dominated by μ.

Definition 2.4. Let μ_1, μ_2, and ν be density functions.

1. Let t and $\mathscr{T}$ be a function and a set of functions from Σ^* to $\mathbb{R}^+$, respectively. The density function μ_2 *t-dominates* μ_1 if $\mu_2(x) \cdot t(x) \geqslant \mu_1(x)$ for all $x \in \Sigma^*$, and μ_2 *$\mathscr{T}$-dominates* μ_1 if there exists a function $t' \in \mathscr{T}$ such that μ_2 t'-dominates μ_1.

2. Let t and $\mathscr{T}$ be a function and a set of functions on $\mathbb{R}^+$, respectively. The density function μ_2 *t on ν-average dominates* μ_1 if there exists a function t', from Σ^* to $\mathbb{R}^+$, such that t' is t on ν-average and μ_2 t'-dominates μ_1, and μ_2 *$\mathscr{T}$ on ν-average dominates* μ_1 if there exists a function t in $\mathscr{T}$ such that μ_2 t on ν-average dominates μ_1.

This definition enables us to consider logarithmic, polynomial, and exponential domination and domination on μ-average. For example, if $\mathscr{T}$ is the set of polynomials, then μ_2 *polynomially dominates* μ_1, denoted by $\mu_1 \preccurlyeq^{\mathrm{p}} \mu_2$, and μ_2 *polynomial on ν-average dominates* μ_1, denoted by $\mu_1 \preccurlyeq^{\mathrm{p},\nu} \mu_2$, respectively. In [16], polynomial domination and polynomial on μ-average domination are called "domination" and "weakly domination," respectively.

We will now give a general definition of "time- and space-bounded on average" for Turing machines.

DEFINITION 2.5. Let M be an oracle Turing machine, A a set, μ a density function, and let t and $\mathscr{T}$ be a function and a set of functions on $\mathbb{R}^+$, respectively. The machine M^A is *t-time bounded on μ-average* if the function Time_M^A is t on μ-average, and M^A is *$\mathscr{T}$-time bounded on μ-average* if there exists a function $t \in \mathscr{T}$ such that M^A is t-time bounded on μ-average. The notions of *t-space bounded on μ-average* and *$\mathscr{T}$-space bounded on μ-average* are defined similarly by using Space_M^A instead of Time_M^A.

For instance, if $\mathscr{T}$ is the set of polynomials, then we say that M^A is *polynomial-time* (*or polynomial-space*) *bounded on μ-average* as in [21, 22]. If a function f is computed by a deterministic transducer which is polynomial-time bounded on μ-average, we say that f is *computable in time polynomial on μ-average.*

We observe that the quantifier characterization of non-deterministic and probabilistic Turing machines holds also in average-case setting. Recall that, for instance, all sets in NP can be characterized by an existential quantifier and deterministic Turing machines as follows: a set D is in NP if and only if there exist a polynomial p and a polynomial-time deterministic Turing machine M such that $D = \{x \mid \exists y[\,|y| = p(|x|) \wedge \langle x, y\rangle \in L(M)]\}$ [41].

PROPOSITION 2.6. *For every set D and every density function μ, the following statements are equivalent*:

1. *There exists a nondeterministic Turing machine M such that $D = L(M)$ and M is polynomial-time bounded on μ-average.*

2. *There exists a function p from Σ^* to $\mathbb{N}$ and a polynomial-time bounded deterministic Turing machine Msuch that p is computable in time polynomial on μ-average and $D = \{x \mid \exists y[\,|y| = p(x) \wedge \langle x, y\rangle \in L(M)]\}$.*

PROPOSITION 2.7. *For every set D, every density function μ, the following statements are equivalent*:

1. *There exists a bounded-error probabilistic Turing machine M such that $D = L(M)$ and M is polynomial-time bounded on μ-average.*

2. *For every function q that is computable in time polynomial on μ-average, there exists a probabilistic Turing machine M such that M is polynomial-time bounded on μ-average, $D = L(M)$ and $\mathrm{Prob}_M[M(x) = D(x)] \geqslant 1 - 2^{-q(x)}$ for all x.*

3. *For every function q that is computable in time polynomial on μ-average, there exists a function f from Σ^* to $\mathbb{N}$ and a polynomial-time bounded deterministic Turing machine M such that f is computable in time polynomial on μ-average, $D = L(M)$, and $\mathrm{Prob}_\nu[\{y \mid x \in D \text{ iff } \langle x, y\rangle \in L(M)\}] \geqslant 1 - 2^{-q(x)}$ for all x, where ν is the uniform density function on $\Sigma^{f(x)}$.*

Proofs. The proof of Proposition 2.6 is straightforward and follows from the standard technique of encoding non-deterministic computation paths into strings and the fact that Time_M is computed by a deterministic transducer that is polynomial-time bounded on μ-average.

A similar argument shows that (3) infers (2) in Proposition 2.7. Clearly (1) follows from (2). Thus, it only remains to show that (1) infers (3). Assume that (1) holds. Then, Time_M is polynomial on μ-average. Now we perform the usual probability amplification (see, e.g., [2, p. 139]). We simulate the machine M $p(n)(= O(q(n)))$ times and accept x as soon as more that $p(n)/2$ simulations accept x, and reject x as soon as more that $p(n)/2$ simulations reject x. Hence, if we choose $f(x) = p(|x|) \cdot k \cdot \mathrm{Time}_M(x)$, then (3) holds.

3. RANDOMIZED DECISION PROBLEMS

The basic objects of average-case complexity theory are (decision or search) problems, together with distributions on instances, i.e., a density function assigns probabilities to instances of those problems. The time and space complexity of an algorithm for that problem is measured under the assumption that the inputs occur according to the given distribution. The hope is to show that even for (some) intractable problems, hard instances occur only with small probability. Hence, some algorithm should run efficiently on average.

Some NP-complete problems, such as the *satisfiability problem*, the *graph 3-colorability problem*, and the *Hamiltonian circuit problem*, can be solved by deterministic algorithms in time polynomial on average with respect to reasonably chosen density functions [9, 7, 40].

This paper will focus only on *decision problems* (readers interested in *search problems* are referred to [4, 6]). For a decision problem D and a density function μ, the pair (D, μ) is called a *randomized* (*decision*) *problem*. Here, (D, μ) means that instances of the decision problem D are given randomly according to μ; in other words, a string s occurs as input to some algorithm deciding D with probability $\mu(s)$. *Average-case complexity classes* are sets of randomized problems. We note that, in [21, 22], Levin has first considered pairs of a decision problem and a distribution function (also called *distribution problems*). See [4] for more details.

We will consider two different types of average-case complexity classes. The first type is defined by a worst-case complexity class and a class of density functions. In the second type of classes, the resource bounds of the complexity class are taken with respect to the given density functions.

DEFINITION 3.1 [4]. Let $\mathscr{C}$ be a complexity class and $\mathscr{F}$ be a class of density functions. The *randomized class* $\langle \mathscr{C}, \mathscr{F}\rangle$ is the set $\{(D, \mu) \mid \in \mathscr{F} \text{ and } D \in \mathscr{C}\}$.

DEFINITION 3.2. Let t be a function on $\mathbb{N}$ and $\mathscr{F}$ be a class of density functions. Time- and space-bounded *average classes* are defined as follows:

1. $\text{Aver}\langle \text{DTIME}(t), \mathscr{F} \rangle = \{(D, \mu) \mid \mu \in \mathscr{F}$ and $D = L(M)$ for a deterministic Turing machine M which is t-time bounded on μ-average$\}$.

2. $\text{Aver}\langle \text{NTIME}(t), \mathscr{F} \rangle = \{(D, \mu) \mid \mu \in \mathscr{F}$ and $D = L(M)$ for a nondeterministic Turing machine M which is t-time bounded on μ-average$\}$.

3. $\text{Aver}\langle \text{DSPACE}(t), \mathscr{F} \rangle = \{(D, \mu) \mid \mu \in \mathscr{F}$ and $D = L(M)$ for a deterministic Turing machine M which is t-space bounded on μ-average$\}$.

4. $\text{Aver}\langle \text{NSPACE}(t), \mathscr{F} \rangle = \{(D, \mu) \mid \mu \in \mathscr{F}$ and $D = L(M)$ for a nondeterministic Turing machine M which is t-space bounded on μ-average$\}$.

5. $\text{Aver}\langle \text{BTIME}(t), \mathscr{F} \rangle = \{(D, \mu) \mid \mu \in \mathscr{F}$ and $D = L(M)$ for a bounded-error probabilistic Turing machine M which is t-time bounded on μ-average$\}$.

Using the above definitions, one can consider average-case analogues of many known time- or space-bounded complexity classes. For example, NP with polynomial-time computable distributions, as defined in [4], is expressed as $\langle \text{NP}, \text{SP-comp} \rangle$. The set of randomized problems solvable in polynomial-time on average (AverageP or AP in [16, 37]) is denoted by $\text{Aver}\langle \text{P}, * \rangle$. Here, $*$ denotes the set of *all* density functions. $\text{Aver}\langle \text{P}, \text{P-comp} \rangle$ and $\text{Aver}\langle \text{NP}, \text{P-comp} \rangle$ (denoted by AP_{P} and ANP_{P} [37]) is the set of problems (D, μ) such that $\mu \in$ P-comp, and D is solvable respectively deterministically and nondeterministically in polynomial-time on μ-average. The class of randomized problems which are solvable in logarithmic-space on average (Average-logspace [4] and averageL in [14]) is denoted by $\text{Aver}\langle \text{L}, * \rangle$.

Note that Ben-David *et al.* [4] use the notation $\text{AverDTime}(t(n))$ to denote $\text{Aver}\langle \text{DTIME}(t), * \rangle$ (also denoted by $\text{AvDTime}(t(n))$ in [28]). Several important randomized decision problems which belong to $\langle \text{NP}, \text{P-comp} \rangle$ can be found in [16].

The next propositions follow immediately from the definitions of the average-case complexity classes.

PROPOSITION 3.3. *Let* $\mathscr{C} \in \{\text{DTIME}(t), \text{NTIME}(t), \text{DSPACE}(t), \text{NSPACE}(t), \text{BPTIME}(t)\}$ *for some increasing function* t *on* $\mathbb{N}$, *and let* $\mathscr{F}$ *be a set of density functions. Then,* $\langle \mathscr{C}, \mathscr{F} \rangle \subseteq \text{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$.

PROPOSITION 3.4. *Let* $\mathscr{F}$ *be a set of density functions and let* t *be a function on* $\mathbb{N}$:

1. $\text{Aver}\langle \text{DTIME}(t), \mathscr{F} \rangle \subseteq \text{Aver}\langle \text{NTIME}(t), \mathscr{F} \rangle \subseteq \text{Aver}\langle \text{NSPACE}(t), \mathscr{F} \rangle$.

2. $\text{Aver}\langle \text{DTIME}(t), \mathscr{F} \rangle \subseteq \text{Aver}\langle \text{DSPACE}(t), \mathscr{F} \rangle \subseteq \text{Aver}\langle \text{DTIME}(2^{t(n)}), \mathscr{F} \rangle$.

It is natural to ask whether $(D, \nu) \in \text{Aver}\langle \text{DTIME}(t), \mathscr{F} \rangle$ implies $(D, \mu) \in \text{Aver}\langle \text{DTIME}(t \circ h), \mathscr{F} \rangle$ if $\mu \in \mathscr{F}$ and ν h-dominates μ for some function h. An affirmative answer for a special case is given by the following lemma. A set $\mathscr{T}$ of functions is said to be *closed under composition with polynomials* if, for any function t and any polynomial p, $t \in \mathscr{T}$ implies $\lambda x . t(p(x)) \in \mathscr{T}$.

LEMMA 3.5. *Let* μ *and* ν *be density functions,* $\mathscr{T}$ *a set of functions on* $\mathbb{R}^+$ *which is closed under composition with polynomials, and let* h *be a function from* Σ^* *to* $\mathbb{R}^+$. *If* ν *polynomially on* μ*-average dominates* μ *and* h *is* $\mathscr{T}$ *on* ν*-average, then* h *is also* $\mathscr{T}$ *on* μ*-average.*

Proof. Assume that $\mu \preccurlyeq^{\mathrm{p}, \mu} \nu$, and h is t on ν-average for some function $t \in \mathscr{T}$. Choose a function q which is polynomial on μ-average such that, for all $x \in \Sigma^*$, $\nu(x) \cdot q(x) \geqslant \mu(x)$. By assumption, $\text{Prob}_\nu[\{x \mid h(x) > t(|x| \cdot r)\}] < 1/r$ for all $r > 0$. Since the set $\mathscr{T}$ is closed under composition with polynomials, without loss of generality, we assume that $h(\varepsilon) < t(0)$ and therefore $\text{Prob}_\mu[\{x \mid h(\varepsilon) > t(0)\}] = 0$. Since q is polynomial on μ-average, there exists a polynomial p such that $\text{Prob}_\mu[\{x \mid q(x) > p(|x| \cdot r)\}] < 1/r$ for all $r > 0$.

Let μ_n and ν_n denote the conditional probability of strings of length n of μ and ν, respectively. Note that if $q(x) \leqslant p(|x| \cdot r)$ for a string x of length n, then $\mu(\Sigma^n) \cdot \mu_n(x) \leqslant \nu(\Sigma^n) \cdot p(n \cdot r) \cdot \nu_n(x)$ and $\nu(\Sigma^n) \cdot \text{Prob}_{\nu_n}[\{x \mid h(x) > t(|x| \cdot r)\}] < 1/r$ for all $n \in \mathbb{N}$ and $r > 0$. Now define g as $g(x) = t(4x^3 \cdot p(2x))$. Since $\mathscr{T}$ is closed under composition with polynomials, g is in $\mathscr{T}$. We note that, for all $n \in \mathbb{N}$ and all $r \in \mathbb{R}^+$, $g(n \cdot r) \geqslant t(n \cdot r^3 \cdot 4n^2 \cdot p(n \cdot 2r))$. It remains to show that h is g on μ-average. For every real number $r \geqslant 1$,

$$
\begin{aligned}
&\text{Prob}_\mu[\{x \mid h(x) > g(|x| \cdot r)\}] \\
&\quad \leqslant \text{Prob}_\mu[\{x \mid q(x) > p(|x| \cdot 2r)\}] \\
&\qquad + \text{Prob}_\mu[\{x \mid q(x) \leqslant p(|x| \cdot 2r) \wedge h(x) > g(|x| \cdot r)\}] \\
&\quad < \frac{1}{2r} + \sum_{n=1}^{\infty} \nu(\Sigma^n) \cdot p(2nr) \\
&\qquad \times \text{Prob}_{\nu_n}[\{x \mid q(x) \leqslant p(2nr) \wedge h(x) > g(nr)\}] \\
&\quad \leqslant \frac{1}{2r} + \sum_{n=1}^{\infty} \nu(\Sigma^n) \cdot p(2nr) \\
&\qquad \times \text{Prob}_{\nu_n}[\{x \mid h(x) > t(|x| \cdot 4n^2 r^3 \cdot p(2nr))\}] \\
&\quad < \frac{1}{2r} + \sum_{n=1}^{\infty} \frac{p(2nr)}{4n^2 r^3 \cdot p(2nr)} = \frac{1}{2r} + \frac{\pi^2}{24r^2} < \frac{1}{r}. \quad \blacksquare
\end{aligned}
$$

THEOREM 3.6. *Let* $\mathscr{C} \in \{\text{L}, \text{P}, \text{NP}, \text{PSPACE}, \text{NPSPACE}, \text{EXP}\}$, *and assume* $(D, \nu) \in \text{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$ *for some set of density functions* $\mathscr{F}$. *For all* $\mu \in \mathscr{F}$, *if* ν *polynomially on* μ*-average dominates* μ, *then* $(D, \mu) \in \text{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$.

Proof. Since the sets of polynomials, logarithms, and exponentials are all closed under composition with polynomials, the theorem immediately follows from Lemma 3.5. $\blacksquare$

From the definition, it is obvious that if $\mathscr{C}_1$ is a proper subset of $\mathscr{C}_2$, then $\langle \mathscr{C}_1, \mathscr{F} \rangle$ is also a proper subset of $\langle \mathscr{C}_2, \mathscr{F} \rangle$. A similar hierarchy result can be shown for the second type of average-case complexity classes. See also [14, 12].

THEOREM 3.7. *Let s and t be space- and time-constructible functions, respectively, and* $s'(n) \in \omega(\log n)$ $s'(n) \in \omega(s(n \cdot f(n)))$, $t'(n) > (n)$ *and* $t'(n) \in \omega(t(n \cdot f(n)))$ *for some nondecreasing, unbounded function f from* $\mathbb{N}$ *to* $\mathbb{R}^+$. *Assume that* $\mathscr{F}$ *contains a density function* μ_f *of the form*

$$\mu_f(x) = \begin{cases} \dfrac{1}{c \cdot f(|x|)} \cdot 2^{-|x|} & \\ \quad \text{if } f(|x|-1) < k^2 \leqslant f(|x|) \text{ for some integer } k > 2, & \\ 0 \quad \text{otherwise,} & \end{cases}$$

where c is a constant with $\frac{1}{4} > c > 0$:

1. $\mathrm{Aver}\langle \mathrm{DSPACE}(s(n)), \mathscr{F} \rangle \subsetneq \mathrm{Aver}\langle \mathrm{DSPACE}(s'(n)), \mathscr{F} \rangle$.

2. $\mathrm{Aver}\langle \mathrm{DTIME}(t(n)), \mathscr{F} \rangle \subsetneq \mathrm{Aver}\langle \mathrm{DTIME}(\log t(n) \cdot t'(n)), \mathscr{F} \rangle$.

Proof. (1) Choose a nondecreasing, unbounded function f such that $\mu_f \in \mathscr{F}$ and $s'(n) \in \omega(s(n \cdot f(n)))$. It is shown in [39, 26] that there exists a set A in $\mathrm{DSPACE}(s'(n))$ which are *random* for $\mathrm{DSPACE}(s(n \cdot f(n)))$, i.e., for every set $B \in \mathrm{DSPACE}(s(n \cdot f(n)))$ and every $\varepsilon > 0$, there exists an integer $n_0 > 0$, such that, for all $n > n_0$,

$$\left| \frac{|A^n \bigtriangleup B^n|}{2^n} - \frac{1}{2} \right| < \varepsilon.$$

Clearly $(A, \mu_f) \in \langle \mathrm{DSPACE}(s'(n)), \mathscr{F} \rangle \subseteq \mathrm{Aver}\langle \mathrm{DSPACE}(s'(n)), \mathscr{F} \rangle$.

We show that $(A, \mu_f) \notin \mathrm{Aver}\langle \mathrm{DSPACE}(s(n)), \mathscr{F} \rangle$. Assume to the contrary that $(A, \mu) \in \mathrm{Aver}\langle \mathrm{DSPACE}(s(n)), \mathscr{F} \rangle$ and let M be a deterministic Turing machine which accepts A and is s-space bounded on μ_f-average, i.e., $\mathrm{Prob}_{\mu_f}[\{x \mid \mathrm{Space}_M(x) > s(|x| \cdot r)\}] < 1/r$ for any real number $r > 0$. Then, a set $D = \{x \mid \mathrm{Space}_M(x) < s(n \cdot f(n)) \wedge M(x) = 1\}$ belongs to $\mathrm{DSPACE}(s(n \cdot f(n))$, and thus A is random for $\{D\}$. Choose $\varepsilon = \frac{1}{8}$. The randomness of A indicates that there exists an integer $n_0 > 0$ such that, for all $n > n_0$,

$$\left| \frac{|A^n \bigtriangleup D^n|}{2^n} - \frac{1}{2} \right| < \frac{1}{8}.$$

Note that, for all $n, \mu_f(\Sigma^n) = 1/(c \cdot f(n))$ if $f(n-1) < k^2 \leqslant f(n)$ for some constant k. It follows that, for some $n > n_0$,

$$\mathrm{Prob}_{\mu_f}[\{x \in \Sigma^n \mid \mathrm{Space}_M(x) > s(n \cdot f(n))\}] \leqslant \frac{1}{f(n)} \cdot c \cdot f(n) < \frac{1}{4}.$$

Therefore, D is identical to A on at least $\frac{3}{4}$ of all strings of length n. This contradicts the randomness of A.

(2) The proof for the time-bounded classes is similar to (1) and follows from the fact that, for every time-constructible function $t'(n) > (n)$ and $t'(n) \in \omega(t(n \cdot f(n))$, there exists a set in $\mathrm{DTIME}(\log t(n) \cdot t'(n))$ which is random for $\mathrm{DTIME}(t(n \cdot f(n)))$ [39, 26]. ∎

COROLLARY 3.8. *Let p and p′ be polynomials such that* $p'(n) \in \omega(p(n))$:

1. $\mathrm{Aver}\langle \mathrm{DTIME}(p(n)), \text{P-comp} \rangle \subsetneq \mathrm{Aver}\langle \mathrm{DTIME}(\log p(n) \cdot p'(n) + n), \text{P-comp} \rangle$.

2. $\mathrm{Aver}\langle \mathrm{DSPACE}(p(n)), \text{P-comp} \rangle \subsetneq \mathrm{Aver}\langle \mathrm{DSPACE}(p'(n) + \log n), \text{P-comp} \rangle$.

We show a basic relationship between worst-case complexity and average-case complexity on strings with high probability. To show this, we first introduce an "interpolation" property of an average-case complexity class $\mathrm{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$.

DEFINITION 3.9. For a sparse set S and a polynomial q, let $\mu_{S,q}$ denote a density function such that $\mu_{S,q}(x) \geqslant 1/q(|x|)$ for all $x \in S$. A class $\mathrm{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$ has the *sparse interpolation property* if, for any set A, any infinite sparse set S and any polynomial q such that $(A, \mu_{S,q}) \in \mathrm{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$, there exists a set $B \in \mathscr{C}$ such that $A \cap S' \subseteq B \subseteq A$. The set B is called an *interpolant* of A and S.

LEMMA 3.10. *For a class* $\mathscr{C} \in \{\mathrm{P}, \mathrm{NP}, \mathrm{BPP}, \mathrm{PSPACE}\}$, $\mathrm{Aver}\langle \mathscr{C}, * \rangle$ *has the sparse interpolation property.*

Proof. We show the case $\mathscr{C} = \mathrm{NP}$. Take any sparse set S and a polynomial q and assume that $(A, \mu_{S,q}) \in \mathrm{Aver}\langle \mathrm{NP}, * \rangle$. There exists a Turing machine M which computes A such that Time_M is p on $\mu_{S,q}$-average for some polynomial p. Note that $\mathrm{Time}_M(x) \leqslant p(|x|/\mu_{S,q}(x))$ for all x with $\mu_{S,q}(x) > 0$. Let N simulate M on input x in $p(|x| \cdot q(|x|))$ steps. If the simulation of M does not terminate within $p(|x| \cdot q(|x|))$ steps, then N rejects x. Let $B = L(N)$. Clearly $B \subseteq A$. Since $q(|x|) \geqslant 1/\mu_{S,q}(x)$ for all $x \in S$, N completely simulates M on all inputs x in S, and thus, $A \cap S = B \cap S$. Clearly N is polynomial-time bounded. Therefore, $B \in \mathrm{NP}$. ∎

In Section 6, we will extend Lemma 3.10 to the kth level of an "average polynomial-time hierarchy" which is an average-case version of the Meyer–Stockmeyer polynomial-time hierarchy.

One of the most interesting open questions is whether NP sets can be solved in average polynomial time for every polynomial time computable distribution, i.e., if

$\langle \mathrm{NP}, \text{P-comp} \rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, * \rangle$ holds or not. Clearly if $\mathrm{P} = \mathrm{NP}$, then $\langle \mathrm{NP}, \mathscr{F} \rangle$ is included in $\mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle$. Ben-David *et al.* [4] first gave a partial answer to this question by showing that $\langle \mathrm{NP}, \text{P-comp} \rangle \nsubseteq \mathrm{Aver}\langle \mathrm{P}, * \rangle$ if $\mathrm{E} \neq \mathrm{NE}$. Under the same assumption, it also holds that $\langle \mathrm{NP}, \text{L-comp} \rangle \nsubseteq \mathrm{Aver}\langle \mathrm{P}, * \rangle$ [4]. Ben-David *et al.* actually show that $\langle \mathrm{NP} \cap \mathrm{TALLY}, \text{L-comp} \rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, * \rangle$ if and only if $\mathrm{NP} \cap \mathrm{TALLY} \subseteq \mathrm{P}$.

Here we consider the bounded-error probabilistic class $\mathrm{Aver}\langle \mathrm{BPP}, * \rangle$ and extend the above result by Ben-David *et al.* to $\mathrm{Aver}\langle \mathrm{BPP}, * \rangle$.

Theorem 3.11. $\langle \mathrm{NP} \cap \mathrm{TALLY}, \text{L-comp} \rangle \subseteq \mathrm{Aver}\langle \mathrm{BPP}, * \rangle$ *if and only if* $\mathrm{NP} \cap \mathrm{TALLY} \subseteq \mathrm{BPP}$.

Proof. From the assumption that $\mathrm{NP} \cap \mathrm{TALLY} \subseteq \mathrm{BPP}$, it immediately follows that $\langle \mathrm{NP} \cap \mathrm{TALLY}, \text{L-comp} \rangle \subseteq \mathrm{Aver}\langle \mathrm{BPP}, * \rangle$. For the converse, we assume that $\langle \mathrm{NP} \cap \mathrm{TALLY}, \text{L-comp} \rangle \subseteq \mathrm{Aver}\langle \mathrm{BPP}, * \rangle$. Choose a density function μ satisfying $\mu(x) \propto (|x|+1)^{-2}$ if $x \in \{0\}^*$, and 0 otherwise. Obviously $\mu \in \text{L-comp}$. By our assumption, for every set $A \in \mathrm{NP} \cap \mathrm{TALLY}$, $(A, \mu) \in \mathrm{Aver}\langle \mathrm{BPP}, * \rangle$. Namely, there is a probabilistic Turing machine M, with exponential small error probability, computing A which runs in time q on μ-average for some polynomial q. This shows that $\mathrm{Time}_M(x) \leqslant q(|x|/\mu(x)) \leqslant q(3|x|(|x|+1)^2)$ for all $x \in \{0\}^*$. Therefore, A is in BPP. ▌

A standard padding argument shows that $\mathrm{NP} \cap \mathrm{TALLY} \subseteq \mathrm{BPP}$ if and only if $\mathrm{NE} \subseteq \mathrm{BPE}$. Hence, we have the following conclusion.

Corollary 3.12. $\mathrm{NE} \nsubseteq \mathrm{BPE}$ *implies* $\langle \mathrm{NP}, \text{P-comp} \rangle \nsubseteq \mathrm{Aver}\langle \mathrm{BPP}, * \rangle$.

4. MANY–ONE AND TURING REDUCTIONS

A theory of average NP-completeness was initiated by Levin in his terse papers [21, 22]. Levin has introduced a polynomial-time many–one reducibility between randomized decision problems and has shown that the *randomized tiling problem*, the "tiling problem" with a natural distribution, is complete for Random-NP or, in our notation, complete for $\langle \mathrm{NP}, \text{P-comp} \rangle$. Intuitively, this reduction from (D_1, μ_1) to (D_2, μ_2) reduces a set D_1 to a set D_2 and ensures a *domination condition* between μ_1 and μ_2 which guarantees that instances in D_1 occurring with high probability are reduced to instances in D_2 occurring with high probability. The notions of deterministic Turing reducibility and random many–one reducibility have been introduced in [21, 4]. The latter is especially suitable for randomized algorithms (see [4, 18, 6]). Also considered so far were logspace many–one reductions [4] and logspace many–one reductions which are p-honest [14].

We first recall from [21, 4] the definition of polynomial time many–one and Turing reducibilities.

Definition 4.1 [21]. Let (D_1, μ_1) and (D_2, μ_2) be randomized decision problems:

1. (D_1, μ_1) is *polynomial-time many–one reducible* to (D_2, μ_2), denoted by $(D_1, \mu_1) \leqslant^{\mathrm{p}}_{m} (D_2, \mu_2)$, if there exists a density function ν and a function f such that

(i) $f \in \mathrm{FP}$;

(ii) for all x, $x \in D_1$ if and only if $f(x) \in D_2$; and

(iii) $\mu_1 \preccurlyeq^{\mathrm{p}} \nu$, and $\mu_2(y) \geqslant \mathrm{Prob}_\nu[\{x \mid f(x) = y\}]$ for all y.

2. (D_1, μ_1) is *average polynomial-time many–one reducible* to (D_2, μ_2), denoted by $(D_1, \mu_1) \leqslant^{\mathrm{p,\,av}}_{m} (D_2, \mu_2)$, if f is polynomial on μ_1-average and $\mu_1 \preccurlyeq^{\mathrm{p}, \mu_1} \nu$ in 1.

The condition (iii) on the density functions in the above definition is simply called the *domination condition* for the reduction. There are several weaker definitions of the domination conditions; however, the one used here simplifies the reductions and guarantees that the reducibilities are reflexive and transitive. See [16] for more discussion.

Definition 4.2 [4]. Let (D_1, μ_1) and (D_2, μ_2) be randomized decision problems:

1. (D_1, μ_1) is *polynomial-time Turing reducible* to (D_2, μ_2), denoted by $(D_1, \mu_1) \leqslant^{\mathrm{p}}_{T} (D_2, \mu_2)$, if there exists a density function ν and an oracle Turing machine M such that

(i) M^{D_2} is polynomial-time bounded;

(ii) $D_1 = L(M, D_2)$; and

(iii) $\mu_1 \preccurlyeq^{\mathrm{p}} \nu$, and $\mu_2(y) \geqslant \mathrm{Prob}_\nu[\{x \mid y \in Q(M, D_2, x)\}]$ for all y.

Here $Q(M, D_2, x)$ is the set of strings queried by M with oracle D_2 on input x.

2. (D_1, μ_1) is *average polynomial-time Turing reducible* to (D_2, μ_2), denoted by $(D_1, \mu_1) \leqslant^{\mathrm{p,\,av}}_{T} (D_2, \mu_2)$, if M is polynomial-time bounded on μ-average and $\mu_1 \preccurlyeq^{\mathrm{p}, \mu_1} \nu$ in 1.

Let α be a reducibility. For a class $\mathscr{C}$, a problem (D, μ) is α-*hard* for $\mathscr{C}$ if every problem (E, ν) in $\mathscr{C}$ is α-reducible to (D, μ), and (D, μ) is α-*complete* for $\mathscr{C}$ if it is in $\mathscr{C}$ and is α-hard for $\mathscr{C}$. If $\mathscr{C}$ is of the form $\langle \mathscr{C}', \mathscr{F} \rangle$ (resp. $\mathrm{Aver}\langle \mathscr{C}', \mathscr{F} \rangle$), then let $\mathscr{C}$-complete abbreviate "many-one complete" (resp. "many-one complete on average") for $\mathscr{C}$.

It is shown in [16] that if (D, μ) is $\leqslant^{\mathrm{p}}_{m}$-hard for $\langle \mathrm{NP}, \text{P-comp} \rangle$ for a language D in EXP and a flat density function μ, then $\mathrm{EXP} = \mathrm{NEXP}$. Note that that the standard density function is flat. Hence, under the condition of $\mathrm{EXP} \neq \mathrm{NEXP}$, no decision problem with the standard density function is $\langle \mathrm{NP}, \text{P-comp} \rangle$-complete. By the result of Wang and Belanger [37], every $\langle \mathrm{NP}, \text{P-comp} \rangle$-complete problem is also $\mathrm{Aver}\langle \mathrm{NP}, \text{P-comp} \rangle$-complete, i.e., $\leqslant^{\mathrm{p,av}}_{m}$-complete for $\mathrm{Aver}\langle \mathrm{NP}, \text{P-comp} \rangle$.

Now we introduce an average-case version of the nondeterministic Turing reducibility. This will be used to build an "average polynomial-time hierarchy" in Section 6. Recall that $\mathrm{Acc}(M, D, x)$ (resp. $\mathrm{Rej}(M, D, x)$) is the set of (codes of) all accepting (resp. rejecting) computation paths of M with oracle D on input x, and $Q(M, D, x, y)$ is the set of strings queried by M with oracle D on input x on computation path y.

Definition 4.3. Let (D_1, μ_1) and (D_2, μ_2) be randomized decision problems:

1. (D_1, μ_1) is *polynomial-time nondeterministic Turing reducible* to (D_2, μ_2), denoted by $(D_1, \mu_1) \leqslant_T^{\mathrm{np}} (D_2, \mu_2)$, if there exist a density function ν and a nondeterministic Turing machine M such that

   i. M^{D_2} is polynomial-time bounded;

   ii. $D_1 = L(M, D_2)$; and

   iii. $\mu_1' \preceq^{\mathrm{p}} \nu$, and $\mu_2(z) \geqslant \mathrm{Prob}_\nu[\{(x, y) \mid z \in Q(M, D_2, x, y)\}]$ for all z,

where μ_1' is the density function induced from μ, M, and D_2 as:

$$\mu_1'(x, y) = \begin{cases} \mu_1(x)/|\mathrm{Acc}(M, D_2, x)| & \\ \qquad \text{if } y \in \mathrm{Acc}(M, D_2, x), & \\ \mu_1(x)/|\mathrm{Rej}(M, D_2, x)| & \\ \qquad \text{if } \mathrm{Acc}(M, D_2, x) = \varnothing & \\ \qquad \text{and } y \in \mathrm{Rej}(M, D_2, x), & \\ 0 \qquad \text{otherwise.} & \end{cases}$$

2. (D_1, μ_1) is *average polynomial-time nondeterministic Turing reducible* to (D_2, μ_2), denoted by $(D_1, \mu_1) \leqslant_T^{\mathrm{np,\,av}} (D_2, \mu_2)$, if M^{D_2} is polynomial-time bounded on μ_1-average and $\mu_1' \preceq^{\mathrm{p}, \mu_1'} \nu$ in 1.

We note that, in the case that the reduction machine always has one computation path, the nondeterministic Turing reduction coincides with deterministic one in Definition 4.2.

In the following, we state basic properties of the reducibilities (cf. [16, 4]).

Proposition 4.4. *Let $A_i = (D_i, \mu_i)$, $i = 1, 2, 3$, be randomized decision problems*:

1. *Polynomial-time reducibility implies average polynomial-time reducibility, i.e., for every $\alpha \in \{m, T\}$ and $\beta \in \{\mathrm{p}, \mathrm{np}\}$, if $A_1 \leqslant_\alpha^\beta A_2$ then $A_1 \leqslant_\alpha^{\beta,\,\mathrm{av}} A_2$.*

2. *Many–one reducibility implies deterministic Turing reducibility, i.e., if $A_1 \leqslant_m^{\mathrm{p}} A_2$ then $A_1 \leqslant_T^{\mathrm{p}} A_2$, and if $A_1 \leqslant_m^{\mathrm{p,\,av}} A_2$ then $A_1 \leqslant_T^{\mathrm{p,\,av}} A_2$.*

3. *Deterministic Turing reducibility implies nondeterministic Turing reducibility, i.e., if $A_1 \leqslant_T^{\mathrm{p}} A_2$ then $A_1 \leqslant_T^{\mathrm{np}} A_2$, and if $A_1 \leqslant_T^{\mathrm{p,\,av}} A_2$ then $A_1 \leqslant_T^{\mathrm{np,\,av}} A_2$.*

4. *Nondeterministic Turing reducibility is reflexive, i.e., $A_1 \leqslant_T^{\mathrm{np}} A_1$ and $A_1 \leqslant_T^{\mathrm{np,\,av}} A_1$.*

5. *Many-one reducibility and deterministic Turing reducibility are reflexive and transitive, i.e., for every $\leqslant_\alpha \in \{\leqslant_m^{\mathrm{p}}, \leqslant_m^{\mathrm{p,\,av}}, \leqslant_T^{\mathrm{p}}, \leqslant_T^{\mathrm{p,\,av}}\}$, $A_1 \leqslant_\alpha A_1$, and if $A_1 \leqslant_\alpha A_2$ and $A_2 \leqslant_\alpha A_3$ then $A_1 \leqslant_\alpha A_3$.*

Proof. The claims (1)–(4) immediately follow from the definitions. The claim for reflexivity in (5) is also obvious. Here we show that $\leqslant_T^{\mathrm{p,\,av}}$ is transitive. The proofs for the transitivity of the other reducibilities are analogous. Now, we assume that $(D_1, \mu_1) \leqslant_T^{\mathrm{p,\,av}} (D_2, \mu_2)$ via a deterministic Turing machine M_1 and a density function ν_2 and assume that $(D_2, \mu_2) \leqslant_T^{\mathrm{p,\,av}} (D_3, \mu_3)$ via a deterministic machine M_2 and a density function ν_2. In what follows, we will show that $(D_1, \mu_1) \leqslant_T^{\mathrm{p,\,av}} (D_3, \mu_3)$.

By definition, there exist two functions f_1 and f_2 which are polynomial on μ_1-average and on μ_2-average, respectively, such that $f_1(x) \cdot \upsilon_1(x) \geqslant \mu_1(x)$ and $f_2(x) \cdot \nu_2(x) \geqslant \mu_2(x)$ for all x. Assume that $f_1(x) > 0$ and $f_2(x) > 0$ for all strings x.

We define a new machine M as follows: on input x, M deterministically simulates M_1 on x, and whenever M_1 queries a string y, M deterministically simulates M_2 on input y. Especially, in the case that x is the empty string ε, M is designed not to query any strings; even if M_2 queries some strings to oracle D_3, but their oracle answers are encoded in a program of M. Clearly $D_1 = L(M, D_3)$. Note that

$$\mathrm{Time}_M^{D_3}(x) \leqslant c \cdot \left(\mathrm{Time}_{M_1}^{D_2}(x) + \sum_{y \in Q(M_1, D_2, x)} \mathrm{Time}_{M_2}^{D_3}(y) \right)$$

for some constant $c > 0$.

Let $f(x) = f_1(x) \cdot (\sum_{y \in Q(M_1, D_2, x)} f_2(y) + 1)$ and choose a density function ν such that $\nu(\varepsilon) = 1 - \sum_x \mu_1(x)/f(x)$ and $f(x) \cdot \nu(x) = \mu_1(x)$ for all strings x different from ε. For each string z, we define a set A_z such that that $(x, y) \in A_z$ if and only if $z \in Q(M_2, D_3, y)$, $y \in Q(M_1, D_2, x)$, and $(x', y) \notin A_z$ for every string $x' < x$. Then, for all $z \in \bigcup_x Q(M, D_3, x)$,

$$\begin{aligned} \mu_3(z) &\geqslant \mathrm{Prob}_{\nu_2}[\{y \mid z \in Q(M_2, D_3, y)\}] \\ &\geqslant \sum_{y : z \in Q(M_2, D_3, y)} \frac{\mu_2(y)}{f_2(y)} \\ &\geqslant \sum_{x : z \in Q(M, D_3, x)} \sum_{y : (x, y) \in A_z} \frac{1}{f_2(y)} \cdot \mu_2(y) \\ &\geqslant \sum_{x : z \in Q(M, D_3, x)} \sum_{y : (x, y) \in A_z} \frac{1}{f_2(y)} \cdot \frac{\mu_1(x)}{f_1(x)} \\ &\geqslant \sum_{x : z \in Q(M, D_3, x)} \frac{\mu_1(x)}{f_1(x) \cdot \sum_{y \in Q(M_1, D_2, x)} f_2(y)} \\ &\geqslant \mathrm{Prob}_\nu[\{x \mid z \in Q(M, D_3, x)\}] \end{aligned}$$

since $\sum_{i=1}^n (1/a_i) \geqslant 1/\sum_{i=1}^n a_i$.

It remains to show that M and f are polynomial on μ_1-average. Let p_1 and p_2 be polynomials such that, for any real number $r > 0$,

$$\mathrm{Prob}_{\mu_1}[\{x \mid \mathrm{Time}_{M_1}^{D_2}(x) < p_1(|x| \cdot r)\}] < 1/r,$$

$$\mathrm{Prob}_{\mu_2}[\{y \mid \mathrm{Time}_{M_2}^{D_3}(y) < p_2(|y| \cdot r)\}] < 1/r.$$

Now we define a polynomial s as $s(z) = c \cdot p_1(2z) \cdot (1 + p_2(2z \cdot p_1(2z))) + c_0$, where $c_0 = \mathrm{Time}_M^{D_3}(\varepsilon)$. Note that $|Q(M_1, D_2, x)| \leqslant \mathrm{Time}_{M_1}^{D_2}(x)$ for all x. Then, for all x and $r > 0$,

$$\begin{aligned}
&\mathrm{Prob}_{\mu_1}[\{x \mid \mathrm{Time}_M^{D_3}(x) > s(|x| \cdot r)\}] \\
&\quad \leqslant \mathrm{Prob}_{\mu_1}[\{x \mid \mathrm{Time}_{M_1}^{D_2}(x) > p_1(|x| \cdot 2r)\}] \\
&\qquad + \mathrm{Prob}_{\mu_1}\Bigg[\Bigg\{x \,\Bigg|\, \mathrm{Time}_{M_1}^{D_2}(x) \leqslant p_1(|x| \cdot 2r) \\
&\qquad \wedge \sum_{y \in Q(M_1, D_2, x)} \mathrm{Time}_{L_2}^{D_3}(y) \\
&\qquad > p_2(p_1(|x| \cdot 2r) \cdot 2r) \cdot p_1(|x| \cdot 2r)\Bigg\}\Bigg] \\
&\quad < 1/2r + \mathrm{Prob}_{\mu_1}[\{x \mid \exists y \in Q(M_1, D_2, x) \\
&\qquad\quad [\mathrm{Time}_{M_2}^{D_3}(y) > p_2(|y| \cdot 2r)]\}] \\
&\quad \leqslant 1/2r + \mathrm{Prob}_{\mu_2}[\{y \mid \mathrm{Time}_{M_2}^{D_3}(y) > p_2(|y| \cdot 2r)\}] < 1/r.
\end{aligned}$$

The proof that f is polynomial on μ_1-average is similar and, thus, the claim is established. ∎

The following lemma shows how the domination condition for Turing reducibilities works.

LEMMA 4.5. *Assume that (E) is computable by a nondeterministic Turing machine N in time polynomial on ν-average*:

1. *Assume that $(D, \mu) \leqslant_T^{\mathrm{p,\,av}} (E, \nu)$ via a machine M and let $h(x) = \sum_{z \in Q(M, D, x)} \mathrm{Time}_N(z)$. Then, h is polynomial on μ-average.*

2. *Assume that $(D, \mu) \leqslant_T^{\mathrm{np,\,av}} (E, \nu)$ via a machine M and let $h(x) = \min_{y \in \mathrm{Acc}(M, D, x)} \sum_{z \in Q(M, D, x, y)} \mathrm{Time}_N(z)$ if $x \in D$; otherwise, $h(x) = \min_{y \in \mathrm{Rej}(M, D, x)} \sum_{z \in Q(M, D, x, y)} \mathrm{Time}_N(z)$. Then, h is polynomial on μ-average.*

Proof. We prove the claim (2). Since $(D, \mu) \leqslant_T^{\mathrm{np,\,av}} (E, \nu)$ via some nondeterministic oracle Turing machine M, there exist a density function υ and a polynomial p_D such that $D = L(M, E)$, $\mu' \preccurlyeq^{\mathrm{p},\,\mu'} \upsilon$, Time_M^E is p_D on μ-average, and $\nu(z) \geqslant \mathrm{Prob}_\upsilon[\{(x, y) \mid z \in Q(M, E, x, y)]$ for all z, where μ' is the density function induced from μ, M, and E as in Definition 4.3. Assume that $\mathrm{Time}_M^E(x) > |x|$ for all x.

Choose a nondeterministic Turing machine N and a polynomial p_E such that $E = L(N)$ and Time_N is p_E on ν-average. Moreover, let p be a polynomial and q a function such that q is p on μ-average and $q(x) \cdot \upsilon(x) \geqslant \mu'(x)$ for all x.

Now define a polynomial s as

$$\begin{aligned}
s(z) = p_D(6z) \cdot p_E(p_D(6z) \cdot 10z^2 \cdot (p_D(6z) + 2)^2 \\
\times p(6z \cdot p_D(6z))) + c_0,
\end{aligned}$$

where $c_0 = \mathrm{Time}_M^E(\varepsilon)$.

We show that h is s on μ-average. For simplicity, let A_x denote $\mathrm{Acc}(M, E, x)$ and let $Q_{x,\,y}$ denote $Q(M, E, x, y)$. Note that $|Q_{x,\,y}|$, $|z| \leqslant \mathrm{Time}_M^E(x)$ and $|\langle x, y\rangle| \leqslant 2\,|x|\,(\mathrm{Time}_M^E(x) + 2)$ if $x \neq \varepsilon$. First we prove that $\mathrm{Prob}_\mu[\{x \in D \mid h(x) > s(r \cdot |x|)\}] < 1/2r$. For any real number $r > 0$,

$$\begin{aligned}
&\mathrm{Prob}_\mu[\{x \in D \mid h(x) > s(r \cdot |x|)\}] \\
&\quad \leqslant \mathrm{Prob}_\mu[\{x \in D \mid \mathrm{Time}_M^E(x) > p_D(6r \cdot |x|)\}] \\
&\qquad + \mathrm{Prob}_\mu[\{x \in D \mid \mathrm{Time}_M^E(x) \leqslant p_D(6r \cdot |x|) \\
&\qquad \wedge \forall y \in A_x \Bigg[\sum_{z \in Q_{x,\,y}} \mathrm{Time}_N(z) > s(r \cdot |x|)]\}\Bigg] \\
&\quad \leqslant \frac{1}{6r} + \mathrm{Prob}_\mu[\{x \in D \mid \forall y \in A_x \exists z \in Q_{x,\,y} \\
&\qquad\quad [\mathrm{Time}_N(z) > p_E(|z| \cdot 10r \cdot |\langle x, y\rangle|^2 \\
&\qquad \times p(6r \cdot |\langle x, y\rangle|))]\}]
\end{aligned}$$

The latter term can be calculated further as

$$\begin{aligned}
&\mathrm{Prob}_{\mu'}[\{(x, y) \mid x \in D \wedge \exists z \in Q_{x,\,y} \\
&\qquad\quad [\mathrm{Time}_N(z) > p_E(|z| \cdot 10r \cdot |\langle x, y\rangle|^2 \\
&\qquad \times p(6r \cdot |\langle x, y\rangle|))]\}] \\
&\quad = \sum_{n=1}^{\infty} \mu'(\Sigma^n) \cdot \mathrm{Prob}_{\mu'_n}[\{(x, y) \mid x \in D \wedge \exists z \in Q_{x,\,y} \\
&\qquad\quad [\mathrm{Time}_N(z) > p_E(|z| \cdot 10rn^2 \cdot p(6rn))]\}] \\
&\quad \leqslant \mathrm{Prob}_{\mu'}[\{(x, y) \mid q(\langle x, y\rangle) > p(6r \cdot |\langle x, y\rangle|)\}] \\
&\qquad + \sum_{n=1}^{\infty} \upsilon(\Sigma^n) \cdot p(6rn) \cdot \mathrm{Prob}_{\upsilon_n}[\{(x, y) \mid \exists z \in Q_{x,\,y} \\
&\qquad\quad [\mathrm{Time}_N(z) > p_E(|z| \cdot 10rn^2 \cdot p(6rn))]\}] \\
&\quad \leqslant \frac{1}{6r} + \sum_{n=1}^{\infty} p(6rn) \cdot \mathrm{Prob}_\nu[\{z \mid \mathrm{Time}_N(z) \\
&\qquad > p_E(|z| \cdot 10rn^2 \cdot p(6rn))\}] \\
&\quad \leqslant \frac{1}{6r} + \sum_{n=1}^{\infty} p(6rn) \cdot \frac{1}{10rn^2 \cdot p(6rn)} = \frac{1}{6r} + \frac{\pi^2}{60r} < \frac{1}{3r}.
\end{aligned}$$

A similar argument shows that $\mathrm{Prob}_\mu[\{x \in \bar{D} \mid h(x) > s(r \cdot |x|)\}] < 1/2r$. Thus, h is s on μ-average. ∎

Lemma 4.6. *Let $\mathscr{F}$ be a set of density functions*:

1. Aver$\langle$NP, $*\rangle$ *is closed under* $\leqslant_m^{\mathrm{p,\,av}}$*-reductions* [16].

2. Aver$\langle$P, $*\rangle$ *is closed under* $\leqslant_T^{\mathrm{p,\,av}}$*-reductions* [4]. *Moreover*, Aver$\langle$P, $\mathscr{F}\rangle = \{(D,\mu) \mid \mu\in\mathscr{F},\ (D,\mu)\leqslant_T^{\mathrm{p,\,av}} (E,\nu)$ *and* $(E,\nu)\in$ Aver$\langle$P, $*\rangle\}$.

3. Aver$\langle$NP, $\mathscr{F}\rangle = \{(D,\mu) \mid \mu\in\mathscr{F},\ (D,\mu)\leqslant_T^{\mathrm{np,\,av}} (E,\nu)$ *and* $(E,\nu)\in\langle$P, $*\rangle\}$.

Proof. The claims (1) and (2) follow from Lemma 4.5. Here we show the claim (3). Clearly if $(D,\mu)\in$ Aver$\langle$NP, $\mathscr{F}\rangle$, then $(D,\mu)\leqslant_T^{\mathrm{np,\,av}}(\phi,\nu_{\mathrm{st}})$. To see the other direction, we assume that $\mu\in\mathscr{F}$, $(D,\mu)\leqslant_T^{\mathrm{np,\,av}}(E,\nu)$ and $(E,\nu)\in\langle$P, $*\rangle$. We will show that (D,μ) belongs to Aver$\langle$NP, $\mathscr{F}\rangle$.

Since $(D,\mu)\leqslant_T^{\mathrm{np,\,av}}(E,\nu)$, there exist a polynomial p_D and a nondeterministic oracle Turing machine M_D such that $D=L(M_D,E)$ and $\mathrm{Time}^E_{M_D}$ is p_D on μ-average. Let M_E be a deterministic Turing machine and p_E a polynomial such that $E=L(M_E)$ and Time_{M_E} is p_E-time bounded.

Note that $\mathrm{Time}_{M_E}(z)\leqslant p_E(\mathrm{Time}_{M_D}(x))$ for all y and all z in $Q(M_D,E,x,y)$. Now we consider a machine M which nondeterministically simulates the computation of M_D, and whenever M_D makes a query z, M deterministically simulates M_E on input z. By definition, D is computed by M, and on each computation path y of M on x, the number of steps that M actually takes is bounded by

$$\begin{aligned}
&c\cdot\left(\mathrm{Time}^E_{M_D}(x)+\sum_{z\in Q(M_D,\,E,\,x,\,y)}\mathrm{Time}_{M_E}(z)\right)\\
&\quad\leqslant c\cdot(\mathrm{Time}^E_{M_D}(x)+\mathrm{Time}^E_{M_D}(x)\\
&\qquad\times\max_{z\in q(M_D,\,E,\,x,\,y)}\mathrm{Time}_{M_E}(z))\\
&\quad\leqslant c\cdot\mathrm{Time}^E_{M_D}(x)\cdot(1+p_E(\mathrm{Time}^E_{M_D}(x))).
\end{aligned}$$

Hence, we can redefine M such that all computation paths of M have the same length. The function Time_M is polynomial on μ-average since $\mathrm{Time}^E_{M_D}$ is polynomial on μ-average, and the set of functions which are polynomial on μ-average is closed under composition with polynomials [16]. Therefore, $(D,\mu)\in$ Aver$\langle$NP, $\mathscr{F}\rangle$. ▮

Although Aver$\langle$P, $*\rangle$ is closed under polynomial-time many–one reducibility, it is known that there exist two problems (A,μ) and (B,ν) such that $(A,\mu)\leqslant_m^{\mathrm{p}}(B,\nu)$, and $(B,\nu)\in$ Aver$\langle$P, P-comp$\rangle$, but $(A,\mu)\notin$ Aver$\langle$P, P-comp$\rangle$ [37].

The following theorem shows the existence of incomparable pairs with respect to $\leqslant_T^{\mathrm{p}}$.

Theorem 4.7. *For every recursive decision problem D not in* P, *there exist density functions μ_1 and μ_2 in* P-comp *such that (D,μ_1) and (D,μ_2) are incomparable, i.e.*, $(D,\mu_1)\not\leqslant_T^{\mathrm{p}}(D,\mu_2)$ *and* $(D,\mu_2)\not\leqslant_T^{\mathrm{p}}(D,\mu_1)$.

Proof. The proof proceeds by a slow diagonalization technique. Let $M_1, M_2, \ldots$ denote a standard enumeration of all deterministic polynomial-time oracle Turing machines. We identify each nonnegative integer i with the $(i+1)$th string on the lexicographic order: $\varepsilon<0<1<00<01<\cdots$. We define two density functions μ_1,μ_2 and an auxiliary function r on $\mathbb{N}$ by the following recursive procedure.

Stage 0. Let $r(0)=0$, $\mu_1(0)\propto 1$, and $\mu_2(0)\propto 1$.

Stage n, $n>0$. The values $r(n),\mu_1(n)$, and $\mu_2(n)$ are defined as follows. Assume that, in $|n|$ steps, the initial segment of the sequence

$$\langle r(0),D(0),\mu_1(0),\mu_2(0)\rangle,\ \langle r(1),D(1),\mu_1(1),\mu_2(1)\rangle,\ \ldots$$

is computed by applying repeatedly the same procedure for previous stages. Let m be the largest integer, if any, for which $\langle r(m),D(m),\mu_1(m),\mu_2(m)\rangle$ is completely computed; if no such m exists, however, then let $r(n)=1$, $\mu_1(n)\propto 1$, and $\mu_2(n)\propto 1$. The values depend on whether $r(m)$ is even or odd.

First suppose that $r(m)$ is even. Let $i=r(m)/2$, and assume that the sequence $M_i^D(0), M_i^D(1), \ldots$ is computed by simulating M_i until either more than $|n|$ steps are done, or on some input y, $M_i^D(y)$ queries to oracle D a string larger than m. Let k be the largest integer for which the simulation of $M_i^D(k)$ can be completed. If there exists a $y\leqslant k$ such that either (i) $M_i^D(y)\neq D(y)$, or (ii) M_i^D on input y queries some w satisfying that $\mu_1(y)>\mu_2(w)=0$, then let $r(n)=r(m)+1$, $\mu_1(n)=0$, and $\mu_2(n)\propto 2^{-(|n|+1)}$. Clearly (D,μ_1) is not Turing reducible to (D,μ_2) via M_i. If there is no such y, then let $r(n)=r(m)$, $\mu_1(n)\propto 2^{-(|n|+1)}$, and $\mu_2(n)=0$.

If $r(m)$ is odd, then let $i=(r(m)-1)/2$ and change the roles of μ_1 and μ_2.

Claim 1. range$(r)=\mathbb{N}$, *where* range$(r)=\{r(z)\mid z\in\mathbb{N}\}$.

Proof of Claim. Assume range$(r)\neq\mathbb{N}$. Take the minimal integer n_0 such that $r(n_0)$ is the maximum in range(r).

First consider the case that $r(n_0)$ is even. Let $i=r(n_0)/2$ and let n be large enough such that $\langle r(n_0), D(n_0), \mu_1(n_0), \mu_2(n_0)\rangle$ is constructed. Note that, for every $y>n$, $\mu_1(y)>0$ and $\mu_2(y)=0$. For every $y>n$, we have $M_i^D(y)=D(y)$, and $M_i^D(y)$ does not query any string w, where $\mu_1(y)>\mu_2(w)=0$. Hence, M_i^D computes D on all inputs, and it queries only strings shorter than n_0. This implies that D is in P, and this contradicts our assumption. The same argument also holds for the case that $r(n_0)$ is odd. ▮

Therefore, μ_1 and μ_2 are well defined. It is not hard to see that $\mu_1,\mu_2\in$ P-comp since, in each stage n, we quit the simulations after $|n|$ steps are done. Thus, we complete the proof. ▮

5. COMPLETE PROBLEMS AND SELF-REDUCIBILITY

The *randomized tiling problem* is the first problem that was shown to be $\langle$NP, P-comp$\rangle$-complete [21, 22]. In the past decade, several other randomized decision problems have been proven to be $\langle$NP, P-comp$\rangle$-complete [13, 15, 16, 29, 35, 36, 38]. One of the most useful randomized problem is the *randomized bounded halting problem* (RH, μ_{RH}) that is defined as follows: $\mathrm{RH} = \{\langle i, x, 1^n\rangle \mid M_i$ accepts x within n steps$\}$ and $\mu_{\mathrm{RH}}(\langle i, x, 1^n\rangle) \propto (|i|+1)^{-2}$ $(|x|+1)^{-2}\,(n+1)^{-2}\,2^{-(|i|+|x|)}$, where $M_0, M_1, \ldots$ is a fixed enumeration of all nondeterministic Turing machines. A proof that (RH, μ_{RH}) is $\langle$NP, P-comp$\rangle$-complete can be found in [13, 16, 4]. We note that each of the complete problems is a pair of an NP-complete set and a *natural* density function. However, a randomized satisfiability problem and a randomized graph 3-colorability problem are not $\langle$NP, P-comp$\rangle$-complete for reasonable natural density functions [9, 40]. In [37], Wang and Belanger show that for every set D, if D is $\leqslant^{\mathrm{p}}_m$-hard for NP, then there exists a density function μ such that (D, μ) is $\leqslant^{\mathrm{p}}_m$-hard for $\langle$NP, P-comp$\rangle$. However, it is not known whether every NP-complete set D has a density function μ such that (D, μ) is $\langle$NP, P-comp$\rangle$-complete.

The (*Turing*) *self-reducibility* has been introduced into worst-case complexity theory by Meyer and Paterson [24]. All known NP-complete problems are self-reducible and every self-reducible set belongs to PSPACE. It is natural to ask whether the notion of self-reducibility is applicable to randomized decision problems.

Definition 5.1. A polynomial-time computable partial order $<$ is *OK* if there exists a polynomial p such that

1. every strictly descending chain is finite and is polynomial in the length of its maximum element, i.e., if $x_k < x_{k-1} < \cdots < x_2 < x_1$ is a descending chain starting from x_1, then $k \leqslant p(|x_1|)$, and
2. for every x and y, $x < y$ implies $|x| \leqslant p(|y|)$.

Definition 5.2. A randomized decision problem (D, μ) is (*Turing*) *self-reducible* if there exists an *OK* partial order and a deterministic oracle Turing machine M such that $(D, \mu) \leqslant^{\mathrm{p}}_T (D, \mu)$ via M, and for every input x, all query strings in the computation of M on input x are smaller than x with respect to the partial order.

Clearly every randomized problem in Aver$\langle$P, $*\rangle$ is self-reducible and every self-reducible randomized problem is in Aver$\langle$PSPACE, $*\rangle$. Moreover, the set of all self-reducible problems is closed under polynomial isomorphism, i.e., if $(D, \mu) \cong^{\mathrm{p}} (E, \nu)$ and (E, ν) is self-reducible, then (D, μ) is self-reducible.

One of the classical self-reducible NP-complete problems is the *satisfiability problem*, SAT. However, we do not know a simple density function μ such that (SAT, μ) is $\langle$NP, P-comp$\rangle$-complete. Franco and Paull [9] show that SAT with a natural probability distribution on formulas is in Aver$\langle$P, $*\rangle$. So, we consider, as a canonical example, the randomized bounded halting problem again. More precisely, we consider the kth level of the randomized bounded halting problem $(\mathrm{RH}^k, \mu_{\mathrm{RH}})$ that is defined as follows. Assume that $M_0, M_1, \ldots$ is an effective enumeration of all nondeterministic oracle Turing machines, and define $\mathrm{RH}(A) = \{\langle i, x, 1^n\rangle \mid M_i^A$ accepts x in less than n steps$\}$. Let $\mathrm{RH}^1 = \mathrm{RH}(\phi)$ and $\mathrm{RH}^{k+1} = \mathrm{RH}(\mathrm{RH}^k)$ for $k \geqslant 1$. We note that RH^k is Σ^{p}_k-complete.

Lemma 5.3. *For any* $k > 0$, $(\mathrm{RH}^k, \mu_{\mathrm{RH}})$ *is* $\langle\Sigma^{\mathrm{p}}_k$, P-comp$\rangle$-*complete.*

Proof. The case $k = 1$ is shown in [4, 13, 16, 38]. Now let $k > 1$. The proof follows [37]. For any set $D \in \Sigma^{\mathrm{p}}_k$ and any density function $\mu \in$ P-comp, we will show that $(D, \mu) \leqslant^{\mathrm{p}}_m (\mathrm{RH}^k, \mu_{\mathrm{RH}})$. Without loss of generality, we assume that $|\mu(x)| \leqslant 2\,|x| + 4$ (see Lemma 1.6 in [16]). Since RH^{k-1} is $\Sigma^{\mathrm{p}}_{k-1}$-complete, there exists a polynomial-time nondeterministic oracle Turing machine M such that $D = L(M, \mathrm{RH}^{k-1})$. Let a function g be defined as follows: on input x, g deterministically computes a minimal string y such that $\mu^*(x^-) < 0.y1 \leqslant \mu^*(x)$. Now consider a machine N: on input y, N first computes a string x that $y = g(x)$, by a binary search in time polynomial in $|x|$, and if x exists, then N nondeterministically simulates M on input x; otherwise, N simply rejects x. Note that $|g(x)| \leqslant q(|x|)$ for some absolute polynomial q, and that $\mu(x) < 2^{-|g(x)|}$. Now let i be an index such that $L(M_i) = L(N)$. Let p be a polynomial time bound of M_i. The desired reduction f is now defined as $f(x) = \langle i, g(x), 1^{p(|x|)}\rangle$. Note that f is one-one and reduces D to RH^k. It suffices to check that f satisfies the domination condition. Since i is a constant in the reduction, it follows that

$$\begin{aligned}
&\mu_{\mathrm{RH}}(\langle i, g(x), 1^{p(|x|)}\rangle)\\
&\quad = \frac{c}{(|i|+1)^2\,(|g(x)|+1)^2\,(p(|x|)+1)^2 \cdot 2^{|i|}}\,2^{-|g(x)|}\\
&\quad \geqslant \frac{1}{s(|x|)} \cdot \mu(x),
\end{aligned}$$

where s is a polynomial such that $c \cdot s(n) \geqslant (|i|+1)^2\,(q(n)+1)^2\,(p(n)+1)^2 \cdot 2^{|i|}$. Hence, $(D, \mu) \leqslant^{\mathrm{p}}_m (\mathrm{RH}^k, \mu_{\mathrm{RH}})$. ▮

Theorem 5.4. *For each* $k > 0$, $(\mathrm{RH}^k, \mu_{\mathrm{RH}})$ *is self-reducible.*

Proof. Without loss of generality, we assume that nondeterministic Turing machines have transition functions with exactly two nondeterministic choices. Consider the following encoding of nondeterministic oracle Turing machines M: let $\langle M\rangle$ be the code of a set of finite states (each state q_i is simply encoded by $\langle 0, i\rangle$), the initial state, the final states and the transition function, where the transition function is given by a table in which each row consists of four quintuples; $(q_i, 0, q_j^1, t^1, s^1)$, $(q_i, 0, q_j^2, t^2, s^2)$, $(q_i, 1, q_j^3, t^3, s^3)$, and $(q_i, 1, q_j^4, t^4, s^4)$.

Let $g(M, x, t)$ be 0 if $t \notin \{0, 1\}$; otherwise, let $g(M, x, t)$ be a code of a nondeterministic Turing machine M' that simulates M on input x, but the first nondeterministic step of M is deterministically done, depending on the value of t. The code of M' is obtained from the code of M with a new initial state and one additional row in the transition table, which describes the first step. We can assume that g is one–one, computable in polynomial time, and $|g(M, x, t)| \leqslant |M| + c \log |M| + c$ for some constant c.

We define an *OK* partial order $<$ on strings of the form $\langle M, \langle x, s\rangle, 1^n\rangle$ as follows: $\langle M', \langle x', s'\rangle, 1^m\rangle < \langle M, \langle x, s\rangle, 1^n\rangle$ if $x' = x$, $s' = st$, where $t \in \{0, 1\}$, $M' = g(M, x, t)$, and $|s'| + m = |s| + n$, i.e., $m = n - 1$. Note that every string of the form $\langle M, \langle x, \varepsilon\rangle, 1^n\rangle$ is the largest string in this order $<$. Each chain starting from $\langle M, \langle x, \varepsilon\rangle, 1^n\rangle$ has at most length n.

Now let N be an oracle Turing machine which, on input $F = \langle M, \langle x, s\rangle, 1^n\rangle$, works as follows: if $n = 0$, then N accepts F exactly when M is in an accepting configuration after $|s|$ deterministic steps; otherwise, N computes two strings $F^0 = \langle g(M, s, 0), \langle x, s0\rangle, 1^{n-1}\rangle$ and $F^1 = \langle g(M, s, 1), \langle x, s1\rangle, 1^{n-1}\rangle$ and accepts exactly when either F^0 or F^1 (or both) appears to be in the oracle.

Clearly N reduces RH^k to RH^k by querying only strings which are smaller than input with respect to $<$. It remains to show that, for some polynomial q, $\mu_{\mathrm{RH}}(F') \geqslant \Sigma_F \mu_{\mathrm{RH}}(F)/q(|F|)$, where F ranges over all strings which are reduced to F' by N. From the construction of the strings F^0 and F^1, it follows immediately that they are only asked on input F. Therefore, it suffices to show that $\mu_{\mathrm{RH}}(F^0) \geqslant \mu_{\mathrm{RH}}(F)/q(|F^0|)$. Let $p(\langle M, \langle x, s\rangle, 1^n\rangle) = (|M| + 1)^2\,(|\langle x, s\rangle| + 1)^2\,(n+1)^2$. Thus,

$$\begin{aligned}
\mu_{\mathrm{RH}}(F^0) &= \mu_{\mathrm{RH}}(\langle g(M, s, 0), \langle x, s0\rangle, 1^{n-1}\rangle)\\
&\geqslant \frac{c}{p(\langle g(M, s, 0), \langle x, s0\rangle, 1^{n-1}\rangle)}\\
&\quad \times 2^{-(|g(M, s, 0)| + |\langle x, s0\rangle|)}\\
&\geqslant \frac{c}{p(\langle g(M, s, 0), \langle x, s0\rangle, 1^{n-1}\rangle)}\\
&\quad \times 2^{-(|M| + c\log|M| + |\langle x, s\rangle| + c + 1)}\\
&\geqslant \frac{c}{2^{c+1} \cdot p(\langle g(M, s, 0), \langle x, s0\rangle, 1^{n-1}\rangle) \cdot |M|^c}\\
&\quad \times 2^{-(|M| + |\langle x, s\rangle|)}\\
&\geqslant \frac{\mu_{\mathrm{RH}}(\langle M, \langle x, s\rangle, 1^n\rangle)}{q(\langle g(M, s, 0), \langle x, s0\rangle, 1^{n-1}\rangle)} = \frac{\mu_{\mathrm{RH}}(F)}{q(|F^0|)}
\end{aligned}$$

for some polynomial q. ▮

Wang and Belanger [38] show that the following $\langle \mathrm{NP}, \text{P-comp}\rangle$-complete problems are polynomially isomorphic to each other: the *randomized bounded halting problem* [13, 16, 4], the *randomized tiling problem* [21, 22, 16], the *randomized Post correspondence problem* [16], and the *randomized word problem for Thue systems* [38]. From Theorem 5.4, we immediately conclude the following.

Corollary 5.5. *The following randomized decision problems are all self-reducible*: *the randomized tiling problem, the randomized Post correspondence problem, and the randomized Word problem for Thue systems.*

We note that if every pair of $\langle \mathrm{NP}, \text{P-comp}\rangle$-complete problems is polynomially isomorphic, then every $\langle \mathrm{NP}, \text{P-comp}\rangle$-complete problem is self-reducible.

6. AVERAGE POLYNOMIAL TIME HIERARCHY

The Meyer–Stockmeyer polynomial-time hierarchy is introduced in [25] based on polynomial-time deterministic and nondeterministic Turing reducibilities and is a central concept in worst-case complexity theory. Here, Turing reductions are used to define new classes over P and NP in an analogous way to the Kleene arithmetical hierarchy.

The theory of average NP-completeness can be similarly generalized to an arbitrary level of the Meyer–Stockmeyer polynomial-time hierarchy by using Turing reducibilities among randomized decision problems. We have already seen in Section 5 that all the classes $\langle \Sigma_k^{\mathrm{p}}, \text{P-comp}\rangle$, $k > 0$ have $\leqslant_m^{\mathrm{p}}$-complete sets. It is natural to ask whether, e.g., $\langle \Sigma_k^{\mathrm{p}}, \text{P-comp}\rangle$ is contained in an average-case version of Δ_k^{p} or not. To answer this question, we introduce a notion of an *average polynomial-time hierarchy*, which is based on average polynomial-time Turing reducibilities discussed in Section 4, in analogy with the Meyer–Stockmeyer polynomial-time hierarchy.

First we define a relativization of the average-case complexity classes, $\mathrm{Aver}\langle \mathrm{P}, \mathscr{F}\rangle$ and $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle$, to an oracle (E, ν).

Definition 6.1. Let (E, ν) be a randomized decision problem and let $\mathscr{F}$ be a set of density functions:

1. $\langle \mathrm{P}, \mathscr{F}\rangle^{(E,\nu)} = \{(D, \mu) \mid \mu \in \mathscr{F}, (D, \mu) \leqslant_T^{\mathrm{p}} (E, \nu)\}$.
2. $\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)} = \{(D, \mu) \mid \mu \in \mathscr{F}, (D, \mu) \leqslant_T^{\mathrm{np}} (E, \nu)\}$.
3. $\mathrm{Aver}\langle \mathrm{P}, \mathscr{F}\rangle^{(E,\nu)} = \{(D, \mu) \mid \mu \in \mathscr{F}, (D, \mu) \leqslant_T^{\mathrm{p,av}} (E, \nu)\}$.
4. $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)} = \{(D, \mu) \mid \mu \in \mathscr{F}, (D, \mu) \leqslant_T^{\mathrm{np,av}} (E, \nu)\}$.

From the definitions of Turing reducibilities, it immediately follows that, for any randomized problem (E, ν), $\mathrm{Aver}\langle \mathrm{P}, \mathscr{F}\rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, \mathscr{F}\rangle^{(E,\nu)}$, $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle \subseteq \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)}$, $\langle \mathrm{P}, \mathscr{F}\rangle^{(E,\nu)} \subseteq \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)}$, and $\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)} \subseteq \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)}$.

Definition 6.2. Let $\mathscr{C}$ be a class of randomized decision problems and let $\mathscr{F}$ be a set of density functions:

1. $\langle \mathrm{P}, \mathscr{F} \rangle^{\mathscr{C}} = \{ (D, \mu) \mid \exists (E, \nu) \in \mathscr{C} [(D, \mu) \in \langle \mathrm{P}, \mathscr{F} \rangle^{(E, \nu)}] \}$.
2. $\langle \mathrm{NP}, \mathscr{F} \rangle^{\mathscr{C}} = \{ (D, \mu) \mid \exists (E, \nu) \in \mathscr{C} [(D, \mu) \in \langle \mathrm{NP}, \mathscr{F} \rangle^{(E, \nu)}] \}$.
3. $\mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle^{\mathscr{C}} = \{ (D, \mu) \mid \exists (E, \nu) \in \mathscr{C} [(D, \mu) \in \mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle^{(E, \nu)}] \}$.
4. $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle^{\mathscr{C}} = \{ (D, \mu) \mid \exists (E, \nu) \in \mathscr{C} [(D, \mu) \in \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle^{(E, \nu)}] \}$.

Lemma 4.6 immediately yields the following closure properties.

Lemma 6.3. *Let $\mathscr{F}$ be a set of density functions*:

1. $\mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle^{\mathrm{Aver}\langle \mathrm{P}, * \rangle} = \mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle^{\langle \mathrm{P}, * \rangle} = \mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle$.
2. $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle^{\langle \mathrm{P}, * \rangle} = \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle$.

Up to now, it is unknown whether "in the unrelativized world" $\langle \mathrm{NP}, \text{P-comp} \rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, \text{P-comp} \rangle$ or not. Recall that if $\mathrm{E} \neq \mathrm{NE}$, then $\langle \mathrm{NP}, \text{P-comp} \rangle \not\subseteq \mathrm{Aver}\langle \mathrm{P}, \text{P-comp} \rangle$ [4]. Here we give two contradicting relativized results: an inclusion and a separation.

Theorem 6.4. 1. $\langle \mathrm{NP}, \text{P-comp} \rangle^{(A, \mu)} \subseteq \mathrm{Aver}\langle \mathrm{P}, \text{P-comp} \rangle^{(A, \mu)}$ *for some randomized problem* (A, μ).

2. $\langle \mathrm{NP}, \text{P-comp} \rangle^{(B, \nu)} \not\subseteq \mathrm{Aver}\langle \mathrm{P}, \text{P-comp} \rangle^{(B, \nu)}$ *for some randomized problem* (B, ν).

Proof. (1) The desired oracle (A, μ) is a special version of the randomized halting problem. Let $\langle N, x, 1^t \rangle$ be in A if the nondetertninistic Turing machine N with oracle A accepts x in less than t steps. Note that this is a valid definition since N can make only queries smaller than $1^t (< \langle N, x, 1^t \rangle)$. We define $\mu(\langle N, x, 1^t \rangle) \propto (|N| + 1)^{-2} (|x| + 1)^{-2} (t + 1)^{-2} \cdot 2^{-(|N| + |x|)}$.

Recall the proof that the randomized halting problem is $\leqslant^{\mathrm{p}}_{m}$-hard for $\langle \mathrm{NP}, \text{P-comp} \rangle$ [4, 13, 16, 38]. By exactly the same argument, it follows that (A, μ) is $\leqslant^{\mathrm{p}}_{m}$-hard for $\langle \mathrm{NP}, \text{P-comp} \rangle^{(A, \mu)}$. Therefore, $\langle \mathrm{NP}, \text{P-comp} \rangle^{(A, \mu)} = \langle \mathrm{P}, \text{P-comp} \rangle^{(A, \mu)}$.

Let B be the oracle set used by Baker *et al.* [1] to separate P from NP using the tally set $L(B) = \{ 0^n \mid \exists y [|y| = n \wedge y \in B] \}$ in $\mathrm{NP}^B - \mathrm{P}^B$ and let $\nu(x) \propto (|x| + 1)^{-2}$ if $x \in B$; otherwise, $(|x| + 1)^{-2} \cdot 2^{-|x|}$. Consider the randomized problem $(L(B), \eta)$, where $\eta(x) \propto (|x| + 1)^{-2}$ if $x \in \{0\}^*$, and 0 otherwise. Clearly $(L(B), \eta)$ is in $\langle \mathrm{NP}, \text{P-comp} \rangle^{(B, \nu)}$, and, thus, it is in $\mathrm{Aver}\langle \mathrm{NP}, \text{P-comp} \rangle^{(B, \nu)}$. Now assume that $(L(B), \eta)$ belongs to $\mathrm{Aver}\langle \mathrm{P}, \text{P-comp} \rangle^{(B, \nu)}$. There exist a deterministic Turing machine M which witnesses that $(L(B), \eta) \leqslant^{\mathrm{np,\,av}}_{T} (B, \nu)$. Since M is polynomial-time bounded on η-average, we have $L(B) \in \mathrm{P}^B$. This is a contradiction against the fact that $L(B) \notin \mathrm{P}^B$. ▮

We now give a definition of an *average polynomial-time hierarchy*, which is an average-case analogue of the Meyer–Stockmeyer polynomial-time hierarchy in worst-case complexity theory.

Definition 6.5. Let $k > 1$ and let $\mathscr{F}$ be a set of density functions:

1. $\mathrm{Aver}\langle \Delta^{\mathrm{p}}_{0}, \mathscr{F} \rangle = \mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{0}, \mathscr{F} \rangle = \mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle$.
2. $\mathrm{Aver}\langle \Delta^{\mathrm{p}}_{k}, \mathscr{F} \rangle = \mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle^{\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k-1}, * \rangle}$.
3. $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k}, \mathscr{F} \rangle = \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle^{\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k-1}, * \rangle}$.
4. $\mathrm{Aver}\langle \mathrm{PH}, \mathscr{F} \rangle = \bigcup_{k \geqslant 0} \mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k}, \mathscr{F} \rangle$.

We remark here that oracle sets used in this definition are not restricted to the class $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k-1}, \mathscr{F} \rangle$ since the domination condition of Turing reducibility already puts a constraint on the complexity of the density function of the oracle.

Note also that Lemma 4.6(1) can be easily extended to the class $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k}, \mathscr{F} \rangle$, namely, $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k}, \mathscr{F} \rangle$ is closed under $\leqslant^{\mathrm{p,av}}_{m}$-reductions.

The following two propositions give reasonable evidence that the average polynomial-time hierarchy above defined has a structure similar to that of the worst-case polynomial-time hierarchy.

Proposition 6.6. *Let $k \geqslant 1$ and let $\mathscr{F}$ be any set of density functions*:

1. $\mathrm{Aver}\langle \Delta^{\mathrm{p}}_{k}, \mathscr{F} \rangle \subseteq \mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k}, \mathscr{F} \rangle$.
2. $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{k}, \mathscr{F} \rangle \subseteq \mathrm{Aver}\langle \Delta^{\mathrm{p}}_{k+1}, \mathscr{F} \rangle$.

Proof. The proposition follows immediately from Definition 6.5. ▮

Proposition 6.7. *Let $\mathscr{F}$ be any class of density functions, then* $\mathrm{Aver}\langle \mathrm{BPP}, \mathscr{F} \rangle \subseteq \mathrm{Aver}\langle \Sigma^{\mathrm{p}}_{2}, \mathscr{F} \rangle$.

Proof. Let (D, μ) be an arbitrary problem in $\mathrm{Aver}\langle \mathrm{BPP}, \mathscr{F} \rangle$. By Proposition 2.7, there exists a polynomial p, a p-time bounded deterministic Turing machine M, and a function f such that f is computable in time polynomial on μ-average, and for all x, there are more than $2^{f(x) - |x|}$ strings w of length $f(x)$ such that $x \in D$ if and only if M accepts $\langle x, w \rangle$. It suffices show that $(D, \mu) \leqslant^{\mathrm{np,\,av}}_{T} (E, \nu)$ for some decision problem $(E, \nu) \in \mathrm{Aver}\langle \mathrm{NP}, * \rangle$.

We first define a nondeterministic oracle Turing machine M_0 with an oracle set E as follows. On input x, M_0 first computes the value $f(x)$, then it guesses a nonnegative integer m, and distinct strings $u_1, ..., u_{f(x)}$ and a string w of length $f(x)$, and queries the string $\langle x, u_1 \cdots u_{f(x)}, w \rangle$ to the oracle E. If the string is in E, then M_0 simulates M on input $\langle x, w \rangle$; otherwise, it rejects the input. Clearly M_0 is polynomial-time bounded on μ-average since f is computable in time polynomial on μ-average.

The desired oracle E is defined as the set computed by the nondeterministic machine M_1 that works as follows: on input $\langle x, u_1 \cdots u_m, w\rangle$, M_1 guesses v of length m and $x_1, ..., x_m$ of length $|x|$, and checks if $M(\langle x_i, u_i \oplus v\rangle) \neq M(\langle x_i, w\rangle)$, for all $i \leqslant m$, where $u \oplus v$ denotes the bitwise addition of u and v. If this is true, then M_0 accepts the input; otherwise, M_0 rejects the input. Then, we have $D = L(M_0, E)$ (for a proof, see [2, pp. 170–173]). Note that M_1 is polynomial-time bounded, and therefore, $E \in \mathrm{NP}$.

We next define the desired density function ν on E. Take the density function μ' induced from μ, M_0 and E and set $\nu(z) = \mathrm{Prob}_{\mu'}[\{(x, y) \mid z \in Q(M_0, E, x, y)\}]$. Hence, we have $(D, \mu) \leqslant_T^{\mathrm{np,\,av}} (E, \nu)$, and consequently (D, μ) is in Aver $\langle \mathrm{NP}, \mathscr{F}\rangle^{(E,\nu)} \subseteq \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle^{\mathrm{Aver}\langle \mathrm{NP},\, *\rangle} = \mathrm{Aver}\langle \Sigma_2^{\mathrm{p}}, \mathscr{F}\rangle$. ∎

The definition of the average polynomial-time hierarchy implies that any average-case complexity class of the hierarchy contains its associated worst-case complexity class together with a set of density functions.

LEMMA 6.8. *Let $k \geqslant 1$ and let $\mathscr{F}$ be any set of density functions*:

1. $\langle \Delta_k^{\mathrm{p}}, \mathscr{F}\rangle \subseteq \langle \mathrm{P}, \mathscr{F}\rangle^{\langle \Sigma_{k-1}^{\mathrm{p}},\, *\rangle} \subseteq \mathrm{Aver}\langle \mathrm{P}, \mathscr{F}\rangle^{\langle \Sigma_{k-1}^{\mathrm{p}},*\rangle} \subseteq \mathrm{Aver}\langle \Delta_k^{\mathrm{p}}, \mathscr{F}\rangle$.
2. $\langle \Sigma_k^{\mathrm{p}}, \mathscr{F}\rangle \subseteq \langle \mathrm{NP}, \mathscr{F}\rangle^{\langle \Sigma_{k-1}^{\mathrm{p}},\, *\rangle} \subseteq \mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle^{\langle \Sigma_{k-1}^{\mathrm{p}},\, *\rangle} \subseteq \mathrm{Aver}\langle \Sigma_k^{\mathrm{p}}, \mathscr{F}\rangle$.

Proof. We show the claim (2) since the proof of (1) is analogous. The proof proceeds by induction on k. The case $k = 1$ is obvious.

Let $k > 1$ and assume that $(A, \mu) \in \langle \Sigma_k^{\mathrm{p}}, \mathscr{F}\rangle$. There is a set $B \in \Sigma_{k-1}^{\mathrm{p}}$, a polynomial p, and a nondeterministic oracle Turing machine M which is p-time bounded such that M computes A with oracle B.

Take μ' induced from μ, M and B as in Definition 4.3. A density function ν on the oracle B is given by $\nu(z) = \mathrm{Prob}_{\mu'}[\{(x, y) \mid z \in Q(M, B, x, y)\}]$. Clearly $(A, \mu) \leqslant_T^{\mathrm{np}} (B, \nu)$. Since $(B, \nu) \in \langle \Sigma_{k-1}^{\mathrm{p}}, *\rangle$, we have $(A, \mu) \in \langle \mathrm{NP}, \mathscr{F}\rangle^{\langle \Sigma_{k-1}^{\mathrm{p}},\, *\rangle}$. Therefore, (A, μ) is in $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F}\rangle^{\langle \Sigma_{k-1}^{\mathrm{p}},\, *\rangle}$. Since, by induction hypothesis, $\langle \Sigma_{k-1}^{\mathrm{p}}, *\rangle$ is included in $\mathrm{Aver}\langle \Sigma_{k-1}^{\mathrm{p}}, *\rangle$, we have $(A, \mu) \in \mathrm{Aver}\langle \Sigma_k^{\mathrm{p}} \mathscr{F}\rangle$. ∎

Lemma 3.10 can be extended into an arbitrary level of the average polynomial-time hierarchy. We first see a key lemma.

LEMMA 6.9. *Let $k \geqslant 0$ and assume that $(A, \mu) \leqslant_T^{\mathrm{np,\,av}} (B, \nu)$ and $(B, \nu) \in \mathrm{Aver}\langle \Sigma_k^{\mathrm{p}}, *\rangle$. For any set S and any polynomial q, there exist sets $C_0 \in \Sigma_{k+1}^{\mathrm{p}}$, $C_1 \in \Pi_{k+1}^{\mathrm{p}}$ and S' such that $A \cap S' \subseteq C_0 \subseteq A$, $\overline{A} \cap S' \subseteq C_1 \subseteq \overline{A}$ and $\mu(S^n) - \mu(S'^n) \leqslant 1/q(n)$ for all $n \in \mathbb{N}$.*

Proof. The proof proceeds by induction on k. The base case $k = 0$ follows from Lemma 3.10.

Let $k \geqslant 1$ and assume that $(A, \mu) \in \mathrm{Aver}\langle \Sigma_{k+1}^{\mathrm{p}}, *\rangle$. By definition, there is a nondeterministic Turing machine M and a randomized decision problem $(B, \nu) \in \mathrm{Aver}\langle \Sigma_k^{\mathrm{p}}, *\rangle$ such that $(A, \mu) \leqslant_T^{\mathrm{np,\,av}} (B, \nu)$ via M. Let p be a polynomial such that $\mathrm{Prob}_{\mu}[\{x \mid \mathrm{Time}_M^B(x) > p(|x| \cdot m)\}] < 1/m$ for any positive real number m.

Let μ' be the density function induced from μ, M, and B. Furthermore, let υ be a density function, such that $\mu' \preccurlyeq^{\mathrm{p},\mu'} \upsilon$ and $\upsilon(z) \geqslant \mathrm{Prob}_{\upsilon}[\{(x, y) \mid z \in Q(M, B, x, y)\}]$. Since $\mu' \preccurlyeq^{\mathrm{p},\mu'} \upsilon$, there exists a function f which is r on μ'-average such that $\upsilon(x, y) \cdot f(x, y) \geqslant \mu'(x, y)$, where r is a polynomial.

Claim 2. There exists a nondeterministic Turing machine M' and a randomized problem (B', υ') in Aver *$\langle \Sigma_k^{\mathrm{p}}, \mathscr{F}\rangle$ such that $(A, \mu) \leqslant_T^{\mathrm{np,\,av}} (B', \upsilon')$ via M', and all strings queried by M' with oracle B' on input x are of length greater than $|x|$.*

Proof. Let $B' = \{z01^n \mid z \in B\}$ and let

$$\nu'(w) = \begin{cases} \nu(z) \cdot \dfrac{\mathrm{Prob}_{\mu'}[\{(x, y) \mid |x| = n \wedge z \in Q(M, B, x, y)\}]}{\mathrm{Prob}_{\mu'}[\{(x, y) \mid z \in Q(M, B, x, y)\}]} & \\ \quad \text{if } w = z01^n \text{ for some } z \text{ and } n, & \\ 0 \quad \text{otherwise.} & \end{cases}$$

We define a new oracle Turing machine M' which works as follows: on input x, simulate M on input x, and whenever M queries a string z, M' queries $z01^{|x|}$ to the oracle. It is easy to see that $(A, \mu) \leqslant_T^{\mathrm{np,\,av}} (B', \upsilon')$ via M'. We next show that (B', υ') is in $\mathrm{Aver}\langle \Sigma_k^{\mathrm{p}}, *\rangle$. Note that (B', υ') is average polynomial-time many–one reducible to (B, ν). Since Aver $\langle \Sigma_k^{\mathrm{p}}, *\rangle$ is closed under $\leqslant_m^{\mathrm{p}}$-reduction, (B', ν') is in $\mathrm{Aver}\langle \Sigma_k^{\mathrm{p}}, *\rangle$. ∎

Therefore, without loss of generality, we assume that M queries only strings of length greater than length of inputs. For each $n \in \mathbb{N}$, it holds that

$$\mathrm{Prob}_{\mu}[\{x \mid \mathrm{Time}_M^B(x) > p(|x| \cdot 3q(n))\}] < 1/3q(n),$$

$$\mathrm{Prob}_{\mu'}[\{(x, y) \mid f(x, y) > r(|\langle x, y\rangle| \cdot 3q(n))\}] > 1/3q(n).$$

Let Flip(x) be Acc(M, B, x) if $x \in D$, or else let it be Rej(M, B, x). We define a set T_n by

$$T_n = \{(x, y) \mid x \in S^n \text{ and } \mathrm{Time}_M^B(x) \leqslant p(|x| \cdot 3q(n)) \text{ and}$$
$$y \in \mathrm{Flip}(x) \text{ and } \mu'(x, y) \leqslant r(|\langle x, y\rangle| \cdot 3q(n)) \cdot \upsilon(x, y)\},$$

and let $T = \bigcup_{n>0} T_n$. It is easy to see that $\mu(S^n) - \mu'(T_n) \leqslant 2/3q(n)$. For any pair (x, y) in T,

$$\mu'(x, y) \leqslant r(|\langle x, y\rangle| \cdot 3q(n)) \cdot \upsilon(x, y)$$
$$\leqslant r((2n + p(n \cdot 3q(n) + 1)) \cdot 3q(n)) \cdot \upsilon(x, y)$$

since $|y| \leqslant p(|x| \cdot 3q(n))$, and thus we have $\mu'(x, y) \leqslant s(n) \cdot \upsilon(x, y)$ for some polynomial s.

Now let $Z = \{z \mid \exists (x, y) \in T[z \in Q(M, B, x, y)]\}$. Recall that (B, ν) is in $\mathrm{Aver}\langle \Sigma_k^{\mathrm{p}}, *\rangle$. Hence, it follows by induction hypothesis that, for any polynomial l, there exist a subset Z' of Z and sets $C'_0 \in \Sigma^{\mathrm{p}}_k$ and $C'_1 \in \Pi^{\mathrm{p}}_k$ such that $B \cap Z' \subseteq C'_0 \subseteq B$, $\bar{B} \cap Z' \subseteq C'_1 \subseteq \bar{B}$ and $\nu(Z^n) - \nu(Z'^n) \leqslant 1/l(n)$. Denote by Z_n (resp. Z'_n) the set of all strings in Z (resp. Z') whose lengths are between n and $p(n \cdot 3q(n))$. Now choose $l(n) = 3q(n) \cdot s(n) \cdot p(n \cdot 3q(n))$ for all $n \in \mathbb{N}$. Then, we have $\nu(Z_n) - \nu(Z'_n) \leqslant 1/3q(n)\, s(n)$.

Now we define $T' = \{(x, y) \in T \mid Q(M, B, x, y) \subseteq Z'\}$. Note that, for all $n \in \mathbb{N}$,

$$T_n - T'_n \subseteq \{(x, y) \in T_n \mid Q(M, B, x, y) \cap (Z_n - Z'_n) \neq \varnothing\}.$$

Hence, it holds that $\nu(Z_n - Z'_n) \geqslant \upsilon(T_n - T'_n)$ for all $n \in \mathbb{N}$. Then, for every n,

$$\begin{aligned} 1/3q(n) &\geqslant (\nu(Z_n) - \nu(Z'_n)) \cdot s(n) \\ &\geqslant (\upsilon(T_n) - \upsilon(T'_n)) \cdot s(n) \geqslant \mu'(T_n) - \mu'(T'_n). \end{aligned}$$

The desired set S' is defined as $S' = \{x \mid \exists y[(x, y) \in T']\}$. Note that $\mu(S'_n) \geqslant \mu'(T'_n)$ holds. It immediately follows from our definition that $S' \subseteq S$ and $\mu(S^n) - \mu(S'^n) \leqslant 1/q(n)$ for all $n \in \mathbb{N}$.

Let M_0 be an oracle Turing machine with oracle X defined as follows: on input x, M_0 simulates M on x in time $p(|x| \cdot 3q(|x|))$, and whenever M queries a string z, M_0 queries both $\langle 0, z\rangle$ and $\langle 1, z\rangle$ to its oracle X. If $\langle 0, z\rangle \in X$ and $\langle 1, z\rangle \notin X$, M_0 continues the simulation with assuming that the oracle answer is "yes"; if $\langle 0, z\rangle \notin X$ and $\langle 1, z\rangle \in X$, then it continues the simulation with the oracle answer "no"; otherwise, it immediately rejects the input x. The machine M_0 accepts x exactly when M halts and accepts it. Similarly, we define a machine M_1 by interchanging the oracle answers and accepts the input if M halts in time $p(|x| \cdot 3q(|x|))$ and rejects it. Now let $C_0 = L(M_0, C'_0 \oplus C'_1)$ and $C_1 = L(M_1, C'_0 \oplus C'_1)$. By definition of the oracle machines M_0 and M_1, it follows that $A \cap S' \subseteq C_0 \subseteq A$ and $\bar{A} \cap S' \subseteq C_1 \subseteq \bar{A}$. ▮

Proposition 6.10. *For $k \geqslant 1$,* Aver$\langle \Delta^{\mathrm{p}}_k, *\rangle$ *and* Aver$\langle \Sigma^{\mathrm{p}}_k, *\rangle$ *have the sparse interpolation property.*

Proof. We show the case Aver$\langle \Sigma^{\mathrm{p}}_k, *\rangle$ here. The case $k = 1$ follows from Lemma 3.10. Let $k \geqslant 2$ and assume that $(A, \mu_{S,q}) \in$ Aver$\langle \Sigma^{\mathrm{p}}_k, *\rangle$ for a sparse set S and a polynomial q. It follows from Lemma 6.9 that there exists a set $C \in \Sigma^{\mathrm{p}}_k$ and a subset S' of S such that $A \cap S' \subseteq C \subseteq A$ and $\mu_{S,q}(S^n) - \mu_{S,q}(S'^n) \leqslant 1/2q(n)$ for all $n \in \mathbb{N}$. It suffices to show that $S' = S$. Assume that there exists a string $x \in S - S'$. Let $n = |x|$. Since $\mu_{S,q}(x) \geqslant 1/q(|x|)$,

$$\frac{1}{q(n)} \leqslant \mu_{S,q}(S^n) - \mu_{S,q}(S'^n) \leqslant \frac{1}{2q(n)}.$$

This is a contradiction. Hence, $S' = S$. ▮

The average polynomial-time hierarchy allows us to construct an average-version of the *high* and *low hierarchy* in NP [30] to refine the structure within NP. It might be possible that some NP-complete problems with natural distributions which are unknown to be either in Aver$\langle$P, $*\rangle$ or $\langle$NP, P-comp$\rangle$-complete fall into a "low (or high) hierarchy in Aver$\langle$NP, P-comp$\rangle$."

Returning to Levin's fundamental question of whether NP, P-comp$\rangle \subseteq$ Aver$\langle$P, $*\rangle$, we can now raise a more general question of whether $\langle \Sigma^{\mathrm{p}}_k$, P-comp$\rangle \subseteq$ Aver$\langle \Delta^{\mathrm{p}}_k, *\rangle$ holds or not. However, to answer this question turns out to be very hard since the following claim closes the gap between average-case and worst-case.

Theorem 6.11. *For any $k \geqslant 1$,* $\langle \Sigma^{\mathrm{p}}_k \cap$ TALLY, L-comp$\rangle \subseteq$ Aver$\langle \Delta^{\mathrm{p}}_k, *\rangle$ *if and only if* $\Sigma^{\mathrm{p}}_k \cap$ TALLY $\subseteq \Delta^{\mathrm{p}}_k$.

Proof. It suffices to prove the "only if" part of the theorem. Let $\mu(x) = 6/\pi^2(|x| + 1)^2$ if $x \in \{0\}^*$, and 0 otherwise. Clearly μ is in L-comp. Suppose that $\langle \Sigma^{\mathrm{p}}_k \cap$ TALLY, L-comp$\rangle \subseteq$ Aver$\langle \Delta^{\mathrm{p}}_k, *\rangle$, and $A \in \Sigma^{\mathrm{p}}_k \cap$ TALLY. Since $(A, \mu) \in$ Aver$\langle \Delta^{\mathrm{p}}_k, *\rangle$, Proposition 6.10 shows the existence of a set $B \in \Delta^{\mathrm{p}}_k$ such that $A \cap \{0\}^* = B \cap \{0\}^*$. Now define $B' = B \cap \{0\}^*$. Since $A \subseteq \{0\}^*$, we obtain $A = B' \in \Delta^{\mathrm{p}}_k$. ▮

It seems unlikely that $\langle \Sigma^{\mathrm{p}}_k$, P-comp$\rangle \subseteq$ Aver$\langle \Delta^{\mathrm{p}}_k, *\rangle$ since, as is believed, some tally Σ^{p}_k sets might not fall into Δ^{p}_k.

Corollary 6.12. *Let $k \geqslant 1$. If* Aver$\langle \Delta^{\mathrm{p}}_k$, P-comp$\rangle =$ Aver$\langle \Sigma^{\mathrm{p}}_k$, P-comp$\rangle$, *then* $\Sigma^{\mathrm{p}}_k \cap$ TALLY $\subseteq \Delta^{\mathrm{p}}_k$.

7. REAL POLYNOMIAL TIME COMPUTABILITY

This section establishes a direct link to the classical framework of worst-case complexity theory. This link casts a light on the essential role of average-case analysis in the study of worst-case complexity.

As we have seen, average-case analysis is very sensitive to the selection of distributions. For example, fast decreasing density functions help average polynomial-time bounded machines to solve hard sets; however, there exist sets in EXP that are not solvable in average polynomial-time if we choose, e.g., a density function ν defined as $\nu(x) \propto (|x| + 1)^{-2}$ for $x \in \{0\}^*$, and 0 otherwise. This approach toward average-case analysis does not capture an important feature of average-case analysis. To see this feature, we try to abstract a notion of "rare instances" under any "reasonable" distribution in order to make the notion independent from individual distributions, and study its general properties.

Observe that the class P is the largest class of sets which are computable in average polynomial-time with respect to every density function [23]. Here we focus on the class of sets which are deterministically computable in average polynomial-time for every density function in P-comp, and refer to it as a "real P over P-comp." First we introduce a more general notion of "real $\mathscr{C}$ over $\mathscr{F}$."

Definition 7.1. Let $\mathscr{C}$ be a complexity class and let $\mathscr{F}$ be a class of density functions. The *real $\mathscr{C}$ over $\mathscr{F}$*, denoted by $\mathscr{C}_{\mathscr{F}}$, is the class of languages D such that $(D, \mu) \in \mathrm{Aver}\langle \mathscr{C}, * \rangle$ for every $\mu \in \mathscr{F}$.

This new definition formalizes a significant property of the associated average-case complexity classes. The next proposition obviously indicates the importance of this notion.

Proposition 7.2. *Let $\langle \mathscr{C}, \mathscr{F} \rangle$ and $\mathrm{Aver}\langle \mathscr{D}, \mathscr{F} \rangle$ be any randomized and average-case complexity classes, respectively. Then, $\mathscr{C} \subseteq \mathscr{D}_{\mathscr{F}}$ if and only if $\langle \mathscr{C}, \mathscr{F} \rangle \subseteq \mathrm{Aver}\langle \mathscr{D}, \mathscr{F} \rangle$.*

Proof. Assume that $\mathscr{C} \subseteq \mathscr{D}_{\mathscr{F}}$ and (A, μ) is in $\langle \mathscr{C}, \mathscr{F} \rangle$. From the fact that A belongs to $\mathscr{D}_{\mathscr{F}}$, it follows that $(A, \mu) \in \mathrm{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$. Conversely, assume that $\langle \mathscr{C}, \mathscr{F} \rangle \subseteq \mathrm{Aver}\langle \mathscr{D}, * \rangle$. Let D be any a set in $\mathscr{D}$. For every $\mu \in \mathscr{F}$, since $(D, \mu) \in \mathrm{Aver}\langle \mathscr{D}, \mathscr{F} \rangle$, we obtain $(D, \mu) \in \mathrm{Aver}\langle \mathscr{C}, \mathscr{F} \rangle$. Hence, D belongs to $\mathscr{D}_{\mathscr{F}}$. ▌

By Proposition 7.2, Levin's original question of whether $\langle \mathrm{NP}, \text{P-comp} \rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, * \rangle$ is simply rephrased as whether $\mathrm{NP} \subseteq \mathrm{P}_{\text{P-comp}}$ holds or not.

Lemma 7.3. *Let $\mathscr{F}$ be any set of density functions which contains the standard density function*:

1. $\mathrm{P} \subseteq \mathrm{P}_{\mathscr{F}} \subseteq \mathrm{E}$.
2. $\mathrm{NP} \subseteq \mathrm{NP}_{\mathscr{F}} \subseteq \mathrm{NE}$.
3. $\mathrm{BPP} \subseteq \mathrm{BPP}_{\mathscr{F}} \subseteq \mathrm{BPE}$.
4. $\mathrm{PSPACE} \subseteq \mathrm{PSPACE}_{\mathscr{F}} \subseteq \mathrm{ESPACE}$.

Proof. Here we give only the proof of (1) since the rest of the claims follow by a similar argument. Since $\langle \mathrm{P}, \mathscr{F} \rangle \subseteq \mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle$, we have $\mathrm{P} \subseteq \mathrm{P}_{\mathscr{F}}$. Now we show that $\mathrm{P}_{\mathscr{F}} \subseteq \mathrm{E}$. Let A be any set in $\mathrm{P}_{\mathscr{F}}$. Since $(A, \nu_{\mathrm{st}}) \in \mathrm{Aver}\langle \mathrm{P}, \mathscr{F} \rangle$, there exists a polynomial p and a deterministic Turing machine M which is p-time bounded on ν_{st}-average such that M computes A. It clearly holds that, for almost all x,

$$\mathrm{Time}_M(x) \leqslant p(|x|/\nu_{\mathrm{st}}(x)) = p(\pi^2 |x| (|x|+1)^2 \cdot 2^{|x|}/6) \leqslant 2^{c\,|x|}$$

for some adequate constant $c > 0$. Therefore, we have $A \in \mathrm{DTIME}(2^{cn})$. ▌

We call $\{\Delta^{\mathrm{p}}_{k\mathscr{F}}, \Sigma^{\mathrm{p}}_{k\mathscr{F}} \mid k > 0\}$ the *real polynomial-time hierarchy* with respect to a set $\mathscr{F}$ of density functions. The next result immediately follows from Lemma 6.8.

Lemma 7.4. *Let $k > 0$ and let $\mathscr{F}$ be any set of density functions*:

1. $\Delta^{\mathrm{p}}_k \subseteq \Delta^{\mathrm{p}}_{k\mathscr{F}}$.
2. $\Sigma^{\mathrm{p}}_k \subseteq \Sigma^{\mathrm{p}}_{k\mathscr{F}}$.

Tally sets play a significant role in average-case analysis. From Lemma 3.10 and Proposition 6.10, the average-case complexity of tally sets turns out to equal the worst-case complexity of them.

Proposition 7.5. *For every $\mathscr{C} \in \{\Delta^{\mathrm{p}}_k, \Sigma^{\mathrm{p}}_k$, BPP, PSPACE$\}$, $\mathscr{C}_{\text{P-comp}} \cap \mathrm{TALLY} \subseteq \mathscr{C}$.*

Proof. The proof is similar to Theorem 6.11. ▌

Recall that REC-comp denotes the set of recursive density functions, i.e., all "computable" density functions (under Church's thesis). If we take REC-comp as a set of density functions $\mathscr{F}$, then the real computable classes collapse to their worst-case counterparts.

To prove this, we show that if $\mathrm{Aver}\langle \mathscr{C}, \text{REC-comp} \rangle$ has the sparse interpolation property, then $\mathscr{C}_{\text{REC-comp}} \subseteq \mathscr{C}$. In the proof of the following lemma, we use the notion of infinite, recursive, proper hard cores [8]. A set H is called a *proper hard core for A with respect to $\mathscr{C}$* if $H \subseteq A$, and for all $D \in \mathscr{C}$, if $D \subseteq A$, then $|D \cap H|$ is finite.

Lemma 7.6. *Let $\mathrm{Aver}\langle \mathscr{C}, \text{REC-comp} \rangle$ be an average-case complexity class. If $\mathrm{Aver}\langle \mathscr{C}, \text{REC-comp} \rangle$ has the sparse interpolation property, then $\mathscr{C}_{\text{REC-comp}} \subseteq \mathscr{C}$.*

Proof. Suppose that $\mathrm{Aver}\langle \mathscr{C}, \text{REC-comp} \rangle$ has the sparse interpolation property. We show that $\mathscr{C}_{\text{REC-comp}} \subseteq \mathscr{C}$ by leading to a contradiction. Now assume that there exists a set A in $\mathscr{C}_{\text{REC-comp}} - \mathscr{C}$. By [8], there exists an infinite, recursive, proper hard core H for A with respect to $\mathscr{C}$. We note that if $\mathscr{C} = \mathrm{P}$, then H is in the class E (see, e.g., [2]). Thus, for any set $B \in \mathscr{C}$, if $B \subseteq A$, then $B \cap H$ is finite. Now let S be a recursive, infinite, sparse subset of H. Let $q(n) = |S \cap \Sigma^n|$. Consider the density function $\mu_{S,q}$ such that $\mu_{S,q}(x) \propto (|x|+1)^{-2} \cdot q(|x|)^{-1}$ for all $x \in S$, and $\mu_{S,q}(x) = 0$ otherwise. Clearly $\mu_{S,q}, \in$ REC-comp. Since $(A, \mu_{S,q}) \in \mathrm{Aver}\langle \mathscr{C}, \text{REC-comp} \rangle$, there exists an interpolant $B' \in \mathscr{C}$ of A and S. We then have $B' \cap H \supseteq S$, and thus $B' \cap H$ is infinite. This contradicts the fact that H is a proper hard core for A. ▌

Theorem 7.7. *Let $k > 0$*:

1. $\Delta^{\mathrm{p}}_{k\text{REC-comp}} = \Delta^{\mathrm{p}}_k$.
2. $\Sigma^{\mathrm{p}}_{k\text{REC-comp}} = \Sigma^{\mathrm{p}}_k$.
3. $\mathrm{BPP}_{\text{REC-comp}} = \mathrm{BPP}$.
4. $\mathrm{PSPACE}_{\text{REC-comp}} = \mathrm{PSPACE}$.

Proof. By Lemma 7.6, it suffices to show that, for $\mathscr{C} \in \{\Delta^{\mathrm{p}}_k, \Sigma^{\mathrm{p}}_k$, BPP, PSPACE$\}$, $\mathrm{Aver}\langle \mathscr{C}, \text{REC-comp} \rangle$ has the sparse interpolation property. This claim for $\mathscr{C} \in \{$BPP, PSPACE$\}$ follows from Lemma 3.10, and the claim for $\mathscr{C} \in \{\Delta^{\mathrm{p}}_k, \Sigma^{\mathrm{p}}_k\}$ follows from Proposition 6.10. ▌

Theorem 7.7 implies that the definition of the average polynomial-time hierarchy in Section 6 is a reasonable generalization of the worst-case polynomial-time hierarchy.

Note that, in the proof of Lemma 7.6, the complexity of the distribution $\mu_{S,q}$ depends only on the complexity of the complexity core. Since all sets not in P have complexity cores in E, we get the following corollary.

Corollary 7.8. $\mathrm{P}_{\text{E-comp}} = \mathrm{P}$.

Very recently, $\mathrm{P}_{\text{P-comp}}$ is shown to be different from P and NP [31].

At the end of this section, we show that, relative to random oracle, $\mathrm{NP}_{\text{P-comp}}$ is different from $\mathrm{P}_{\text{P-comp}}$ with probability 1. In other words, Lebesgue measure of the set $\{X \mid \mathrm{P}^X_{\mathrm{P}^X\text{-comp}} \neq \mathrm{NP}^X_{\mathrm{P}^X\text{-comp}}\}$ is 1, where P^X-comp denotes the set of density functions whose distributions are polynomial-time computable relative to oracle X.

Definition 7.9. Let X be a set of strings, and let $\mathscr{F}^X$ be a set of density functions relative to X:

1. Let $\mathrm{P}^X_{\mathscr{F}^X}$ be the collection of all sets A such that, for any density function μ in $\mathscr{F}^X$, $(A, \mu) \leqslant^{\mathrm{p,\,av}}_T (X, v)$ for some density function v.

2. Let $\mathrm{NP}^X_{\mathscr{F}^X}$ be the collection of all sets A such that, for any density function μ in $\mathscr{F}^X$, $(A, \mu) \leqslant^{\mathrm{np,\,av}}_T (X, v)$ for some density function v.

Proposition 7.10. *With probability* 1, $\mathrm{P}^X_{\mathrm{P}^X\text{-comp}} \neq \mathrm{NP}^X_{\mathrm{P}^X\text{-comp}}$ *relative to a random oracle X.*

Proof. For any oracle A, $\mathrm{P}^A_{\mathrm{P}^A\text{-comp}} = \mathrm{NP}^X_{\mathrm{P}^X\text{-comp}}$ clearly implies $\mathrm{NP}^A \cap \mathrm{TALLY} \subseteq \mathrm{P}^A$. Bennett and Gill [3] have proven that, relative to a random oracle X, $\mathrm{NP}^X \cap \mathrm{TALLY} \nsubseteq \mathrm{P}^X$ with probability 1. Hence, we get the desired result. ∎

8. CONCLUSIONS AND OPEN PROBLEMS

We have discussed structural properties of average-case complexity classes. Especially reducibilities have played a central role in our study of structural properties of those classes. This paper has introduced an average-case counterpart of the Meyer–Stockmeyer polynomial-time hierarchy based on the deterministic and nondeterministic Turing reducibilities between randomized decision problems, and we have seen that this hierarchy has a structure similar to its counterpart in worst-case complexity. We give some problems still open in this paper:

1. Let D be any Σ^{p}_k-complete set. Does there exist a "natural" density function μ such that (D, μ) is $\leqslant^{\mathrm{p}}_m$-complete for $\langle \Sigma^{\mathrm{p}}_k, \text{P-comp} \rangle$? For example, if A is NP-complete with p-honest reductions, then there exists a density function μ whose distribution is polynomial-time computable relative to #P such that (A, μ) is $\leqslant^{\mathrm{p}}_m$-hard for $\langle \mathrm{NP}, \text{P-comp} \rangle$.

2. Are all $\langle \Sigma^{\mathrm{p}}_k, \text{P-comp} \rangle$-complete problems Turing self-reducible ? Can we extend the notion of polynomial-time Turing self-reducibility for randomized problems by allowing the reduction to be polynomial-time bounded on average?

3. In Lemma 2.6, $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle$ is characterized in terms of both deterministic machine models and logical formulas with the existential quantifier. Can this result be extended to characterize the class $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_k, \mathscr{F} \rangle$?

4. Is $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_k, \mathscr{F} \rangle$ contained in $\mathrm{Aver}\langle \Delta^{\mathrm{p}}_k, \mathscr{F} \rangle$ for some reasonable set $\mathscr{F}$? Recall that if $\mathrm{Aver}\langle \Delta^{\mathrm{p}}_k, \text{P-comp} \rangle$ is equal to $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_k, \text{P-comp} \rangle$, then all tally Σ^{p}_k-sets are in Δ^{p}_k.

5. Is $\mathrm{Aver}\langle \Sigma^{\mathrm{p}}_k, \mathscr{F} \rangle$ different from $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle^{\langle \Sigma^{\mathrm{p}}_{k-1}, * \rangle}$, $k > 1$, for a reasonable set $\mathscr{F}$, such as P-comp?

6. In worst-case complexity theory, many important complexity classes, such as UP, CP and $\oplus$P, are used to classify intractable problems and to investigate their structural properties. Is it reasonable to consider those classes in average-case complexity theory?

7. Very recently, it is shown that $\mathrm{P}_{\text{P-comp}} \neq \mathrm{P}$ [31]. Can we extend this result to show that $\Sigma^{\mathrm{p}}_{k\text{P-comp}} \neq \Sigma^{\mathrm{p}}_k$ or $\Delta^{\mathrm{p}}_{k\text{P-comp}} \neq \Delta^{\mathrm{p}}_k$?

8. Is it possible to show that some NP-complete problem with some natural density function is in the "low hierarchy in $\mathrm{Aver}\langle \mathrm{NP}, \mathscr{F} \rangle$"?

9. What is a reasonable relativization of classes, such as $\mathrm{Aver}\langle \mathrm{BPP}, \mathscr{F} \rangle$ and $\mathrm{Aver}\langle \mathrm{PSPACE}, \mathscr{F} \rangle$?

10. Recall from [17] the definition of the time-complexity of a nondeterministic Turing machine M. Here we define Time_M to be the minimal length of accepting computations of M on input x if one exists; otherwise, $\mathrm{Time}_M(x)$ is always set to 1. Can we develop a theory founded on this type of nondeterministic Turing machines which are polynomial-time (or polynomial-space) bounded on μ-average? In this setting, for example, we can prove that $\mathrm{Aver}\langle \mathrm{P}, * \rangle \neq \mathrm{Aver}\langle \mathrm{NP}, * \rangle$ by choosing a nonrecursive, recursively enumerable set $A = \{M(0), M(1), ...\}$ by a deterministic machine M and defining a density function μ by: $\mu(x) \propto (|x|+1)^{-2} \cdot 2^{-\Sigma^n{}_{i=0}\mathrm{Time}_M(i)}$ if $x \in A$ and $n = \min\{k \mid M(k) = x\}$, or else 0.

11. An alternative definition of real polynomial-time hierarchy is given by: $\Delta^{\mathrm{p}}_{k\mathscr{F}} = \mathrm{P}^{\Sigma^{\mathrm{p}}_{k-1}{}^{\mathscr{F}}}_{\mathscr{F}}$ and $\Sigma^{\mathrm{p}}_{k\mathscr{F}} = \mathrm{NP}^{\Sigma^{\mathrm{p}}_{k-1}{}^{\mathscr{F}}}_{\mathscr{F}}$. Is it possible to develop a theory based on these $\Delta^{\mathrm{p}}_{k\mathscr{F}}$ and $\Sigma^{\mathrm{p}}_{k\mathscr{F}}$? For example, it is not hard to show that $\Delta^{\mathrm{p}}_{k\text{REC-comp}} = \Delta^{\mathrm{p}}_k$ and $\Sigma^{\mathrm{p}}_{k\text{REC-comp}} = \Sigma^{\mathrm{p}}_k$ also in this setting.

ACKNOWLEDGMENTS

The authors thank Professor José Balcázar and Professor Uwe Schöning for organizing workshops in Barcelona and Ulm in 1991, where many discussions on average-case complexity theory took place, and they thank the anonymous referees who gave them useful comments and also criticism for an early draft. The second author is specially grateful to Professor Uwe Schöning for allowing him to stay at the Univetsität Ulm from April to August in 1991.

REFERENCES

1. T. Baker, G. Gill, and R. Solovay, Relativizations of the P = ?NP question, *SIAM J. Comput.* **4** (1975), 431–442.
2. J. L. Balcaźar, J. Diáz, and J. Gabarró, "Structural Complexity I, II," Springer-Verlag, Berlin/Heidelberg, 1988, 1990.
3 C. H. Bennett and J. Gill, Relative to a random oracle A, $P^A \neq NP^A \neq$ co-NP^A with probability 1, *SIAM J. Comput.* 10 (1981), 96–113.
4. S. Ben-David, B. Chor, O. Goldreich, and M. Luby, On the theory of average case complexity, *J. Comput. System Sci.* **44** (1992), 193–219.
5. L. Berman and J. Hartmanis, On isomorphism and density of NP and other complete sets, *SIAM J. Comput.* **6** (1977), 305–322.
6. A. Blass and Y. Gurevich, Randomized reductions of search problems, *SIAM J. Comput.* **22** (1993), 949–975.
7. B. Bollobas, T.I. Fenner, and A. M. Frieze, An algorithm for finding Hamiltonian cycle in a random graph, *in* "Proceedings, 17th ACM Symposium on Theory of Computing, 1985," pp. 430–439.
8. R. Book, D. Z. Du, and D. Russo, On polynomial and generalized complexity cores, *in* "Proceedings, 3rd Conference on Structure in Complexity Theory, 1988," pp. 236–250.
9. J. Franco and M. Paull, Probabilistic analysis of the Davis Putnum procedure for solving the satisfiability problem, *Discrete Appl. Math.* **5** (1983), 77–87.
10. M. R. Garey and D. J. Johnson, "Computers and Intractability, A Guide to the Theory of NP-Completeness," Freeman, New York, 1979.
11. J. Gill, Computational complexity of probabilistic Turing machines, *SIAM J. Comput.* **6** (1977), 675–695.
12. M. Goldmann, P. Grape, and J.Håstad, On average time hierarchy, *Inform. Proces. Lett.* **49** (1994), 15–20.
13. O. Goldreich, "Towards a Theory of Average Case Complexity (A Survey)," Technical Report No. 507, Israel Institute of Technology, 1988.
14. P. Grape, "Two Results in Average Case Complexity," Technical Report TRITA-NA-9105, Royal Institute of Technology, Stockholm, Sweden, 1991.
15. Y. Gurevich, Complete and incomplete randomized NP problems, *in* "Proceedings, 28th IEEE Symposium on Foundations of Computer Science, 1987," pp. 111–117.
16. Y. Gurevich, Average case complexity, *J. Comput. System Sci.* **42** (1991), 346–398.
17. J. E. Hopcroft and J. D. Ullman, "Introduction to Automata Theory, Languages, and Computation," Addison–Wesley, Reading, MA, 1979.
18. R. Impagliazzo and L. A. Levin, No better ways to generate hard NP-instances than picking uniformly at random, *in* "Proceedings, 31th IEEE Symposium on Foundations of Computer Science, 1990," pp. 812–821.
19. D. Joseph and P. Young, Some remarks on witness functions for non-polynomial and noncomplete sets in NP, *Theoret. Comput. Sci.* **39** (1985), 225–237.
20. K. I. Ko and H. Friedman, Computational complexity of real functions, *Theoret. Comput. Sci.* **20** (1982), 323–352.
21. L. Levin, Problems, complete in "average" instance, *in* "Proceedings, 16th ACM Symposium on Theory of Computing, 1984," p. 465.
22. L. Levin, Average case complete problems, *SIAM J. Comput.* **15** (1986), 285–286.
23. M. Li and P. M. B. Vitányi, Average case complexity under the universal distribution equals worst-case complexity, *Inform. Process. Lett.* **42** (1992) 145–149.
24. A. Meyer and M. Paterson, "With What Frequency Are Apparently Intractable Problems Difficult?," Technical Report MIT/LCS/TM-126, MIT, 1979.
25. A. R. Meyer and L. J. Stockmeyer, The equivalence problem for regular expressions with squaring requires exponential time, *in* "Proceedings, 13th IEEE Symposium on Switching and Automata Theory, 1972," pp. 125–129.
26. M. Mundhenk and R. Schuler, Random languages for non-uniform complexity classes, *J. Complexity* **7** (1991), 296–310.
27. K. W. Regan, Minimum-complexity pairing functions, *J. Comput. System Sci.* **45** (1992), 285–295.
28. R. Reischuk and C. Schindelhauer, Precise average case complexity, *in* "Proceedings, 10th Symposium on Theoretical Aspect of Computer Science," Lecture Notes in Computer Science, Vol. 665, New York/Berlin, Springer-Verlag, 1993, pp. 650–661.
29. R. E. Schapire, "The Emerging Theory of Average-Case Complexity," Technical Report, MIT/LCS/TM-431, MIT, 1990.
30. U. Schöning, A low and a high hierarchy within NP, *J. Comput. System Sci.* **27** (1983), 14–28.
31. R. Schuler, Some properties of sets tractable under every polynomial-time computable distribution, *Inform. Process. Lett.* **55** (1995), 179–184.
32. L. J. Stockmeyer, The polynomial-time hierarchy, *Theoret. Comput. Sci.* **3** (1977), 1–22.
33. L. Valiant, The complexity of computing the permanent, *Theoret. Comput. Sci.* **5** (1979), 189–201.
34. R. Venkatesan and L. Levin, Random instances of a graph coloring problem are hard, *in* "Proceedings, 20th ACM Symposium on Theory of Computing, 1988," pp. 217–222.
35. R. Venkatesan and S. Rajagopalan, Average case intractability of Diophantine and matrix problems, *in* "Proceedings, 24th ACM Symposium on Theory of Computing, 1992," pp. 632–642.
36. J. Wang, Average case completeness of a word problem for groups, *in* "Proceedings, 27th ACM Symposium on Theory of Computing, 1995," pp. 325–334.
37. J. Wang and J. Belanger, On average P vs. average NP, *in* "Complexity Theory—Current Research" (K. Ambos-Spies, S. Homer, and U. Schöning, Eds.) pp. 47–67, Cambridge Univ. Press, Cambridge, 1993.
38. J. Wang and J. Belanger, On the NP-isomorphism problem with respect to random instances, *J. Comput. System Sci.* **50** (1995), 151–164.
39. R. E. Wilber, Randomness and the density of hard problems, *in* "Proceedings, 24th IEEE Symposium on Foundations of Computer Science, 1983," pp. 335–342.
40. H. Wilf, Some examples of combinatorial averaging, *Amer. Math. Monthly* **92** (1985), 250–261.
41. C. Wrathall, Complete sets and the polynomial-time hierarchy, *Theoret. Comput. Sci.* **3** (1977), 23–33.