CHAPTER 4

Nondeterminism and Randomized Computation

4.1. Nondeterminism

4.1.1. Nondeterministic Machines. One of the greatest insights of complexity theory is that of designing an algorithm that proceeds according to a distribution of behavior. This insight is, at first face, obviously impractical. Only on further reflection does it become so practical as to be the industry standard in many applications.

The most naive approach to this paradigm, perhaps, is to allow a machine to do almost anything, as long as we have an efficient algorithm to verify success. To be concrete, we give the following definition.

DEFINITION 4.1.1. Let $S \subseteq \{0,1\}^*$, and $f : \mathbb{R} \to \mathbb{R}$. We say that S is of class $\mathbf{NTIME}(f)$ if there is some Turing machine T such that on input $x \in \{0,1\}^*$, the machine T halts in at most $f(|x|)$ steps, returning output from $\{0,1\}$, with the additional provision that $x \in S$ if and only if there is some $\sigma \in 2^{<\omega}$ such that $T(x,\sigma) = 1$.

We call T the *verifier* for S, and we call σ the *proof* that $x \in S$. The idea is that $x \in S$ if and only if there is a proof of this fact, and the verifier T checks whether the proof is, in fact, a "correct" proof. Note, of course, that because of the bounded running time of the verifier, we have a natural bound of $f(|x|)$ on the length of σ, without needing to explicitly bound the length of the proof. Since the verifier can inspect at most one bit of σ at each time step, if there is no σ shorter than that on which $T(x,\sigma) = 1$, then there is none at all.

This definition is not transparently one of randomized computation, but of deterministic verification. Consequently, we also give the following definition.

DEFINITION 4.1.2. Let $S \subseteq \{0,1\}^*$ and $f : \mathbb{R} \to \mathbb{R}$. We say that S is of class $\mathbf{NTIME^{comp}}(f)$ if there is some oracle Turing machine T such that on input $x \in \{0,1\}^*$, the machine T halts in at most $f(|x|)$ steps, returning output from $\{0,1\}$, with the additional provision that $x \in S$ if and only if there is some $\sigma \in \{0,1\}^{<\omega}$ such that $T^\sigma(x) = 1$.

We will show that these two classes are equal. However, our provisional definition of $\mathbf{NTIME^{comp}}(f)$ displays the "random" character more clearly: The oracle is viewed as a set of coin flips, and the machine "accepts" or "rejects" x, possibly in light of the values of those coin flips. The condition for membership in $\mathbf{NTIME^{comp}}(f)$ is exactly that there is some sequence of coin flips on which the machine accepts. The randomness in this oracle is exaclty the "nondeterminism" of nondeterministic computation.

PROPOSITION 4.1.3. *The class* $\mathbf{NTIME^{comp}}(f)$ *is equal to the class* $\mathbf{NTIME}(f)$.

PROOF. Let S be of class $\mathbf{NTIME}(f)$, with verifier T. Then we define an oracle machine $\overline{T}$ so that $\overline{T}^{\sigma}(x) = T(x, \sigma)$ for all $x \in \{0,1\}^*$, with equal running time. Now $\overline{T}$ witnesses that S is of class $\mathbf{NTIME^{comp}}(f)$. The converse is exactly symmetric. □

Interesting as the questions of nondeterminism may be at the level of complexity (frequently runtime, occasionally space or some other resource), much of the distinction disappears entirely at the level of computability. Since the proof σ has bounded length, a Turing machine with adequately large time bounds could scan all possible proofs, and report whether it accepts any of them. As we will see in a later section, something of the distinction reemerges in more generalized models of computation.

4.1.2. NP and the Polynomial Hierarchy. The nondeterministic model of computation is fundamentally not one of probability, except in the most naive sense. Membership in the class is determined by the existence or nonexistence of a single element of the space of random coins — in the limit, a measure zero set.

On the other hand, this model does give rise to one of the most storied problems of complexity theory, and this problem is, as we shall see, closely entangled with problems on other models of computation which are more genuinely probabilistic. We state this problem now.

DEFINITION 4.1.4. Let $S \subseteq \{0,1\}^*$.

(1) We say that S is of class $\mathbf{P}$ if there is some Turing machine T and some polynomial p such that on input $x \in \{0,1\}^*$, the machine T halts in at most $p(|x|)$ steps, returning ouput from $\{0,1\}$, where $T(x) = 1$ if and only if $x \in S$.
(2) We say that S is of class $\mathbf{NP}$ if S is of class $\mathbf{NTIME}(p)$ for some polynomial p.

PROBLEM 4.1.5. Is $\mathbf{P} = \mathbf{NP}$?

The literature on this problem is extensive, but the problem is still so open that Gasarch [**192**] has polled computer scientists to find their opinions on not only the answer, but also whether and when a solution will be found (in fairness, a majority feel that a solution is possible and negative). Since the subject of nondeterministic computation is, in its own right outside the proper scope of the present book, we will not even attempt a cursory review of the literature, although the interested reader will do well to consult [**28**].

Just as the Continuum Hypothesis ($2^{\aleph_0} = \aleph_1$), when it resisted solution, gave rise to the much stronger Generalized Continuum Hypothesis ($2^{\aleph_\alpha} = \aleph_{\alpha+1}$), which at first seemed no more obviously true or false, the $\mathbf{P}$ versus $\mathbf{NP}$ problem has a more general analogue, the collapse of the Polynomial Hierarchy.

DEFINITION 4.1.6. We define the Polynomial Hierarchy, Σ_i^p, Π_i^p, and Δ_i^p, as follows:

(1) $\Sigma_1^p = \mathbf{NP}$
(2) A set S is of class Π_n^p if and only if its complement is of class Σ_n^p
(3) A set $S \subseteq \{0,1\}^*$ is of class Σ_{n+1}^p if and only if there is a Π_n^p set $Q(\tau, y)$ and for any $x \in \{0,1\}^*$ there is a $\sigma \in \{0,1\}^{q(|x|)}$ such that $x \in S$ if and only if $Q(\sigma, x)$.

(4) $\Delta_n^p = \Sigma_1^p \cap \Pi_1^p$.

By contrast with the arithmetical hierarchy, in which we know that every possible inclusion is strict, relatively little is known about the strictness of this hierarchy. It is not even obvious that $\Delta_1^p = \mathbf{P}$. There are, for each i, problems that are Σ_i^p complete. Moreover, we can give some criteria for collapse.

PROPOSITION 4.1.7. *If* $\mathbf{P} = \mathbf{NP}$*, then* $\Sigma_i^p = \Sigma_{i+1}^p$ *for all* i*.*

PROOF. We proceed by induction on i. For $i = 1$, since $\Sigma_1^p = \mathbf{NP}$, then under the stated hypothesis we have $\Sigma_1^p = \mathbf{P} = \Pi_1^p$. Consequently, $\Sigma_2^p = \mathbf{NP} = \Sigma_1^p$. Similarly, by induction, $\Sigma_{i+1}^p = \mathbf{NP} = \Sigma_i^p$. □

This is, *a priori*, not the only way in which the hierarchy could collapse. It is, for instance, possible that $\Sigma_i^p \subsetneq \Sigma_{i+1}^p$ exactly for $i < k$ for some k. In that case, we say that the Polynomial Hierarchy collapses at level k. The uncertainty concerning whether the hierarchy collapses at some point, and, if so, where, enhances the contrast of this structure with the arithmetical hierarchy — for instance, the arithmetical hierarchy can have no complete set precisely because the hierarchy is strict: if an "arithmetical hierarchy complete" set occurred in Σ_i^0, it could not be complete, as it would not admit reduction from a properly Σ_{i+1}^0 set. In the Polynomial Hierarchy, though, that could just be the level at which the hierarchy collapses.

Really, the better analogy is with the analytic hierarchy — although the same differences remain: the analytic hierarchy is known to be strict in a very strong sense, and little is known unconditionally about the structure of the polynomial hierarchy. In essense, the step to pass from Π_n^p to Σ_{n+1}^p is very much like a function quantifier, an issue to be explored more fully in the next section.

4.1.3. Descriptive Complexity. One of the most powerful insights of computability theory is the connection between degrees of unsolvability on one hand and logics on the other. In an appropriate context, the sets defined by Σ_1^0 formulas are precisely the computably enumerable ones, the sets defined by computable infinitary formulas are exactly those computable from (perhaps transfinitely) iterated Turing jumps, and those defined by Π_1^1 formulas exactly those computable from Kleene's $\mathcal{O}$.

This connection continues at the level of complexity, as well. We consider structures in the *language of strings*, which consists of one binary predicate that is axiomatically determined to be a linear ordering, and one unary predicate X, which will be interpreted as the set of bits with a 1. A finite structure in this language is thus interpreted as a binary string, of length equal to its cardinality.

The standard reference in this area is Immerman's survey [**249**], and the treatment in this section is indebted to his. The following theorem, first proved by Fagin in his 1973 Ph.D. thesis [**164**], is fundamental in the area.

THEOREM 4.1.8. *Let* $S \subseteq 2^{<\omega}$*. Then* S *is of class* **NP** *if and only if* S *is axiomatizable in the language of strings by a finitary* Σ_1^1 *formula.*

PROOF. Let T be an oracle Turing machine running in time n^k witnessing that $S \in \mathbf{NP}$. We can arrange (if necessary, by slowing T) that T uses exactly one random bit at each time step. Let $\Gamma = (\gamma_i : i \leq g)$ be a list of all pairs (q, σ), where q is either a state of T or a new symbol $-$; and σ is from the tape alphabet of T.

We let $\left(\mathfrak{C}_{s,t} : s,t \leq n^k\right)$ be an array in which $\mathfrak{C}_{s,t}$ is a pair $(q,\sigma) \in \Gamma$ so that σ is the contents of cell s at time t, and such that if T is on cell s at time t, then q is the state of the machine at that time, and otherwise $q = -$. For each $i \leq g$, we define a predicate $C_i(s,t)$ to denote the pairs s,t such that $\mathfrak{C}_{s,t} = \gamma_i$.

Let Δ be the string of random bits used by T in the first n^k steps. We then construct a formula as follows: We let $\alpha(\mathfrak{C},\tau)$ say that $\mathfrak{c}_{s,0}$ codes the input string τ, and let $\beta(\mathfrak{C})$ say that $C_i(s,t) \to \neg C_j(s,t)$ for $i \neq j$ (that is, that at each time, a particular cell may have only one value). We let $\xi(\mathfrak{C})$ state that $\left(\mathfrak{C}_{s,n^k} : s \leq n^k\right)$ includes the accept state. Finally, we write a sentence $\eta(\mathfrak{C},\Delta)$ stating that for all t, the stage $\mathfrak{C}_{s,t+1}$ is the next step prescribed by T after $\mathfrak{C}_{s,t}$ on random bit $\Delta(t)$. Now let

$$\varphi(\mathfrak{C},\Delta,\tau) = \alpha(\mathfrak{C},\tau) \wedge \beta(\mathfrak{C}) \wedge \xi(\mathfrak{C}) \wedge \eta(\mathfrak{C},\Delta).$$

Now $\tau \in S$ if and only if $\exists(\mathfrak{C},\Delta)\ \varphi(\mathfrak{C},\Delta,\tau)$.

On the other hand, any finitary Σ^1_1 formula axiomatizing S can be expressed as $\exists f \psi(f,\tau)$, where ψ has no second-order quantifiers. Now a Turing machine can verify in polynomial time that a particular sequence f and input τ satisfy $\psi(f,\tau)$, so that S is of class **NP**. □

This result generalizes throughout the polynomial hierarchy, as seems to have been first recorded by Stockmeyer [**427**].

THEOREM 4.1.9. *Let $S \subseteq 2^{<\omega}$. Then S is of class Σ^p_n (respectively, Π^p_n) if and only if S is axiomatizable in the language of strings by a finitary Σ^1_n (respectively, Π^1_n) formula.*

PROOF. Notice that Π^p_n sets are exactly the complements of Σ^p_n sets, just as Π^1_n sets are the complements of Σ^1_n sets. Consequently, it suffices to prove the result for Σ^p_n. We have already established the result for Σ^p_1.

Suppose that the Σ^p_n (respectively, Π^p_n) sets are exactly the Σ^1_n (respectively, Π^1_n) sets. Then the result $\Sigma^1_n \subseteq \Sigma^p_n$ follows exactly as in the original case. For the opposite inclusion, the inductive definition of the polynomial hierarchy admits a straightforward translation from a Σ^p_n definition to a Σ^1_n definition. □

4.2. Randomized Turing Machines

4.2.1. Complexity Classes Defined by Randomization. We now turn to a more properly probabilistic notion of computation. It is natural enough to ask that a computation give the correct answer with some high probability, but it is important to ask in what domain the probability lives. One can (and we do, in Section 4.5) carry out a satisfactory theory in which we consider a probability measure on the space of inputs to the computation, and consider the probability that a randomly selected input is one on which the computation is correct. The larger body of work, though, has been in a different direction.

Consider the Miller-Rabin primality test, first put forward in [**342, 374**]. To determine whether n is prime, we perform several tests. In each test, we choose a random positive integer $a < n$, and use it as a test for whether n is composite. To do this, we compute $x = a^{n-1} \bmod n$. If $x \neq 1$, then n cannot be prime by Fermat's Little Theorem. In the course of the calculation, we might also discover a non-trivial square root of 1 modulo n, in which case we again know that n is composite (a corollary to Fermat's Little Theorem). We perform these tests (with new random choices of a) as often as we like (in practice, there is a formula that will give the

number of tests necessary to give the desired level of certainty of correctness). If these checks reveal that n is composite, we report as much. Otherwise, we report it to be prime.

To those familiar with computably enumerable (or perhaps Π^0_1) sets, the issue here is clear: we are giving output of both positive (prime) and negative (composite) kinds, based only on negative information. Of course, if we report that n is composite, it is based on good evidence, and what we reported was correct. If we report that n is prime, it is possible that we just got unlucky with the choice of the a's in all of the tests. Under appropriate (and very strong) number-theoretic assumptions (a strengthening of the Riemann Hypothesis), such unluckiness is impossible, and the algorithm is simply correct for determining whether n is prime. Without venturing for the wild fancies of number theory quite this far, if n truly is composite, there are still quite a few of the possible values for a that must be witnesses, and the probability of not finding any of them in several attempts is quate small (indeed, the probability is 2^{-t}, where t is the number of tests performed; the interested reader may find a full exposition and analysis of the algorithm in [**124**]). While a polynomial-time algorithm for primality testing is now known, methods like the Miller-Rabin test are still the industrial standard for most applications, because their correctness is high enough for industrial needs, and they run faster.

The one part of the previous algorithm that is not obviously "algorithmic" is the clause, "choose a random positive integer $a < n$." Indeed, depending on what, exactly, we mean by it, this operation is not likely to be strictly computable. As we saw in Chapter 3, an important foundation of algorithmic randomness is that if we have an algorithm to compute something, it cannot be very random. Consequently, a broader definition is needed. The following definition, introduced in [**133**], formalizes the notion of a Turing machine that can make some random choices, and then produce an output that is correct with high probability in the domain of the random choices.

Definition 4.2.1. A *randomized Turing machine* (sometimes called a *probabilistic Turing machine*) is a Turing machine equipped with an oracle for an element of 2^ω, called the *random bits*.

We can then talk reasonably about a randomized Turing machine M having output k on input n with probability p if the Lebesgue probability measure of the set of random bit strings x such that $M^x(n) = k$ is equal to p.

The addition of the random bits presents an abundance of new questions. For instance, we classically ask about whether a Turing machine halts. We can now ask, for instance about the probability of halting. We consider a Turing functional $T : 2^{<\omega} \to 2^{<\omega}$, in the sense that T is a Turing machine which, on oracle σ will output the sequence $T(\sigma)$. We call such a functional *prefix-free* if, whenever $T(\sigma_1)\downarrow$ and $T(\sigma_2)\downarrow$, then σ_1 is not an initial segement of σ_2. A *universal* prefix-free machine is a Turing functional U such that for any prefix-free machine T there is a string $\nu_T \in 2^{<\omega}$ such that $T(\sigma) = U(\nu_T\sigma)$ for all $\sigma \in 2^{<\omega}$.

Following Chaitin [**101**], we define Ω_U, for a fixed universal prefix-free machine U, to be probability (in the sense of the standard probability measure on $2^{<\omega}$) of the set of σ such that $U(\sigma)\downarrow$. This is often described as the probability that a random (prefix-free) Turing machine halts on a random input. To see this interpretation, interpret U as a randomized Turing machine whose output, for a fixed oracle, is either undefined or constant (over all inputs). We then interpret the random bits

as giving both the U-code for an algorithm (in the first several bits) and the input (whose arity may be an arbitrary finite number). In this interpretation, all U-codes of the same length are given equal probability, and, given a U-code, all inputs (of the correct length) have equal probability.

THEOREM 4.2.2 (Chaitin [**101**]). *The probability Ω that a random prefix-free Turing machine will halt on a random input is a left-c.e. 1-random real.*

PROOF. To show that Ω is left-c.e. we take the computably enumerable set $H = \{\sigma | U(\sigma)\downarrow\}$. Since each element of U has finite length, we can, uniformly in n, add the probabilities $2^{|\sigma|}$ for all σ enumerated into H by time n. Since this sequence is monotonically increasing and approximates Ω from the left, the result follows.

To show that Ω is 1-random, we follow the proof of [**145**]. We first describe a particular prefix-free machine M in the following way. At stage s, we consider whether $U(\tau)$ threatens to be a name for Ω witnessing that $K(\Omega \upharpoonright s) < s - c$, where c is the coding constant for M in U. That is, we find n such that $\omega_n \upharpoonright s = \Omega \upharpoonright s$ (this is possible by the proof that Ω is left-c.e.), and search for a τ of length at most $s - c$ such that $U(\tau)_n = \omega_n \upharpoonright s$. If such a τ exists, we find λ outside the range of U_n, and set $M(\tau) = \mu$.

We now verify that this procedure prevents τ as a name for $\Omega \upharpoonright s$. We must have ν with $|\nu| \leq |\tau| + c < s$ such that $U(\nu) = M(\tau)$. However, $U_n(\sigma) \neq \mu$ for any σ, so that $U(\nu)$ does not halt within n steps, and so $\Omega \upharpoonright s \neq \omega_n \upharpoonright s$. Thus, $U(\tau) \neq \Omega \upharpoonright s$ for any τ shorter than $s - c$, as required. □

Several similar results have been achieved recently, in work of Barmpalias, Cenzer, and Porter [**42, 43**]. Of course, the exact numerical values in all of these results, including Chaitin's, depend dramatically on the choice of a universal machine. The following result from [**42**] is typical.

THEOREM 4.2.3 (Barmpalias–Cenzer–Porter). *The following conditions on a real number x are equivalent:*

1. *There is a universal oracle Turing machine U such that the probability that U produces a computable output when reading from a random oracle is x.*
2. *x satisfies both of the following conditions:*
   (a) *$x \in (0, 1)$,*
   (b) *$x = \alpha - \beta$, where α and β are $\emptyset'$-left-c.e. real numbers.*

PROOF. Assuming 1, condition 2a is trivial, so we establish 2b. We begin by showing that x is the difference of two $\emptyset'$-right-c.e. numbers. Let C be a Π^0_2 class. Then $\emptyset'$ can produce a decreasing sequence of rationals $(c_n : n \in \omega)$ converging to the measure of C, since, as non-members of C are enumerated, measure can be subtracted. Now given a Turing functional T, the class of all reals X such that $T(X)$ is total is a Π^0_2 class. If we can subtract from its measure the measure of the X such that $T(X)$ is total and not computable, we will have the intermediate result. To this end, consider the set I of all X such that $T(X)$ is total and $X \in \bigcup_{i \in \omega} V_i^{T(X)}$, where $(V_i : i \in \omega)$ is a universal Martin-Löf test. Let X be a 2-random real. We will show that $X \in I$ if and only if $T(X)$ is total and $T(X)$ is not computable.

Indeed, let $T(X)$ be total and non-computable. Then if $X \notin \bigcap_{i\in\omega} V_i^{T(X)}$, then X would be random relative to $T(X)$, but $T(X) \leq_T X$, and so $T(X)$ must be Δ^0_2. However, no 2-random X computes a non-computable Δ^0_2 set, so that $X \in I$. On the other hand, if $X \in I$ and $T(X)$ is 2-random, it must follow that $T(X)$ is not computable.

Now this set I is a Π^0_2 class (and thus has $\emptyset'$-right-c.e. measure), and has the same measures as the set of X such that $T(X)$ is total and not computable. Since being a difference of $\emptyset'$-left-c.e. reals is equivalent to being a difference of $\emptyset'$-right-c.e. reals, condition 2b follows.

The proof from 2 to 1 is more difficult. It makes use of the following well-known fact (see Section 5.1 of [**145**]), which will be useful in other results later.

LEMMA 4.2.4. *The following are equivalent:*

(1) *The real number α is X-left-c.e.*
(2) *There is a prefix-free machine M such that α is the measure of the domain of M^X.*
(3) *There is a $\Sigma^0_1(X)$-class of measure α.*

Moreover, these equivalences hold uniformly.

PROOF. If α is X-left-c.e., we can express α as an infinite sum $\sum_{i\in\omega} 2^{-n_i}$ where $(n_i : i \in \omega)$ is a computable sequence. By the Kraft-Chaitin theorem, suitably relativized, we have a machine M whose domain has measure α. The domain of this machine will then be a $\Sigma^0_1(X)$-class of appropriate measure. Finally, X can enumerate the lower cut of the measure of a $\Sigma^0_1(X)$-class. □

Let α, β be $\emptyset'$-left-c.e. reals and let γ be a 2-random $\emptyset'$-left-c.e. real. By a result of Demuth [**137**], the numbers $\alpha_0 = \alpha + \gamma$ and $\beta_0 = \beta + \gamma$ are 2-random $\emptyset'$-left-c.e. reals, which, of course, have the same difference as α and β. So we can assume that α, β are 2-random.

If $\alpha - \beta$ is also 2-random, then by a result of Rettinger and Zheng [**467**], we know that $\alpha - \beta$ must be either $\emptyset'$-left-c.e. or $\emptyset'$-right-c.e. In the case that it is $\emptyset'$-left-c.e., we take a universal machine V_0 such that $P_X\left(V_0(X) \equiv_T \emptyset\right) = \alpha_0 - \beta_0$, where α_0, β_0 are $\emptyset'$-left-c.e. reals. Now by a result of [**294**] there must be some i such that $(\alpha - \beta) - 2^{-i}\alpha_0$ is $\emptyset'$-left-c.e. and $(\alpha - \beta) - 2^{-i}(\alpha_0 - \beta_0) \in \left(0, 1 - 2^{-i}\right)$. From these two points, if follows that $(\alpha - \beta) - 2^{-i}(\alpha_0 - \beta_0)$ is $\emptyset'$-left-c.e. We can then find a machine M_0 such that $M_0(\sigma)$ is undefined for σ comparable with 0^i and where $P_X\left(M(X) \equiv_T \emptyset\right) = (\alpha - \beta) - 2^{-i}(\alpha_0 - \beta_0)$. We then define a machine V so that

$$V(\rho) = \begin{cases} V_0(\sigma) & \text{if } \rho = 0^i\sigma \\ M_0(\rho) & \text{otherwise} \end{cases}$$

Now $P_X\left(V(X) \equiv_T \emptyset\right) = \alpha - \beta$.

If $\alpha - \beta$ is 2-random and $\emptyset'$-right-c.e., then we let δ, e be such that

$$(\alpha - \beta) - 2^{-e}\delta \in \left(2^{-e}, 1\right)$$

and such that δ is a $\emptyset'$-left-c.e. real. Then, as before, we can find a machine V such that $P_X\left(V(X) \equiv_T \emptyset\right) = \delta$.

Now we can find a Σ^0_2 prefix-free set S of strings incomparable with 0^e and such that $\mu[\![S]\!] = 1 - ((\alpha - \beta - 2^{-e}\delta)$. Consequently, we can build a machine N

such that $P_X(N(X) \equiv_T \emptyset) = (\alpha - \beta) - 2^{-e}\delta$. We deine a machine M so that

$$M(\rho) = \begin{cases} N(\rho) & \text{if } \rho \perp 0^e \\ V(\sigma) & \text{if } \rho = 0^e\sigma \end{cases}$$

Now $P_X(M(X) \equiv_T \emptyset) = P_X(N(X) \equiv_T \emptyset) + 2^{-e}P_X(V(X) \equiv_T \emptyset) = \alpha - \beta$.

In the case that $\alpha - \beta$ is not 2-random, we pick a 2-random $\emptyset'$-left-c.e. δ. By a result of [**44**], it follows (for any e that $(\alpha - \beta) - 2^{-e}\delta$ must be $\emptyset'$-right-c.e. We then pick e such that $(\alpha - \beta) - 2^{-e}\delta \in (2^{-e}, 1)$, and pick N, V, and M as in the previous case. □

The question already arises in the original paper [**133**] whether randomized Turing machines represent any genuine gain in power over deterministic Turing machines. Of course, something of that question depends on what we expect the machine to do. The authors of that paper prove that there is a set which is not computably enumerable, but which is enumerated with positive probability by an object much like a randomized Turing machine, although their formalism is not quite equivalent and this result fails under the definitions given above. Modern approaches to the question have asked either more or less. On the one hand, we might ask that the algorithm be correct with probability bounded above $\frac{1}{2}$, something like the Miller-Rabin algorithm. In that case, we could simulate the algorithm on a (much slower) deterministic machine by the following algorithm: To compute $f(n)$, list sequences in $2^{<\omega}$, and run the randomized machine with those sequences as random bits. When the machine gives the same value on a set of such strings that collectively have measure more than $\frac{1}{2}$, that must be the correct answer, so our deterministic machine reports it.

Of course, in the contexts where the Miller-Rabin algorithm is favored over the AKS class of deterministic primality tests on the basis of being slightly faster but not quite as certain, this method of massive simulation is not practical. Consequently, a large literature now centers on this second path of setting some correctness standard for a randomized Turing machine, and asking whether the same performance could be matched by a deterministic machine, *under given time constraints*. This gives rise the the following definitions.

DEFINITION 4.2.5 (Gill [**198**]). Let $S \subseteq \{0,1\}^*$.

(1) We say that S is of class **BPP** if and only if there is some randomized Turing machine T and some polynomial p such that on input $x \in \{0,1\}^*$, the machine T halts in at most $p(|x|)$ steps, returning ouput from $\{0,1\}$, where $T(x) = \chi_S(x)$ with probability at least $\frac{2}{3}$.
(2) We say that S is of class **RP** if and only if there is some randomized Turing machine T and some polynomial p such that on input $x \in \{0,1\}^*$, the machine T halts in at most $p(|x|)$ steps, returning ouput from $\{0,1\}$, where if $x \in S$, we have $T(x) = 1$ with probability at least $\frac{2}{3}$, and if $x \notin S$, then $T(x) = 1$ with probability 0.
(3) We say that S is of class **coRP** if and only if the complement of S is of class **RP**.
(4) We say that S is of class **ZPP** if and only if there is some randomized Turing machine T and some polynomial p such that on input $x \in \{0,1\}^*$, the machine T halts in $p(|x|)$ steps *in expectation*, returning ouput from $\{0,1\}$, where $T(x) = \chi_S(x)$.

As in many parts of complexity theory, much is unknown about how these classes relate to one another, to **P**, to **NP**, and to the enormous collection of other classes that have been defined. The following standard result, though, expemplifies the connectedness of these classes and their tractability (at least, relative to the rest of complexity theory).

PROPOSITION 4.2.6. $\mathbf{ZPP} = \mathbf{RP} \cap \mathbf{coRP}$.

PROOF. Suppose T_1 witnesses that $S \in \mathbf{RP}$, and that T_2 witnesses that $S \in \mathbf{RP}$. Then we can simultaneously run $T_1(x)$ and $T_2(x)$, in total time deterministically equal to the sum of two polynomials in $|x|$. If T_1 and T_2 agree, then they must be correct. If not, we repeat the procedure. Since they agree with probability at least $\frac{2}{3}$ on each repetition, we expect agreement on the first trial, so that the full procedure runs in expected polynomial time.

On the other hand, if T witnesses that $S \in \mathbf{ZPP}$, with polynomial p bounding the expected run time of T, we run T for $3p(|x|)$ steps, and then return the output of T, if it halts in that time, and 1 otherwise.

LEMMA 4.2.7 (Markov-Chebyshev Inequality; see p. 47 if [**409**]). *Let X be a non-negative random variable with expected value $E(X)$. Then $P(X \geq \epsilon) \leq \frac{E(X)}{\epsilon}$.*

Since the run-time is non-negative, we apply the Markov-Chebyshev inequality to see that the probability that the actual run time exceeds three times its expectation is at most $\frac{1}{3}$. Consequently, this algorithm witnesses that S is in **RP**. Similarly, if we output 0 when T does not halt in the prescribed time, we see that S is in **coRP**. □

The second half of the preceding proof illustrates an important method in probability: the use of so-called "concentration inequalities." These results give (often quite loose) bounds on the probability that a random variable takes values in some range. Frequently, the conditions for the inequality to hold are broad enough that, by appropriate choice of parameters, these loose bounds can be made good enough to prove something useful. We will see more examples, especially in Chapter 6, but the technique is worth noting in this proof. The Markov-Chebyshev inequality is an important and simple example of a concentration inequality. It has as consequenses several other concentration inequalities of interest. Proofs of these statements can be found in [**409**].

COROLLARY 4.2.8. *Let X be a random variable and $\epsilon > 0$. Then the following hold:*

(1) $P(|X| \geq \epsilon) \leq \frac{E(X)}{\epsilon}$
(2) $P(|X| \geq \epsilon) \leq \frac{E\left(X^2\right)}{\epsilon^2}$
(3) $P(|X - E(X)| \geq \epsilon) \leq \frac{var(X)}{\epsilon^2}$

The centrality of the concentration inequality method in the proof of Proposition 4.2.6 may explain the often-cited contrast between the case of randomized computation (where we know that $\mathbf{ZPP} = \mathbf{RP} \cap \mathbf{coRP}$) and the case of nondeterministic computation (where we emphatically do not know whether $\mathbf{P} = \mathbf{NP} \cap \mathbf{coNP}$). While we know at least something about the spread of even an arbitrary random variable, no such structure exists on mere path existence, as is called for in nondeterministic computation.

While work on the descriptive complexity of **BPP** is ongoing, it is worthwhile to note that progress has been made. For instance, Eickmeyer and Grohe [**154**] have reported a logic that captures **BPP**.

4.2.2. Effective Completeness for Continuous First Order Logic. The classical effective completeness theorem is foundational for computable model theory. Let T be a complete first-order theory such that there is a Turing machine to decide, for each first-order sentence of the signature, whether that sentence is or is not in T (we call such a theory *decidable*). Then there is a model $\mathcal{M} \models T$ such that the atomic diagram of $\mathcal{M}$ is classically computable as a set of Gödel codes. This is, in effect, due to the constructive character of Henkin's proof of classical completeness.

Of course, this is no guarantee that an arbitrary model of T will be computable. Indeed, unless T is automorphically trivial any computable model is isomorphic to a non-computable model. However, even when we weaken the question to which models have isomorphic copies which are computable, there are typically many which do not, and a large literature exists on the question of which models of a given decidable theory have, or do not have, computable copies in this sense.

The converse to effective completeness is false. The standard model of arithmetic, $(\mathbb{N}, +, \cdot, 0, 1)$, for instance, is computable, but is a model of true arithmetic, a highly undecidable theory. Moreover, effective completeness is far from the only way to guarantee the existence of a computable structure. Construction of a model by Barwise compactness, or even by direct construction are widespread in the literature.

Nevertheless, the effective completeness theorem provides an important connection between effectiveness of the theory and effectiveness of at least some models. In the present section, we will show that a similar result relates continuous first order logic and randomized computation. The results of this section were originally proved in [**88**].

The first consideration in a randomized effective completeness theorem is to describe the kind of object that will be said to be effective. On the side of the theory, this is straightforward. As in the first order case, we define a complete theory to be one that is the theory of some model — in the continuous case, if $T = Th(\mathcal{M})$ is the set of all continuous $\mathcal{L}$-sentences φ such that $\mathcal{M}(\varphi) = 0$. We now define what it means for a continuous theory to be decidable.

Definition 4.2.9 ([**60**]). Let L be a continuous signature and Γ a set of formulas of L.

(1) We define
$$\varphi_\Gamma^\circ := \sup\{\mathcal{M}(\varphi) : \mathcal{M} \models \Gamma\}.$$
(2) If T is a complete continuous first-order theory, we say that T is *decidable* if and only if there is a (classically) computable function f, called a *decision procedure for* T such that $f(\varphi)$ is an index for a computable real number equal to φ_T°.

In effect, this means that we can, uniformly in φ, compute the truth value of φ required by T of all its models. It is worth remembering that if φ is a sentence, then $1 \dot{-} \varphi$ is also a sentence, so that f will also compute the truth value of $1 \dot{-} \varphi$ required by T. Since a particular structure will have a single truth value for φ, a complete theory T will include information on both, so that the value of φ is

uniquely determined. In a decidable theory, it is uniquely determined to be a real number which is computable, uniformly in φ.

We now turn our attention to the models. As in the classical case, a model is identified with its atomic diagram, but since constants are not continuous in the most interesting continuous structures, some modification is required. We replace each true constants of classical first-order logic with a unary predicate for the distance from the constant's interpretation. In this way, we arrive at the following definition.

DEFINITION 4.2.10. Let L be a computable continuous signature (i.e. one where the sets of predicate and function symbols, the set of moduli, and the arity function are all computable). Let $\mathcal{M}$ be a continuous L-structure. Let $L(\mathcal{M})$ be the expansion of $\mathcal{L}$ by a constant c_m for each m in the universe of $\mathcal{M}$ (i.e. a unary predicate $c_m \in \mathcal{R}$ where $c_m^{\mathcal{M}}(x) := d(x, m)$). Then

(1) The *continuous atomic diagram* of $\mathcal{M}$, written $D(\mathcal{M})$ is the set of all pairs (φ, p), where φ is a quantifier-free (i.e. sup- and inf-free) sentence in $L(\mathcal{M})$ and $\mathcal{M}(\varphi) = p$.
(2) The *continuous diagram* of $\mathcal{M}$, denoted $D^*(\mathcal{M})$, is the set of all pairs (φ, p), where φ is a sentence in $L(\mathcal{M})$ and $\mathcal{M}(\varphi) = p$.

Equipped with this definition, we define what it means for a structure to be effective in the sense of randomized computation.

DEFINITION 4.2.11. Let $\mathcal{M}$ be a continuous structure.

(1) We say that $\mathcal{M}$ is *probabilistically computable* if and only if there is some randomized Turing machine T such that, for every pair $(\varphi, p) \in D(\mathcal{M})$, the machine T accepts φ with probability p.
(2) We say that $\mathcal{M}$ is probabilistically decidable if and only if there is some randomized Turing machine T such that for every pair $(\varphi, p) \in D^*(\mathcal{M})$, the machine T accepts φ with probability p.

We now set about to prove the effective completeness theorem for randomized computation and continuous first order logic. Much of the work of constructing the model is carried out in [**60**]; only some effectivization was necessary to complete the following theorem in [**88**].

THEOREM 4.2.12. *Let T be a complete decidable continuous first-order theory. Then there is a probabilistically decidable continuous structure $\mathcal{M} \models T$.*

PROOF. We first extend T to a consistent set Γ of formulas which is Henkin complete; that is, a formula in which for every formula φ, every variable x, and every pair $p < q$ of diadic rationals, there is a constant c such that

$$\left(\sup_x \varphi \dot{-} q\right) \wedge (p \dot{-} \varphi[c/x]) \in \Gamma.$$

This notion of Henkin completeness was used in [**60**]. We let $L_0 = L$ and $\Gamma_0 = T$. At stage n, we add, for each formula $\varphi \in L_n$ a new constant $c_{(\varphi,x,p,q)}$ to form L_{n+1}. We form Γ_{n+1} by adding the formula

$$\left(\sup_x \varphi \dot{-} q\right) \wedge \left(p \dot{-} \varphi[c_{(\varphi,x,p,q)}/x]\right).$$

We observe that this construction of $\Gamma = \bigcup_i \Gamma_i$ is effective uniformly in L and T.

We can also — still uniformly in L, T, and s, compute a sequence Δ_s of sets such that $\Delta^0 := \bigcup_{s\in\omega} \Delta_s$ such that $\Gamma \subseteq \Delta^0$ and such that for any $\varphi, \psi \in L^* = \bigcup_{n\in\omega} L_n$, we have either $\varphi \dot{-} \psi \in \Delta^0$ or $\psi \dot{-} \varphi \in \Delta^0$. Indeed, we start with $\Delta_0 = \Gamma_0$, and at stage s, we take the first φ, ψ requiring attention, and check (effectively, since the theory is decidable) whether Δ_s proves that $\psi \dot{-} \varphi$ has value less than $\frac{1}{2^n}$. Accordingly, we add either $(\psi \dot{-} \varphi) \dot{-} \frac{1}{2^n}$ or $(\varphi \dot{-} \psi) \dot{-} \frac{1}{2^n}$ to form Δ_{s+1}.

As an intermediate step, we construct two machines: an accepting machine M_A and a rejecting machine M_R. At stage s, if $\Delta_s \vdash \varphi \dot{-} \frac{k}{2^n}$, we will enumerate $1 - \frac{k}{2^n}$ into the left cut of $\alpha_{\varphi,A}$. Similarly, if $\Delta_s \vdash \frac{k}{n} \dot{-} \varphi$, we enumerate $\frac{k}{2^n}$ into the left cut of $\alpha_{\varphi,R}$.

Now, using the uniformity of Lemma 4.2.4, we produce a single machine M_A such that $M_A(\varphi)$ halts with probability $\alpha_{\varphi,A}$ and a single machine M_R such that $M_R(\varphi)$ halts with probability $\alpha_{\varphi,R}$, so that the domains of $M_A(\varphi)$ and $M_R(\varphi)$ are disjoint. We now define our final machine M by running both M_A and M_R on φ, accepting if M_A halts and rejecting if M_R halts. Now by construction of Δ_s, the machine M accepts φ with probability exactly $\mathcal{M}(\varphi)$. □

Complexity is a notoriously fraught area for computable model theory, because of sensitivity to representation of the elements of an infinite structure — indeed, a choice of whether a natural number is represented by tallies or by binary representation can make a difference of exponential time. Still, it is worthwhile to note the following connection to the complexity-theoretic aspects of randomized computation, proved in [**88**].

PROPOSITION 4.2.13. *We say that a continuous structure is polynomial time if and only if there is some polynomial-time randomized Turing machine that, for any $(\varphi, p) \in D(\mathcal{M})$, accepts φ with probability p.*

1. *Let $A \in$ **BPP**. Then there is a polynomial-time probabilistically computable structure $\mathcal{M}$, with a polynomial-time function $f : \{0,1\}^* \to \mathcal{M}$ such that there is a quantifier-free formula $\varphi(x)$ such that $\mathcal{M}(\varphi(x)) \leq \frac{1}{3}$ for $x \in f(A)$, and $\mathcal{M}(\varphi(x)) \geq \frac{2}{3}$ for $x \notin f(A)$.*
2. *Let $\mathcal{M}$ be a polynomial time probabilistically computable structure, and let A, B be quantifier-free definable subsets of $\mathcal{M}^n$, whose distance is bounded away from zero, and where $A \cup B$ is polynomial-time computable. Then each of A and B is of class **BPP**.*

PROOF. For the first point, we make a structure on the universe of which A is naturally a subset, with the discrete metric, and a unary predicate $\mathcal{A}(x)$, interpreted as the probability that x is accepted by the machine that witnesses $A \in$ **BPP**. For the second point, for any $\bar{a} \in A \cup B$, we can decide whether $\bar{a} \in A$ by evaluating $\mathcal{M}(\varphi(\bar{a}) \dot{-} \psi(\bar{a}))$. □

4.2.3. Pseudorandom Generators and Derandomization. Consider a set S of class **BPP**. The obvious quick way to check whether $n \in S$ is to provide some random bits and see what the randomized algorithm does on them. Since the algorithm is required to be right a great majority of the time, we could take a majority vote of the results.

Of course, this doesn't work quite so easily. In particular, the entirety of Chapter 3 explores the many ways in which it impossible for us to simply "provide some random bits." An interesting hypothesis, which is the subject of intense

current interest in complexity theory, is that it may not be necessary to be random. It may only be necessary to be so much like random that the algorithm can't tell that it isn't random.

Obviously the literature in this area is large and fast-moving. Knuth [**286**] described many particular algorightms for generation of numbers that were "random enough" for certain applications, and the use of a random variable with one distribution to "simulate" (to generate a random variable with) another distribution is well-established in probability (see [**384**]). However, the literature on general-purpose pseudoranom generators of the type we describe here seems to have started with a paper of Yao [**465**]. The interested reader will find a much more extensive survey in [**205**] and [**28**], including most of the material of the present section.

Many of the results in this area are — perhaps necessarily (see Theorem 4.2.16) expressed in the language of Boolean circuits. In this model a k-ary Boolean circuit of size m takes k boolean inputs, and applies binary AND and OR, and unary NOT operations to compute a function, with m total operations performed.

DEFINITION 4.2.14. Let $\ell : \mathbb{N} \to \mathbb{N}$ be monotonically increasing. A *pseudorandom generator of stretch* ℓ is a deterministic Turing machine G such that for andy $\sigma \in \{0,1\}^k$,

(1) The machine G runs in time polynomial in $2^k \ell(k)$ steps and outputs a string of length $\ell(k)$, and
(2) If U_j is a uniform random binary string of length j, then for every Boolean circuit D_k of size $\ell(k)^2$, the circuit accepts $G(U_k)$ and $U_{\ell(k)}$ with probabilities that differ by less than $\frac{1}{6}$.

Something should be said about the formulation of this definition. In particular, the use of Boolean circuits in item 2 is striking. Generally, Boolan circuits are a non-uniform model of computation — the circuits on different numbers of inputs are not, *a priori*, required to have anything to do with one another, which allows such counterintuitive results as the fact that every set of natural numbers in unary representation is computable by a system of relatively small circuits. The point in the present application, though, is that nothing that a randomized Turing machine might want to do with its random bits should depend so sensitively on the difference between $G(U_k)$ and $U_{\ell(k)}$ that the outcome of a majority vote could be changed.

It also needs to be recognized that the *existence* of pseudorandom generators is already deeply problematic. As with much of complexity theory, so much is unknown that one can find entire books developing the consequenses of and conditions for the existence of objects whose existence itself is unknown. However, this custom is at odds with the usual practice of mathematics, and so it needs to be noted. The payoff is that many complexity classes based on randomization collapse under the assumption that certain pseudorandom generators exist.

Of course, even if a pseudorandom generator exists, it only reduces the number of random bits needed. If the randomized algorithm needs $\ell(k)$ random bits, we must still come up with k of them. Another obstacle still to be overcome is that the definition only calls for the pseudorandom generator to run in time exponential in its input. It is not obvious that an exponential time algorithm should require anything built from it to run in polynomial time. Nevertheless, the following theorem shows that the definition given is enough.

THEOREM 4.2.15. *If for some $\epsilon > 0$ there is a pseudorandom generator of stretch $k \mapsto 2^{\epsilon k}$, then* $\mathbf{BPP} = \mathbf{P}$.

PROOF. Let G be a pseudorandom generaotr of stretch $k \mapsto 2^{\epsilon k}$. Let T be a randomized Turing machine witnessing that $S \in \mathbf{BPP}$. We define a deterministic Turing machine U as follows. On input n, suppose that T runs in s steps. let k be the least natural number such that $2^{\epsilon k} \geq s$. We let U search over all binary strings σ of length k, computing, for each one, $T^{G(\sigma)}(n)$, and output the majority result. Note that the computation $\sigma \mapsto T^{G(\sigma)}(n)$ can be represented by a circuit of size $2^{2\epsilon k}$, for each n, this transformation must accept with the same probability (plus or minus error of less than $\frac{1}{6}$) as the transformation $\tau \to T^{\tau}(n)$, where τ ranges over all strings of length $2^{\epsilon k}$. Since T was correct on set of $\{0,1\}^{2^{\epsilon k}}$ with probability greater than $\frac{2}{3}$, and since the error introduced by G was less than $\frac{1}{6}$, the majority vote still gives the correct solution. Since U searches over strings of length k, it must compute G a total of 2^k times, each of which takes time polynomial in $2^{k+\epsilon k}$. However, k is approximately $\log s$, where s is polynomial in the size of n, so that U runs in polynomial time. □

The same strategy applies much more broadly, allowing many complexity classes to be collapsed if a pseudorandom generator of the right stretch exists.

As has been pointed out, the existence of pseudorandom generators is problematic. There are several well-known results giving other difficult conditions necessary or sufficient to the existence of pseudorandom generators with particular stretch. An important example is a result of Nisan and Wigderson [**355**].

THEOREM 4.2.16. *Let $\ell : \mathbb{N} \to \mathbb{N}$ be monotonically increasing, and $\ell(n) \leq 2^n$ for all n. Then the following are equivalent:*

1. *There is some $c > 0$ such that* **EXPTIME** *cannot be approximated by Boolean circuits of size $\ell(m^c)$.*
2. *There is some $c > 0$ such that there is a pseudorandom generator of stretch ℓ.*

While the existence of pseudorandom generators sufficient to collapse **BPP** to **P** is currently unknown, there are objects called expander graphs, unconditionally known to exist, that give some reduction in the length of the string of random bits needed by certain randomized Turing machines. These graphs arise naturally in the study of random graphs, and will be seen in Section 5.1.1.

4.3. Interactive Proofs

4.3.1. Interactive Proofs as Games. The complexity classes arising from randomized computation can be viewed as a modification of those arising from nondeterminism: Instead of requiring that there exist one path, we require that there exist a lot of paths. Another modification in a similar spirit views **NP** as a proof environment. We have already hinted at this view in the description of the "proof" and the "verifier" in comments after the definition of **NTIME**.

In this view, an agent (the "prover") using a lot of computing power finds a proof, if one exists, and then communicates it to the verifier, who is required to check it in polynomial time. We can imagine an environment in which more complex communication takes place between the prover and the verifier. This is the idea

behind interactive proofs, introduced in slightly different forms independently by [**207, 37**].

DEFINITION 4.3.1. Let $S \subseteq \{0,1\}^*$. We say that S is of class **IP** if and only if there is a randomized polynomial-time Turing machine V and a natural number constant k with the following properties:

(1) If $\sigma \in S$, there is a function $P : \{0,1\}^* \to \{0,1\}^*$ so that the sequence $V(\sigma), P(\sigma, V(\sigma)), V(\sigma, V(\sigma), P(\sigma, V(\sigma))), \ldots$ achieves a 1 in the kth term with probability at least $\frac{2}{3}$.
(2) If $\sigma \notin S$, then for any function $P : \{0,1\}^* \to \{0,1\}^*$, the sequence described in part 1 achieves a 1 in the kth term with probability at most $\frac{1}{3}$.

The idea is that if $\sigma \in S$, there should be some prover that can prove this to the verifier's general satisfaction, but if $\sigma \notin S$, no prover should reliably convince the verifier otherwise.

Some examples will illustrate the significance of the definition. Recall that the isomorphism on finite graphs is known to be of class **NP**, since it is easy to check whether a given function is an isomorphism.

PROPOSITION 4.3.2 (Goldreich–Micali–Wigderson 1991 [**206**]). *The isomorphism problem for finite graphs is of class* **IP**.

PROOF. A straightforward proof arises from the fact that the isomorphism problem for finite graphs is **NP**. To achieve this, the prover need only find an isomorphism and send it to the verifier. The verifier then checks (in polynomial time) that the function given is, in fact, an isomorphism.

Another proof system is possible, though, which has the additional interesting property that the prover does not have to reveal any information to the verifier beyond the fact that the two graphs are isomorphic (a so-called *zero knowledge proof*). On input (G_0, G_1), the prover selects uniformly at random a permutation π of the finite set of vertices of G_0, and interprets this as a graph $H \cong G_0$, and sends H to the verifier. The verifier chooses $\alpha \in \{0,1\}$ uniformly at random, and sends α to the prover. If $\alpha = 0$, the prover sends π, and if $\alpha = 1$, the prover sends $\pi \circ f$, where f is an isomorphism from G_0 to G_1, if any such isomorphism exists. The verifier checks whether the function sent is an isomorphism. If so, the procedure is repeated a second time, accepting if the second function is also an isomorphism. If either function fails to be an isomorphism, it rejects.

If $G_0 \cong G_1$, then the prover will always be able to send the verifier an isomorphism. Otherwise, the verifier will reject with probability at least $\frac{3}{4}$, since with that probability it will have asked about isomorphism of H with G_1 at least once. □

Graph non-isomorphism is not known to be in **NP**. In that sense, the following result was initially viewed as surprising.

PROPOSITION 4.3.3 (Goldreich–Micali–Wigderson 1991 [**206**]). *The non-isomorphism problem for finite graphs is of class* **IP**.

PROOF. On input (G_0, G_1), the verifier chooses α_0, α_1 uniformly at random, and selects uniformly at random a permutation π_i of G_{α_i}, interpreting it as a graph $H_i \cong G_{\alpha_i}$, and sends (H_0, H_1) to the prover. The prover computes, for each

i, whether $H_i \cong G_0$ or $H_i \cong G_1$, and sends the verifier (β_0, β_1), where $H_i \cong G_{\beta_i}$. The verifier accepts if $\beta_i = \alpha_i$ for each i, and rejects otherwise.

If G_0 and G_1 are not isomorphic, then each H_i is isomorphic to exactly one of G_0, G_1, and the prover can distinguish the two cases, so that $\vec{\beta} = \vec{\alpha}$ and the verifier accepts. Otherwise, the probability that both $\alpha_0 = \beta_0$ and $\alpha_1 = \beta_1$ is at most $\frac{1}{4}$. □

Although much is unknown about the complexity of graph isomorphism and non-isomorphism, these two results suggest, at least, that **IP** is a very large class. In 1990, it was shown that **IP** contains the polynomial hierarchy, and shortly afterward that it is equal to **PSPACE**, the class of problems decidable in arbitrary time using a polynomially bounded number of cells on the tape of the Turing machine.

PROPOSITION 4.3.4 ([**313**]). **IP** *contains the polynomial hierarchy.*

PROOF. The permanent $per(A)$ of an $r \times r$ matrix A is the sum

$$\sum_{\sigma \in S_r} \left(\prod_{i=1}^{r} a_{i\sigma(i)} \right).$$

Equivalently,

$$per(A) = \sum_{i=1}^{r} per\left(A_{\{1i\}}\right),$$

where the matrix minor $A_{\{ij\}}$ is the result of deleting the ith row and jth column of A. There is a nondeterministic machine M such that for any 0-1 matrix A, the computation $M(A)$ has $per(A)$ accepting paths (this follows from the recursive definition just given); that is, the permanent function is of class $\#\mathbf{P}$.

Moreover, the permanent function is known to be complete in this class [**449**]. Consequently, the set

$$L = \{(A, s) : A \text{ is a 0-1 matrix and } per(A) = s\}$$

is complete among problems solvable in polynomial time with an oracle for solving problems in $\#\mathbf{P}$. Since this class contains **PH**, by a result of Toda [**445**], it suffices to show that L has an interactive proof system.

To prove that $(A, s) \in L$, where A is an $r \times r$ matrix with entries from $\{0, 1\}$, we use the following procedure. The prover picks a prime p such that $r! < p < 2^r$, and sends both p and a proof that p is prime to the verifier. All later arithmetic in the procedure is done modulo p. Set $\mathcal{L}_0 = \{(A, s)\}$.

In stage t, the verifier checks whether $\mathcal{L}_t$ is a singleton (B, q) where B is a 1×1 matrix. If so, the verifier will accept if $q = per(B)$, and reject otherwise.

If $\mathcal{L}_t$ is a singleton with a larger matrix, the verifier will construct the minors $B_{\{1,i\}}$ for each i, and send them to the prover. The prover returns the permanent $q_i = per\left(B_{\{1,i\}}\right)$. The verifier then computes

$$\sum_{i=1}^{r} b_{1i} q_i.$$

If this quantity is not equal to q, the verifier rejects. Otherwise,

$$\mathcal{L}_{t+1} = \{(B_{\{l,i\}}, q_i) : 1 \leq i \leq dim(B)\}.$$

If $\mathcal{L}_t$ is not a singleton, then the verifier chooses the first two pairs in $\mathcal{L}_t$, say (B_1, q_1) and (B_2, q_2) and sends them to the prover. Note that at any time t, all matrices in $\mathcal{L}_t$ have the same dimension. The prover computes

$$f(x) = per\left(C + x(D - C)\right),$$

a polynomial of degree at most equal to the dimension of C and D, and sends $f_0, \ldots f_k$ to the verifier, where

$$f(x) = \sum_{i=0}^{k} f_i x^i.$$

The verifier then evaluates this polynomial on 0 and 1, and checks whether $f(0) = c$ and $f(1) = d$. If either is not equal, the verifier rejects. Otherwise, the verifier sets

$$\mathcal{L}_{t+1} = \left(\mathcal{L}_t - \{(C, c), (D, d)\}\right) \cup \{\left((C + a(D - C)\right), g(a))\}$$

for some a chosen uniformly at random from $\mathbf{Z}_p$.

The verifier will certainly either accept or reject by the time t at which $\mathcal{L}_t$ consists of a singleton whose matrix is 1×1. This will happen in at most $2(r - 1) + (r - 2) + \cdots + 1$ steps. If $per(A) = s$, then the verifier will accept (A, s).

We now consider the case in which $per(A) \neq s$, and show that the probability of accepting is low. Notice that the verifier will eventually accept if and only if at some stage t every pair $(B, q) \in \mathcal{L}_t$ has the property that $per(B) = q$. This cannot attain in either of the cases where $\mathcal{L}_{t-1}$ is a singleton. We assume, then, that we have a stage t in which there is some $(B, q) \in \mathcal{L}_t$ such that $per(B) \neq q$, but there is no such $(B, q) \in \mathcal{L}_{t+1}$. What must be shown is that the step which replaces two entries of $\mathcal{L}_t$ by one will, with high probability, preserve the property that $per(B) \neq q$ for some element of $\mathcal{L}$.

Let $(C, c), (D, d) \in \mathcal{L}_t$, with either $per(C) \neq c$ or $per(D) \neq d$. Note that if C, D are of dimension $r \times r$, the polynomial $f(x) = per(C + x(D - C))$ over $\mathbb{Z}_p$ is of degree at most r. At $x = 0$ this polynomial should compute the permanent of C, and at 1 it should compute the permanent of D. Suppose that the prover has sent a degree r polynomial g different from f. If the verifier does not reject, $g(0) = c$ and $g(1) = d$, so one of them does not match the permanent of the appropriate matrix, so that f and g are not identical. Consequently, they can agree on at most r elements of $\mathbb{Z}_p$, so that the probability that a, chosen uniformly at random from $\mathbb{Z}_p$ is a point of agreement is at most $\frac{r}{p}$. Since $p > r!$ and the total probability of a spurious acceptance is at most

$$\frac{\left(2(r - 1) + (r - 2) + \cdots + 1\right) r}{p},$$

the proof is complete. □

The following result is originally due to Shamir [**401**], but we follow the simpler proof by Shen [**407**] as described in [**28**].

THEOREM 4.3.5. **IP** = **PSPACE**.

PROOF. Consider the set B of propositional formulas $\varphi(X_1, \ldots, X_n)$ with propositional atoms $X_1, \ldots, X_n$. A *quantified Boolean formula* is an expression of the kind

$$Q_1 X_1 Q_2 X_2 \cdots Q_n X_n \varphi(X_1, \ldots, X_n),$$

where $\varphi \in B$ and where for each i, the quantifier Q_i is either $\forall$ or $\exists$. If ψ is a quantified Boolean formula, we say that it is true if it is a true predicate sentence where the quantifiers range over the Boolean constants $\top$ and $\bot$. The problem of determining which quantified Boolean formulas are true is known to be **PSPACE** complete (see, for instance, [**28**]). Consequently, to show that **PSPACE** $\subseteq$ **IP** it suffices to show that this problem has an interactive proof system.

We can read any propositional formula as a polynomial: we interpret $\neg X_i$ as $(1 - X_i)$, we interpret conjunction as a product; disjunction may be interpreted by deMorgan's law. If the variables take values from $\{0, 1\}$, the polynomial P_φ interpreting φ will also take values from $\{0, 1\}$, and this value will correspond to the truth of the formula. We can interpret $\forall X_i$ as a product of the cases $X_i = 0$ and $X_i = 1$, and $\exists X_i$ as a sum, where the formula is true if the polynomial evaluates to some value greater than zero. A first intuition to give interactive proof for true Boolean formulas is simply to give interactive proofs for an equation asserting the value of the appropriate polynomial. However, since the products tend to increase the degree of the polynomial, they may give us a polynomial that the time-bounded verifier cannot evaluate.

To avoid this, we introduce another "quantifier"-like operator. We define

$$L_i : \mathbf{Z}_p[X_1, \ldots X_n] \to \mathbf{Z}_p[X_1, \ldots, X_n]$$

by

$$\begin{aligned} L_i(p(\overline{X})) \quad := \quad & X_i p(X_1, \ldots, X_{i-1}, 1, X_{i+1}, \ldots, X_n) + \\ & +(1 - X_i)p(X_1, \ldots, X_{i-1}, 0, X_{i+1}, \ldots, X_n). \end{aligned}$$

This "linearization" operator has the property that for all $\overline{X} \in 2^n$, we have $L_i(p(\overline{X})) = p(\overline{X})$. Note also that $L_i(p)$ always has degree 1 in X_i. Consequently, we have

$$\begin{aligned} & Q_1 X_1 Q_2 X_2 \cdots Q_n X_n \varphi(X_1, \ldots, X_n) = \\ = Q_1 X_1 L_1 Q_2 X_2 L_1 L_2 \cdots Q_n X_n L_1 L_2 \cdots L_n \varphi(X_1, \ldots, X_n). \end{aligned}$$

Moreover, the expression on the right has size $O(n^2)$.

Now let $f \in \mathbb{Z}_p[X_1, \ldots, X_n]$, and suppose, for induction, that for any $\bar{a} \in \mathbb{Z}_p^n$, there is an interactive proof system for $f(\bar{a}) = C_0$ in which the verifier accepts with probability 1 if the equation is true and rejects with probability $1 - \epsilon$ if it is false. To verify $\exists X_1 f(X_1, a_2, \ldots, a_n) = C_1$, the prover sends a polynomial $g \in \mathbb{Z}_p[X_1, \ldots, X_n]$ of appropriate degree r, and the verifier checks it for equality to $f(X_1, a_2, \ldots, a_n)$ in the following way: If $g(0) + g(1) \neq C_1$, the verifier rejects. Otherwise, it sends the prover a, uniformly randomly chosen from $\mathbb{Z}_p$, and asks for a proof that $g(a, a_2, \ldots, a_n) = f(X_1, a_2, \ldots, a_n)$. As in the proof of Theorem 4.3.4, the probability that distinct g, f of degree r agree on a is at most $\frac{r}{p}$. Similarly, to verify $\forall X_1 f(X_1, a_2, \ldots, a_n)$, we follow the same procedure, except that the verifier checks $g(0)g(1) = C_1$. Finally, to verify $LX_1 f(X_1, a_2, \ldots, a_n)$, the prover sends a polynomial $g(X_1, \ldots, X_n]$ of appropriate degree, and the verifier checks if $a_1 g(0) + (1 - a_1)g(0) = C_1$. If not, it rejects. Otherwise, it picks a random $a \in \mathbb{Z}_p$, and asks for a proof that $g(a) = f(a, a_2, \ldots, a_k)$.

As in the proof of the previous theorem, with high probability the new statement to be verified is still false if the original was false. □

Note that the statement that $S \in \mathbf{IP}$ is a statement about winning strategies for certain games. The games in question are similar to, but not precisely like, several types of games that are well-studied. At least one reasonable construction is this: If $\sigma \notin S$, then there is no winning strategy for player P in a game where P wins if V accepts and loses otherwise. If $\sigma \in S$, then P does have a winning strategy for this game.

The random coins are at least one level of preventing a straightforward interpretation of this, and they are essential to the definition of **IP**. This randomness can be incorporated by viewing the game as a *stochastic game* in the sense of Shapley [**403**]. In these games, when the players choose their alternatives at a particular stage, the game moves to a new position determined by a random variable that depends on the position and the alternatives chosen.

The usual treatment of these games, however, centers on their expectation, rather than the concentration required in the definition of the class **IP**. For instance, these games have equilibria [**337, 336, 454**]. Condon [**116**] showed that the complexity of deciding which player has the greatest chance of winning is of class $\mathbf{NP} \cap \mathbf{coNP}$, at least for a large class of stochastic games. What is wanted for **IP** is not the expected value of the game, but the probability that each player will win.

An approach to this has been made in recent research of Kiefer and others [**281**]. In this paper, it is shown that these games sometimes exhibit a property called "strong determinacy." This is exactly the condition that one player has a strategy that will enforce that a winning condition holds with probability at least (or at most) c. If $c = 1$, all such games are strongly determined. However, if $c \in (0, 1)$, which is what we care about for interactive proofs, this condition may not hold.

There is still another obstace. As Condon [**116**] points out, much of the literature on stochastic games does not impose resource bounds on the players, and the assymetric resource bound (polynomial-time verifier, unbounded prover) in interactive proofs is a significant barrier between the game research and the **IP** research.

While the quantum features of $\mathbf{MIP}^*$ are beyond the scope of the present work, is is germane to point out to the reader that the interactive proofs described here are a natural entry point to understanding the recent result that $\mathbf{MIP}^* = \mathbf{IP}$, proved in [**253**], and the related work on the Connes Embedding Problem [**202**].

4.3.2. Interactive Proofs as Proofs. While complexity theorists are perfectly comfortable calling the objects of the previous section "proofs," this will likely seem strange to logicians. We are accustomed to seeing a proof as a sequence, to be sure, but it is a sequence whose every step consists of syntactically meaningful "calculations" to determine the truth value of a proposition. More exact definitions could be given, and have been. Is there, then, some meaningful sense in which an interactive proof (in the sense of the previous section) is a proof?

There is actually something like interactive proof in logic. Hintikka and others have described a "Game-theoretic semantics" for various logics, including classical first-order logic (see, for instance, the survey [**230**]). This does not transparently encompas the full power of **IP**, but it suggests a direction for further work in this area.

DEFINITION 4.3.6. Given a first-order L-structure $\mathcal{M}$ and an L-sentence φ, we define the game $G(\varphi, \mathcal{M})$ between a *prover* and a *verifier* by induction on the form of φ as follows:

(1) If φ is an atomic formula, the prover wins if φ is true in $\mathcal{M}$, and loses otherwise.
(2) If $\varphi = \neg\psi$, then the prover and the verifier trade positions and continue the game as $G(\psi, \mathcal{M})$.
(3) If $\varphi = \psi_1 \vee \psi_2$, then the prover chooses $i \in \{1, 2\}$, and the game continues as $G(\psi_i, \mathcal{M})$.
(4) If $\varphi = \psi_1 \wedge \psi_2$, then the verifier chooses $i \in \{1, 2\}$, and the game continues as $G(\psi_i, \mathcal{M})$.
(5) If $\varphi = \exists x \psi(x)$, then the prover chooses some c in the domain of $\mathcal{M}$, and the game continues as $G(\psi(c), \mathcal{M})$.
(6) If $\varphi = \forall x \psi(x)$, then the verifier chosses some c in the domain of $\mathcal{M}$, and the game continues as $G(\psi(c), \mathcal{M})$.

Now if φ is true, then the prover always has a winning strategy — a way of making all necessary choices in such a way as to win. If φ is false, then the verifier always has a winning strategy. Indeed, in keeping with the results of Section 4.1.3, if $\mathcal{M}$ is finite, then both the verifier and the prover, when they have a winning strategy, have one that they can compute in time polynomial in the size of the problem instance $(\varphi, \mathcal{M})$.

In that sense, the game-theoretic semantics of Hintikka represent another, perhaps more concrete proof, that for any finite structure $\mathcal{M}$, the set of sentences true in $\mathcal{M}$ is of class **IP**. At the same time, the moves of the game look very much like an actual proof of φ (from, say, the complete diagram of $\mathcal{M}$).

Of course, this game uses very little of the power available in **IP**. Not only is the prover using a time-bounded strategy, but the randomization is completely absent, and with it the opportunity for the wrong conclusion to be reached with some positive probability.

It is not immediately clear what strengthening of the proof system of Definition 4.3.6 would be logically sensible and have a chance of matching the power of **IP**.

Hintikka and Sandu point out that game-theoretic semantics have been used for quite a long time for extremely diverse logics. It may be reasonable, then, to ask what a game-theoretic semantics for continuous first-order logic would look like. This seems like a difficult question for several reasons. At the level of quantifiers, there is, in general, no *a priori* reason that suprema and infema must be realized, meaning that those clauses must look very different from the ones in Definition 4.3.6.

Moreover, even handling $\psi_1 \dot{-} \psi_2$ seems difficult. However, if the players are allowed randomization, it seems germane to point out that if r is picked uniformly at random from $[0, 1]$, the truth value of $\psi_1 \dot{-} \psi_2$ corresponds exactly to the probability that $\mathcal{M}(\psi_2) \leq r \leq \mathcal{M}(\psi_1)$. Making a game for that is still not straightforward, but it gives some cause for hope that there may be something like a proof system corresponding to **IP**.

4.4. Word Problems

??

In Section 4.2, we suggested two kinds of randomization that could be considered in computation. The first kind, which we have considered up to this point, involves an extrinsic randomization: the algorithm has access to some random bits that it can use, for instance, to simulate, or pick some "place" to check. A second possibility is randomization in the input. This approach has a long history in the theory of algorithms (see, for instance, [**124**]), where it is used to consider how an algorithm would perfom under conditions other than the best or worst case. Examples arising in group theory have led to a collection of new work in computability.

4.4.1. Groups with Solvable Word Problem. Up to this point, our description of randomized computation has focused on a randomization within the algorithm. There is a very different dimension of randomization possible, and it occurs commonly in practice: randomization in the input.

Consider, for example, the simplex algorithm in linear programming. It is well-known that the worst case complexity of the algorithm is exponential [**284**]. However, the algorithm empirically runs in such short time on almost all practically useful problem instances to make it the industry standard [**79**]. Intuitively, the "hard" instances of the problem that establish the worst-case complexity are relatively rare, and "most" instances allow a short running time.

In complexity, one way to handle this is with "average-case" complexity analysis, which can be sensitive to a choice of distribution on inputs. Moreover, a computation that does not halt at all, if it arises with any nonzero probability, must cause significant problems with a complexity measure based on the expected value. A more robust method is used for computability, and it first arose in the context of group theory.

Group theory is perhaps a natural field for the consideration of such problems. The word problem for groups is not solvable, but the standard examples proving that it is unsolvable are not only special groups; they are groups in which particular — apparently rare — words witness the unsolvability.

We first consider the classical problem, and in the following section we will take up the model of computability used to formalize the informally rare nature of the obstructions to computability.

It is well-known that a group can be presented by a list of generators and a list of relations. In particular, we write

$$G = \langle \{g_i : i \in I\} | \{r_i : i \in J\} \rangle$$

where each r_i is a word in the letters g_i and their inverses. In this presentation system, G is the quotient of the free group on generators $\{g_i : i \in I\}$ by the smallest normal subgroup containing all the relators $\{r_i : i \in G\}$. Any group can be presented in this way, and the presentations of groups are highly non-unique. Indeed, a significant open problem in group theory, the Andrews-Curtis Conjecture, involves recognizing presentations of the trivial group [**20**]. Additional background on group presentations can be found in standard algebra texts, such as [**248**], or in the group theory text [**385**], which gives one of the standard expositions on the unsolvability of the word problem.

In 1911, Dehn [**134**] identified three "Fundamental Problems Of Infinite Discontinuous Groups." He envisioned each group being given by finitely many generators and relations. The three problems were as follows:

The Word Problem: (Dehn called this the "Identity Problem.") In a particular presentation of a particular group, give an algorithm which will, given an element of the group (in the form of a word on the generators and their inverses, decide whether that element is equal to the identity or not.

The Conjugacy Problem: (Dehn called it the "Transformation Problem.") In a particular presentation of a particular group, give an algorithm which will, given two elements of the group, decide whether those elements are conjugate or not.

The Isomorphism Problem: Given two group presentations decide whether the two groups are isomorphic or not.

Of course, for many well-known classes of groups, these problems are known to be solvable. Dehn himself showed that the word problem for fundamental groups of certain manifolds must be solvable [**135**]. Of course, positive solutions tend to require less logical sophistication than negative solutions. However, after the definition of computability in the 1930s, it was clear, at least, what a negative solution could look like.

Indeed, not long after Dehn, Thue had posed the word problem for semigroups [**443**], and Post [**372**] proved it unsolvable in 1947 by a method that, with appropriate modificaitons, would also settle the word problem for groups.

THEOREM 4.4.1 (Post). *There is a finitely presented semigroup P such that the word problem for P is not computable.*

PROOF. Let T be a universal Turing machine with states $Q = \{q_0, \ldots, q_{n_s}\}$, where q_0 is the halting state, and tape alphabet $S = \{\sigma_1, \ldots, \sigma_{n_A}\}$. Consider the free semigroup F generated by $Q \cup S$. Within this semigroup computation states are represented by the words with exactly one letter from Q. We form the semigroup P with relations prescribing that two computation states are equal if T prescribes a transition from one to another, and that q_0 is equal to the identity.

The result follows from the fact that it is not possible to decide, from input n, whether a universal Turing machine will halt with an empty tape on input n. □

To produce a *group* with unsolvable word problem, we must define the computation state words more carefully: a word is said to be *special* if it is of the form $\Sigma_1 q_i \Sigma_2$, where $\Sigma_1, \Sigma_2 \in S^{<\omega}$. The difficulty of groups is that with inverses, words may be equivalent in more subtle ways.

Novikov [**356**], and independently Boone [**72, 73**] demonstrated that there is indeed a finitely generated group with unsolvable word problem. The parts of the proof that concern us here are not the most difficult parts of the proof. The full algebraic details, including the proof of the critical lemma, can be found in Rotman's book [**385**]. Since most of those details are not germane to the present work, the reader is referred to that treatment, while we give only a brief outline of the proof below.

THEOREM 4.4.2. *There is a group with unsolvable word problem.*

PROOF. Let S and Q be as in Post's construction, and let $R = \{B_i = \Gamma_i\}$ be the set of relators in Post's construction. Let G be generated by

$$S \cup Q \cup \{k, t, x, y\} \cup \{\ell_i : i \in R\} \cup \{r_i : i \in R\}$$

and defined by the following relations (for all values in the range of b and i):

$$\begin{aligned} s_b y &= y y s_b \\ x s_b &= s_b x x \\ s_b \ell_i &= y \ell_i s_b \\ r_i s_b &= s_b x r_i x \\ B_i &= \ell_i \\ \Gamma_i &= r_i \\ t\ell_i &= \ell_i t \\ ty &= yt \\ r_i k &= k r_i \\ xk &= kx \\ \left(q^{-1}tq\right) k &= k \left(q^{-1}tq\right). \end{aligned}$$

We note that all of the words of Post's semigroup are still words of G, and the definition of "special" words above is still meaningful. Again, it is the special words that will allow us to encode computation.

Let Δ be a special word in G which is equal in P to q_0. Then we can show from the relators that in G, we have $\Delta = Lq_0R$, where L is a word in $\{y, \ell_i : i \in R\}$ and R is a word in $\{x, r_i : i \in R\}$. Then, by the commutativity relations of G, it follows that $\left(\Delta^{-1}t\Delta\right) k = k \left(\Delta^{-1}t\Delta\right)$. Through some involved group-theoretical arguments, it is possible to show that the converse also holds. At this point, we know that there can be no algorithm to decide, for any special word, whether $\left(\left(\Delta^{-1}t\Delta\right) k\right) \left(k \left(\Delta^{-1}t\Delta\right)\right)^{-1}$ is the identity. □

4.4.2. Groups with Generically Solvable Word Problem. In some sense the proof of Theorem 4.4.2 is unsatisfying: While the result is, in fact, established, there is nothing to prevent the existence of an algorithm that would solve the word problem for the vast majority of words in the group. The words representing the obstruction are called "special" for a reason (considerably predating any of the results or considerations of the present section). They are, in fact, relatively rare.

In 2003, Kapovich, Myasnikov, Schupp, and Shpilrain formalized this dissatisfaction [**261**]. In particular, they showed that for large classes of groups — including the group constructed in Theorem 4.4.2, the word problem is almost solvable, in the sense that there is an algorithm (indeed, a linear time algorithm) that will solve almost all instances of the word problem.

To this point, we have not yet defined what "almost all" should mean in this context, and a reader who has kept in mind that all of the structures involved are countable is doubtless ready to see a definition.

Definition 4.4.3. Let Σ be a finite alphabet, and let $(X^*)^k$ be, as usual, the set of k-tuples of words on X. Let B_n be the set of elements of $(X^*)^k$ where the sum of the lengths of the words involved is at most n, and $S \subseteq (X^*)^k$. Then the *asymptotic density* of S, denoted $\rho_a(S)$ is defined as

$$\limsup_{n\to\infty} \frac{|S \cap B_n|}{|B_n|}.$$

In this sense, "almost all" means on a set of density 1, or (equivalently) except for a set of density 0. Of course, there is still some flexibility in the definition, and we will see in the following section how this flexibility leads to several notions of effectiveness. For the present, however, we follow the original work.

DEFINITION 4.4.4. Let $P \subseteq (X^*)^k$. We say that P is *generically computable* if there is a Turing machine T with domain S such that $\rho_a(S) = 1$, and such that if $\sigma \in S$, we have $T(\sigma) = \chi_P(\sigma)$.

THEOREM 4.4.5. *Let G be a finitely generated group, such that G has a finite index subgroup with infinite quotient $\overline{G}$ for which the word problem is solvable. Then the word problem for G is generically solvable.*

Before considering the proof of this theorem, we point out the following corollary, which clarifies the situation of the Boone group.

COROLLARY 4.4.6. *The group G constructed in Theorem 4.4.2 has generically solvable word problem.*

PROOF. Consider the subgroup $\overline{G}$ of G generated by the set $\{r_i : i \in R\}$. Now this $\overline{G}$ is a quotient of G, and is a non-Abelian free group on its generators. Since such a group has solvable word problem, the Corollary follows from the Theorem. □

Actually, Theorem 4.4.5 follows from another technical result of the same paper. Given a group presentation and a subgroup H, the *membership problem* for H is that of determining whether a word is an element of H. Note that the word problem for a group is equivalent to the membership problem for its trivial subgroup.

PROPOSITION 4.4.7. *Let G be a finitely generated group and $H \leq G$ a finitely generated subgroup of infinite index. Let $G_1 \geq H$ be a finite-index subgroup of G, and $\phi : G_1 \to \overline{G}$ an epimorphism to some group $\overline{G}$. If $\overline{H} := \phi(H)$ is contained in an infinite index $\overline{K} \leq \overline{G}$ and the membership problem for $\overline{K}$ is solvable, then the membership problem for H is generically solvable.*

Before entering the interesting details of Proposition 4.4.7, let us see how it implies Theorem 4.4.5.

PROOF OF THEOREM 4.4.5. Let G_1 be a finite index subgroup of G with infinite quotient $\overline{G}$, where $\overline{G}$ has solvable word problem, with $\phi : G_1 \to G$ the standard projection. Let H be the trivial subgroup of G and $\overline{K}$ the trivial subgroup of $\overline{G}$. Then $\varphi(H) \leq \overline{K}$. Note that $\overline{K}$ has infinite index in $\overline{G}$, and that the word problem of $\overline{G}$ is exactly the membership problem of $\overline{K}$, so that the membership problem of $\overline{K}$ is, by assumption, solvable. By Proposition 4.4.7, the membership problem for H is generically solvable. But this problem is exactly the word problem for G. □

We now undertake the proof of Proposition 4.4.7. As usual, the interested reader will want to follow up with the original paper, which has both stronger conclusions and more details of the proof.

PROOF OF PROPOSITION 4.4.7. Let A be a generating set of length k for G and let B be a finite generating set for G_1. Let $\pi : F(A) \to G$ be the standard projection of the free group on A, denoted $F(A)$ to G. Let $K_1 := \phi^{-1}\left(\overline{K}\right)$, and

$K_2 := \pi^{-1}(K_1)$. Note that $\overline{K}$ has infinite index in $\overline{G}$ by assumption, and K_1 has infinite index in G_1. Consequently, K_2 has infinite index in $F(A)$.

We let z_n denote the number of words σ in $A \cup A^{-1}$ of length n, with $\pi(\sigma) \in K_2$, and note that there are

$$\frac{(2k)^{n+1} - 1}{2k - 1}$$

distinct words in $(A \cup A^{-1})$ of length n. Consequently, the assymptotic density of K_2 is given by

$$\lim_{n\to\infty} \frac{z_n}{\frac{(2k)^{n+1}-1}{2k-1}}.$$

By assumption, $H \leq K_1$, so that if σ is a word in $A \cup A^{-1}$, but not in K_2, it follows that $\pi(w) \in G - H$. Consequently, if we can prove that the asymptotic density of K_2 is 0, the proof is done.

We define the *Cayley graph of* $F(A)$, as the graph whose vertices are the elements of $F(A)$, and where two vertices σ, τ are adjacent if and only if $\sigma = \tau x$ for some $x \in A$. In the standard graph metric, where adjacent vertices have distance 1, the ball of radius n about the origin contains exactly the words of length at most n.

LEMMA 4.4.8. *Let* Γ *be the Cayley graph of a group with* d*-generators, where* $d \geq 3$*. Let* ζ_n *denote the number of paths of length at most* n *from the identity to the identity. Then*

$$\lim_{n\to\infty} \frac{\zeta_n}{d^n} = 0.$$

PROOF. if b_n denotes the number of paths without backtracks of length at most n from the identity to the identity, and $\beta := \limsup_{n\to\infty} \sqrt[n]{b_n}$. It is known [**46**] that we can assume $\beta < d$, so that there is some $N_0 \geq 1$ and $0 < b < d$, so that

$$\lim_{n\to\infty} \frac{b_n}{d^n} \leq \frac{b^n}{d^n} \to 0.$$

The lemma follows, by some well-established limit comparison theorems. □

It follows from the definition of the Cayley graph that $z_n = \zeta_n$. Consequently, by the lemma, K_2 has asymptotic density 0, as required. □

The group theory is important in this theorem, and to generalize this notion of computability it is necessary to replace it. In particular, the notion of asymptotic density depended on properties of random words in a group, or equivalently, random walks in a graph. It will be the business of the next section to pose a definition of a notion of computation that carries the sense of a generically solvable word problem, but in a more general setting.

While we turn from the group theoretic origins of generic computability for the present, we will return to them. There is important work in this area on dynamics, to which we will return in Chapter 8, and random walks in a graph will be important when we take up the subject of expander graphs in Chapter 5.

4.5. Generic and Coarse Computability

4.5.1. Generic Computability. In 2012, Jockusch and Schupp [**254**] extended the theory of generic solvability to subsets of $\mathbb{N}$, the native realm of computability theory. By identifying $n \in \mathbb{N}$ with its unary representation, the definitions of the previous section reduce to something reasonable on the natural numbers.

DEFINITION 4.5.1. Let $S \subseteq \mathbb{N}$.

(1) The *density of S up to n*, denoted $\rho_n(S)$ is given by
$$\frac{S \cap \{0, 1, 2, \ldots, n\}}{n+1}$$
(2) The *density of S*, denoted $\rho(S)$ is given by
$$\lim_{n\to\infty} \rho_n(S).$$

Under the interpretation of natural numbers by their unary representations, this definition coincides exactly with the one in Section 4.4.2. We have the corresponding definition.

DEFINITION 4.5.2. *generically computable—textbf* Let $P \subseteq \mathbb{N}$. We say that P is generically computable if and only if there is a Turing machine T such that $T(n) = \chi_P(n)$ wherever T is defined, and T is defined on a set of density 1.

Jockusch and Schupp made several initial observations about these sets.

PROPOSITION 4.5.3. *Every Turing degree contains a set which is generically computable.*

PROOF. Let $\mathbf{d}$ be a Turing degree, with $A \in \mathbf{d}$. We set $B = \{2^m : m \in A\}$. Note that $B \in \mathbf{d}$, since the set of natural numbers which are not powers of 2 is computable. We define a Turing machine T so that
$$T(n) = \begin{cases} 1 & \text{if } n = 2^k \text{ for some } k \\ \uparrow & \text{otherwise} \end{cases}$$
Now $T(n) = \chi_B(n)$ for all $n \in dom(T)$, and $dom(T)$ has density 1. □

Even after the structural evidence of the generically solvable word problem, it is still natural, after Proposition 4.5.3 to wonder how nearly trivial the concept of generic computability could be.

PROPOSITION 4.5.4. *Every nonzero Turing degree contains a set which is not generically computable.*

PROOF. Let $\mathbf{d} >_T \emptyset$, and let $A \in \mathbf{d}$. Let
$$R_k = \{m : 2^k | m \text{ and } 2^{k+1} \nmid m\}.$$
We write
$$\mathcal{R}(A) = \bigcup_{k \in A} R_k.$$
Now if T were a Turing machine witnessing that $\mathcal{R}(A)$ is generically computable, we could decide whether $n \in \mathcal{R}(A)$ by running T on all multiples of 2^n until it answers on one. We know that T must answer on some element of R_n, since this set has density $2^{-k} > 0$. Since $\mathcal{R}(A) \equiv A$, the result follows. □

Recall that a set is said to be *bi-immune* if both it and is complement are infinite, with no infinite computably enumerable subsets.

PROPOSITION 4.5.5. *No bi-immune set is generically computable.*

PROOF. Suppose that Turing maching T witnesses that A is generically computable. Let C_0 be the set of n such that $T(n) = 0$ and C_1 the set of n such that $T(n) = 1$. Now $C_0 \subseteq \overline{A}$ and $C_1 \subseteq A$, and $C_0 \cup C_1$ has density 1. Consequently, C_0 and C_1 cannot both be finite, so that A cannot be bi-immune. □

Of course, probabilities other than 1 are still interesting, although they do depend more sensitively on the distribution of possible inputs. Since we, by definition, have fixed a distribution (the uniform distribution), it is possible to talk about these sets. Downey, Jockusch, and Schupp gave the following definition, which extends generic computability in this natural way.

DEFINITION 4.5.6. Let $S \subseteq \mathbb{N}$. We define the lower density of S, denoted $\rho_\ell(S)$ as

$$\liminf_{n\to\infty} \frac{|S \cap \{0, 1, \ldots, n\}|}{n+1}.$$

We then say, for any $r \in \mathbb{R}$, that a set $A \subseteq \mathbb{N}$ is partially computable at density r if there is a Turing machine T such that $T(n) = A(n)$ for all n in the domain of T, and such that the domain of T has lower density greater than or equal to r.

In the same paper, several observations were made about this definition. To begin with, this definition differs significantly from the definition of generic computability.

PROPOSITION 4.5.7. *Every nonzero Turing degree contains a set which is partially computable exactly at the densities strictly less than 1.*

PROOF. Let the sets R_k and the operator $\mathcal{R}$ be as in the proof of Proposition 4.5.4. Let $\mathbf{d}$ be a nonzero Turing degree, with $A \in \mathbf{d}$. Let $\epsilon > 0$, and note that $\mathcal{R}(A)$ is partially computable at density $1 - \epsilon$, for a Turing machine witnessing its partial computability need not have in its domain any element of R_k for sufficiently large k. As we observed, however, $\mathcal{R}(A)$ cannot be partially computable at density 1. □

In this sense, it is easier to believe in a set which is partially computable at all densities less than 1 than it is to believe in one which is partially computable at some proper subset of those densities. The following result shows that such sets do exist.

THEOREM 4.5.8. *Let $r \in [0, 1]$. Then there is a set A which is partially computable at density r, but not at any higher density, if and only if r is left-Σ^0_3.*

PROOF. Let $r \in [0, 1]$ be left-Σ^0_3. We first produce a computably enumerable set C with lower density r and a simple set S of density 0.

Toward the first goal, we first produce a Δ^0_2 set $\tilde{C}$ with lower density r. We construct a Δ^0_2-computable sequence $(q_s : s \in \mathbb{N})$ where $\liminf_{s\to\infty} q_s = r$. Let A be the set of rationals less than r, a Σ^0_3 set. We can then construct a Δ^0_2 sequence $(A_s : s \in \mathbb{N})$ of finite sets such that for all $x \in A$, we have $x \in A_s$ for all sufficiently large s, such that there are infinitely many s such that $A_s \subseteq A$, and such that

$0 \in A_s$ for all s. We set $q_s := \max A_s$. Now we will inductively define $\tilde{C} = \bigcup_{i \in \mathbb{N}}$ and a sequence $(s_i : i \in \mathbb{N})$. At stage 0, we set $s_0 = 1$ and $\Gamma_0 = \{0\}$. At stage t, we check whether

$$\frac{|\Gamma_{t-1}|}{s_{t-1}+1} < q_t.$$

If so, we find the least k such that

$$\frac{|\Gamma_{t-1}|+k}{s_{t-1}+1+k} \geq q_t$$

and set $s_{t+1} = s_t + k$ and $\Gamma_t = \Gamma_{t-1} \cup \{s_t+1, \ldots, s_t+k\}$. Otherwise, we find the least k such that

$$\frac{|\Gamma_{t-1}|}{s_{t-1}+1+k} < q_t$$

and set $s_{t+1} = s_t + k$ and $\Gamma_{t+1} = \Gamma_t$. All of these operations can be completed with a Δ^0_2 oracle, so that $\tilde{C}$ is Δ^0_2, as required.

Now let $\tilde{C}_s$ be a computable approximation to $\tilde{C}$. At stage 0, set $C_0 = \emptyset$ and $t(n,0) = n+1$, and $t(-1,n) = -1$ for all n. At stage $s+1$, we let n_s be the least $n \leq s$, if any, such that $\rho_{t(n,s)}\left(\tilde{C}_s\right) \neq \rho_n(C_s)$. We then find a finite set $C_{s+1} \supseteq C_s$ and a number $\hat{t}$ greater than any number in C_s and greater than $t(n,s)$ such that $\rho_c(C_{s+1}) = \rho_n(C_s)$, with the additional requirements that C_{s+1} agrees with C_s between n_s and s, and that C_{s+1} includes an initial segment of $[t(n,s), \infty)$. It can be shown that for each $k > 0$ there are only finitely many s where $n_s = k$, and that $\rho_{\lim_{s\to\infty} t(k,s)}(C_s) = \rho_k\left(\tilde{C}_s\right)$. We set $C = \bigcup_{s \in \mathbb{N}} C_s$, so that C is a computably enumerable set of lower density r.

We also produce a simple set S of density 0. Indeed, for each e, we look for the first $n > e^2$ with $n \in W_e$. If such and e exists, we include it in S. Now any computably enumerable set disjoint from S must be finite, but S contains at most e elements less than e^2 for each e, so that

$$\rho(S) \leq \lim_{n\to\infty} \frac{n}{n^2} = 0.$$

Let $A = C \cup S$. Then A is computable at density r, because $C \subseteq A$. However, if $r' > r$ and T witnesses that A is partially computable at density r', then the set $\{n : T(n) = 0\}$ must be disjoint from S (since $S \subseteq A$), infinite (since density $r' - r > 0$ of the domain of T must be outside A), and computably enumerable. Since S is simple, this is impossible.

On the other hand, let r fail to be left-Σ^0_3, and let T witness that A is partially computable at density r. Suppose T has domain D. From the definitions, it is clear that $r' := \rho_\ell(D) = \liminf_{n\to\infty} q_n$, where $(q_n : n \in \mathbb{N})$ is a Δ^0_2 sequence of rationals. If r' is irrational, then for any rational q, and for all but finitely many of the q_n, we have $q < q_n$, so that r' is left-Σ^0_3. Since $r' \geq r$, we know that $r' > r$, with A partially computable at density r'. □

It is worth noting the similarities between the construction of $\tilde{C}$ in the last proof and the proof of Theorem 4.2.12. In both cases, the construction proceeds by recruiting witnesses to the necessary case to establish a needed probability.

Definition 4.5.9. Let $A \subseteq \mathbb{N}$. Then the *partial computability bound of* A, denoted $\alpha(A)$, is the supremum of all r such that A is computable at density r.

PROPOSITION 4.5.10. *Let $r \in (0,1)$. Then there is a set with partial computability bound r.*

PROOF. Let $r = \sum_{i=1}^{\infty} b_i 2^{-1}$ where $b_i \in \{0,1\}$ and infinitely many of the b_i are 1. Let S be the simple set of density 0 constructed above, and let

$$R_k = \{m : 2^k | m \text{ and } 2^{k+1} \nmid m\}$$

as before. We set $D = \bigcup_{b_i=1} R_i$, and set $A = D \cup S$. As before, A is partially computable at any density strictly less than r, but not at any density greater than r. □

4.5.2. Coarse Computability. Jockusch and Schupp [**254**] identified and studied a second notion of approximate computability. As we suggested earlier, the key difference is in what it means — even after agreeing on the meaning of "almost all" — for an algorithm to solve almost all instances of the problem. Of course, all correct solutions are alike. The real question concerns what the algorithm does on the places where it does not give a correct solution. In generic computability, the algorithm must be correct everywhere it halts: it can either be correct or not halt at all.

In this second notion, however, the algorithm is required to halt on all inputs, and be correct almost everywhere.

DEFINITION 4.5.11. Let $P \subseteq \mathbb{N}$. We say that P is coarsely computable if there is a Turing machine T which is defined on all natural numbers, and such that for some set $S \subseteq \mathbb{N}$ of density one, we have $T(n) = \chi_P(n)$ for all $n \in S$.

Jockusch and Schupp point out that this notion says something important about the initial issue of word problems for groups. Indeed, the following result, in light of the proofs of the previous section, suggests that the "coarse" in the definition is very coarse indeed. Importantly, we need not know *where* the machine goes wrong, only that it do so very seldom.

PROPOSITION 4.5.12. *Every finitely generated group has coarsely computable word problem.*

PROOF. If G is a finitely generated infinite group, then the set of words not equal to the identity has density 1, so that a Turing machine rejecting every input coarsely computes the word problem. If G is finite, then its word problem is computable. □

In view of the near-triviality of the preceding proof, it is tempting to think that coarse computability would be a notion permissive to the point of uselessness. However, a recent paper of Greenberg, J. Miller, Shen, and Westrick [**209**] demonstrates that its approach via the density of the points of agreement is exactly what is needed to settle an important question in effective dimension and algorithmic randomness. We say that two sets are *coarsely equivalent* if they agree on a set of density 1. We first note a preliminary result.

PROPOSITION 4.5.13. *If two sequences are coarsely equivalent, they have the same effective Hausdorff dimension.*

THEOREM 4.5.14 ([**209**]). *A sequence has effective Hausdorff dimension 1 if and only if it is coarsely equivalent to a 1-random sequence.*

PROOF. Let X be 1-random. By Proposition 3.4.7, it has effective Hausdorff dimension 1, and by Proposition 4.5.13, any Y coarsely equivalent to X must have the same effective Hausdorff dimension. The second direction is more involved.

Assume X has effective Hausdorff dimension 1, and let

$$s_m = dim(X \upharpoonright_{[2^m, 2^{m+1})} | X \upharpoonright_{2^m}).$$

Let

$$P = \{Y : \forall n \; K(Y \upharpoonright_n) \geq n\},$$

and let $\mathbb{P}$ denote the set of finite initial segments of elements of P. Let d denote the Hamming distance, and let E be the set of all sequences $(e_m : m \in \mathbb{N})$ such that $e_0 = 1$ and such that for all k either $e_k \in \left\{e_{k-1}, \frac{e_{k-1}}{2}\right\}$. For each $\bar{e} \in E$, we define a relation $\sim_{\bar{e}}$ on strings so that $\sigma \sim_{\bar{e}} \tau$ if and only if for all k we have

$$d\left(\sigma \upharpoonright_{[2^{k-1}, 2^k)}, \tau \upharpoonright_{[2^{k-1}, 2^k)}\right) \leq e_k.$$

We now define a sequence $(\epsilon_m : m \in \mathbb{N}) \in E$ such that for all m we have

(1) $\lim_{m \to \infty} \epsilon_m = 0$, and
(2) There is some $\tau \in \mathbb{P}$ of length 2^m with $\tau \sim_{\epsilon_0, \ldots, \epsilon_m} X \upharpoonright_{2^m}$.

We claim that for each m there is $\nu \in \mathbb{P}$ of length 2^{m+1} such that

$$\nu \sim_{(\epsilon_0, \ldots, \epsilon_m)} X \upharpoonright_{2^{m+1}} .$$

To this end, let A be the set of strings η of length 2^m such that for some $\hat{\tau} \in \mathbb{P}$ we have

$$\hat{\tau} \sim_{(\epsilon_0, \ldots, \epsilon_{m-1})} X \upharpoonright_{2^m}$$

and $\hat{\tau}\eta \in \mathbb{P}$. Since $\mathbb{P}$ is co-c.e., it follows that A is co-c.e. It can be shown that there is some $q < 1$ such that if there are at most 2^{q2^m} strings π whose distance from A is greater than ϵ_m. Call the computably enumerable set of all such strings B. Then, conditioned on $X \upharpoonright_{2^m}$, each $\pi \in B$ has a description of length $q2^m + m + O(1)$. Consequently, for sufficiently large m, every $\pi \in B$ will satisfy

$$dim(\pi | X \upharpoonright_2^m) < s_m$$

so that $X \upharpoonright_{[2^m, 2^{m+1})} \notin B$. In that case, there must be some $\eta \in A$ such that

$$d(\eta, X \upharpoonright_{[2^m, 2^{m+1})}) \leq \epsilon_m.$$

Let $\hat{\tau}$ witness that $\eta \in A$, and take $\nu = \hat{\tau}\eta$.

We now take the set Q_m of all such strings $\nu \in \mathbb{P}$ of length 2^{m+1} such that

$$\nu \sim_{(\epsilon_0, \ldots, \epsilon_m)} X \upharpoonright_{2^{m+1}} .$$

We have shown that these sets are nonempty, and every element of Q_m has a prefix from Q_{m-1}. By compactness, there is $Y \in P$ such that for all m, we have $Y \upharpoonright_{2^m} \in Q_m$. Since the differences between X and Y on a sequence of intervals whose length grows exponentially tends to 0, we have Y coarsely equivalent to X. □

Again, as in the case of generic computability, one can loosen the requirement of density 1 in the definition. Although the idea was implicit in [**146**], this was first done explicitly in [**231**].

DEFINITION 4.5.15. Let $A \subseteq \mathbb{N}$, and $r \in [0, 1]$. We say that A is coarsely computable at density r if there is a computable set B such that the lower density of the set of all n on which A and B agree is at least r.

DEFINITION 4.5.16. Let $A \subseteq \mathbb{N}$. Then the *coarse computability bound of* A, denoted $\gamma(A)$, is the supremum of all r such that A is coarsely computable at density r.

As we will see in Theorem 4.5.20, there are sets A with $\gamma(A) = 1$ which are nevertheless not coarsely computable.

4.5.3. Relationships between Generic and Coarse. It is natural to think that generic and coarse computability might be closely related, although intuition, by its nature, may vary from person to person. In fact, the two are nearly orthogonal. The first several results we consider in this area (as well as stronger forms of some of them) are from [**254**].

PROPOSITION 4.5.17. *There is a computably enumerable set which is coarsely computable but not generically computable.*

PROOF. Consider the simple set S of density 0 constructed in the proof of Theorem 4.5.8. Since S has density 0, it is coarsely computable. However, if machine T witnesses that S is generically computable, take $C_i = \{n : T(n) = i\}$, as before, and note that $C_0 \cup C_1$ has density 1. Since C_1 is contained in a set of density 0, it must also have density 0, leaving C_0 an infinite computably enumerable subset of the complement of S. □

PROPOSITION 4.5.18. *There is a generically computable computably enumerable set which is not coarsely computable.*

PROOF. If A_0, A_1 are disjoint computably enumerable sets whose union has density 1, both A_0 and A_1 are generically computable; indeed, a Turing machine can enumerate each set, and can give output exactly on $A_0 \cup A_1$. If we can construct such sets where A_1 is not coarsely computable, then the theorem will hold.

We recall the definition of the sets R_e from Proposition 4.5.4, and, as usual, let W_e denote the domain of machine T_e. We initialize $A_{0,0} = A_{1,0} = \emptyset$ and $r(e, 0) = min(R_e)$. At stage s, for each $e \leq s$, we check whether $R_e \upharpoonright_{r(e,s)} \subseteq W_{e,s+1}$.

If so, then we let

$$F = R_e \upharpoonright_{r(e,s)} - (A_{0,s} \cup A_{1,s}) .$$

For each $n \in F$, we include n in $A_{0,s+1}$ if $T_e(n) = 1$, and in $A_{1,s+1}$ otherwise. We set $r(e, s+1)$ so that at most half of the elements of $R_e \upharpoonright_{r(e,s+1)}$ are in $A_{0,s+1} \cup A_{1,s+1}$. Consequently, T_e and $A_{1,s+1}$ will differ on every element of F.

Otherwise, we find the least z in R_e which is greater than $r(e, s)$ with $z \notin A_{1,s}$. We then set $A_{1,s+1} = A_{1,s} \cup \{z\}$ and leave $A_{0,s+1} = A_{0,s}$ and $r(e, s + 1) = r(e, s)$.

Let $A_i = \bigcup_{j \in \mathbb{N}} A_{i,j}$. Now if T_e is total, then we have constructed a set of positive measure on which T_e and A_1 disagree, so that A_1 cannot be coarsely computable by any T_e. On the other hand, for each e, we will have almost every element of R_e enumerated into either A_0 or A_1 at some stage. Consequently, $A_0 \cup A_1$ has density 1, as required. □

PROPOSITION 4.5.19. *There is a computably enumerable set which is neither coarsely computable nor generically computable.*

PROOF. Recall from Proposition 4.5.4 the definition of the sets R_e, as before. Now we define

$$A = \bigcup_{e \in \mathbb{N}} (W_e \cap R_e) .$$

Now for each e, we know that A differs from the complement of W_e on all of R_e. Consequently, A cannot be coarsely computable, since if T witnesses that A is coarsely computable, the set $N = \{n : T(n) = 0\}$ is computably enumerable, and A can differ from the complement of this set only with density 0.

Now, toward contradiction, let T witness that A is generically computable. Let $C_i = \{n : T(n) = i\}$, and note, as usual, that C_0 is contained in the complement of A, that C_1 is contained in A, and that $C_0 \cup C_1$ has density 1. Then A can differ from the complement of C_0 only on elements outside $C_0 \cup C_1$, a set of density 0. However, C_0 is computably enumerable, and A must differ from it on a set of positive density. □

In view of these limitations, the following results on partial and coarse computability bounds, from a paper of Hirschfeldt, Jockusch, McNicholl, and Schupp [**231**], seem striking.

THEOREM 4.5.20. *For any $A \subseteq \mathbb{N}$, we have $\alpha(A) \leq \gamma(A)$.*

PROOF. Let $\epsilon > 0$, and let T be a partial Turing machine such that T agrees with A on its domain, and such that the domain of T has density $\alpha(A) - \frac{\epsilon}{2}$. We will find a computable computable C within the domain of T with lower density at most $\frac{\epsilon}{2}$ less than the lower density δ of the domain of T. Given such a C, we can define a Turing machine U to accept those n for which $n \in C$ and $T(n) = 1$, and to reject all others. Now U agrees with T on all of C, but U is total, so that A is coarsely computable at density $\alpha(A) - \epsilon$.

We now construct C. Let q be a rational between $\delta - \frac{\epsilon}{2}$ and δ. There is some n_0 so that the density of the domain of T up to n is at least q for all $n > n_0$. Now we define C to be the subset of the domain of T consisting of all elements k less than the maximum over all n between n_0 and k^2 of the least s such that the density up to n of the domain of T_s is greater than q. This set is computable. On the other hand, $\rho_n(B) \geq q - \frac{1}{n}$, so that the lower density of C is at least $\delta - \frac{\epsilon}{2}$. □

This inequality gives us, as a consequense, a set which has coarse computability bound 1, but which is not coarsely computable, since we know that there is a set which is generically computable but not coarsely computable. The two bounds need not be close.

PROPOSITION 4.5.21. *There is a set whose partial computability bound is 0, but whose coarse computability bound is $\frac{1}{2}$.*

PROOF. Let $I_n = [n!, (n+1)!)$, and define $\mathcal{I}(A) = \bigcup_{n \in A} I_n$. Now if T witnesses that $\mathcal{I}(A)$ is coarsely computable at density greater than $\frac{1}{2}$, then we can compute A in fhe following way: for some n_0 and for all $n > n_0$, the machine T must agree with $\mathcal{I}(A)$ on more than half of the elements of I_n. In that case, a majority vote decides whether $n \in A$. On the other hand, $\mathcal{I}(A) \equiv_T A$, so $\mathcal{I}(A)$ is coarsely computable just in case A was computable.

On the other hand, any set of positive lower density must intersect almost all of the sets I_n, so that if $\mathcal{I}(A)$ is partially computable at density $r > 0$, we again have A computable. □

Moreover, below either a 1-generic or a weak 2-random, the inequality of Theorem 4.5.20 becomes equality, so that all generically computable sets become partially computable. More exactly,

THEOREM 4.5.22. *Let $A \subseteq \mathbb{N}$, and $r \in [0,1]$. Let $B \leq_T A$. If A is either 1-generic or weakly 2-random, then if B is partially computable at density r, then B is coarsely computable at density r.*

PROOF. We first let $B = \Phi^A$, with T witnessing that B is partially computable at density r.

Consider first the case that A is 1-generic. We define a notion of forcing

$$P = \left\{\sigma \in 2^{<\omega} : \Phi^\sigma \perp T\right\}.$$

Note that P is computably enumerable. If there is n such that $A \upharpoonright_n \in P$, then B disagrees with T somewhere. Otherwise, since A is 1-generic, we have some n such that all τ extending $A \upharpoonright_n$ are outside P. In that case, we will define a total Turing machine U that agrees with B on a set of density at least r.

To define the behavior of U on input m, find $\sigma \supseteq A \upharpoonright_n$ such that $\Phi^\sigma(m)\downarrow$, and set $U(m) = \Phi^\sigma(m)$. Since $A \supseteq A \upharpoonright_n$, some such σ can always be found. However, since σ was not in S, we know that $U(m)$ must be compatible with B, so that U agrees with B thoughout the domain of T, which has density at least r.

Now if A is weakly 2-random, we define a Π^0_2 class P so that $A \in A$. We set

$$P = \left\{X : \Phi^X \text{ is total and compatible with } T\right\}.$$

Since A is weakly 2-random and included in P, we know that P must have positive measure. The Lebesgue density theorem provides that there is some string γ such that P includes at least relative measure .8 of the extensions of γ, so that for each n there is some $i_n \in \{0,1\}$ so that

$$\mu\left(\left\{X : \Phi^X(n) = i_n\right\}\right) \geq .4.$$

Then we define U so that $U(n) = i_n$. Now U is a total Turing machine that agrees with B throughout the domain of T, so that B is coarsely computable at density r. □

Bibliography

1. M. Abért, N. Bergeron, I. Biringer, T. Genander, N. Nikolov, J. Raimbault, and I. Samet, *On the grown tof L^2-invariants for sequences of lattices in Lie groups*, Annals of Mathematics **185** (2017), 711–790.
2. M. Abért, Y. Glasner, and B. Virág, *Kesten's theorem for invariant random subgroups*, Duke Mathematical Journal **163** (2014), 465–488.
3. N. Ackerman, C. Freer, A. Kwiatowska, and R. Patel, *A classification of orbits admitting a unique invariant measure*, Annals of Pure and Applied Logic **168** (2017), 19–36.
4. N. Ackerman, C. Freer, J. Nešetřil, and R. Patel, *Invariant measures via inverse limits of finite structures*, European Journal of Combinatorics **52** (2016), 248–289.
5. N. Ackerman, C. Freer, and R. Patel, *Invariant measures concentrated on countale structures*, Forum of Mathematics, Sigma **4** (2016), 1–59.
6. ———, *Countable infinitary theories admitting an invariant measure*, preprint, 2017.
7. ———, *The entropy function of an invariant measure*, Proceedings of the 14th and 15th Asian Logic Conferences, World Scientific Publishing, 2019, pp. 3–34.
8. N. Ackerman, C. Freer, and D. Roy, *On the computability of conditional probability*, Journal of the Association for Computing Machinery **66** (2019), 23:1–23:40.
9. ******E. W. Adams, *The logic of conditionals*, Reidel, 1975.
10. E. W. Adams, *A primer of probability logic*, CSLI Lecture Notes, no. 68, CSLI Publications, 1999.
11. S. Adams and A. S. Kechris, *Linear algebraic groups and countable Borel equivalence relations*, Journal of the American Mathematical Society **13** (2000), 909–943.
12. Scott Adams, *Dilbert*, (2001), October 25, `https://dilbert.com/strip/2001-20-25`.
13. B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter, *Molecular biology of the cell*, 5th ed., Garland, 2008.
14. M. Aldana, S. Coppersmith, and L. P. Kadanoff, *Boolean dynamics with random couplings*, Perspectives and Problems in Nonlinear Science, Springer, 2003, pp. 23–89.
15. K. Allen, L. Bienvenu, and T. A. Slaman, *On zeros of Martin-Löf random Brownian motion*, Journal of Logic and Analysis **6** (2014), 1–36.
16. J.-P. Allouche and J. Shallit, *Automatic sequences*, Cambridge, 2003.
17. N. Alon, R. A. Duke, H. Lefmann, V. Rödel, and R. Yuster, *The algorithmic aspects of the regularity lemma*, Journal of Algorithms **16** (1994), 80–109.
18. N. Alon and J. H. Spencer, *The probabilistic method*, third ed., Wiley-Interscience Series in Discrete Mathematics and Optimization, Wiley, 2008.
19. R. Alvir, W. Calvert, G. Goodman, V. Harizanov, J. Knight, A. Morozov, R. Miller, A. Soskova, and R. Weisshaar, *Interpreting a field in its Heisenberg group*, preprint, 2020.
20. J. J. Andrews and M. L. Curtis, *Free groups and handlebodies*, Proceedings of the American Mathematical Society **16** (1965), 192–195.
21. U. Andrews, I. Goldbring, and H. J. Keisler, *Definable closure in randomizations*, Annals of Pure and Applied Logic **166** (2015), 325–341.
22. ———, *Independence in randomizations*, Journal of Mathematical Logic **19** (2019), 1950005.
23. U. Andrews and H. J. Keisler, *Separable models of randomizations*, Journal of Symbolic Logic **80** (2015), 1149–1181.
24. U. Andrews, S. Lempp, J. S. Miller, K. M. Ng, L. San Mauro, and A. Sorbi, *Universal computably enumerable equivalence relations*, The Journal of Symbolic Logic **79** (2014), 60–88.
25. U. Andrews and A. Sorbi, *The complexity of index sets of classes of computably enumerable equivalence relations*, The Journal of Symbolic Logic **81** (2016), 1375–1395.

26. A. Arana, *Logical and semantic purity*, Protosociology **25** (2008), 36–48.
27. A. Arnould and P. Nicole, *The Port Royal Logic*, Gordon, 1861.
28. S. Arora and B. Barak, *Computational complexity*, Cambridge, 2009.
29. Е. А. Асарин and А. В. Покровский, Применение колмогоровской сложности к анализу динамики упровляемых систем, Автоматика и Телемеханика **1** (1986), 25–33.
30. M. Aschenbrenner, A. Dolich, D. Haskell, D. Macpherson, and S. Starchenko, *Vapnik-Chervonenkis density in some theories without the independence property, II*, Notre Dame Journal of Formal Logic **54** (2013), 311–363.
31. ———, *Vapnik-Chervonenkis density in some theories without the independence property, I*, Transactions of the American Mathematical Society **368** (2016), 5889–5949.
32. C. J. Ash and J. F. Knight, *Computable structures and the hyperarithmetical hierarchy*, Studies in Logic and the Foundations of Mathematics, vol. 144, Elsevier, 2000.
33. K. B. Athreya, J. M. Hitchcock, J. H. Lutz, and E. Mayordomo, *Effective strong dimension in algorithmic information and computational complexity*, SIAM Journal on Computing **37** (2007), 671–705.
34. J. Avigad, *Inverting the Furstenberg correspondence*, Discrete and Continuous Dynamical Systems **32** (2012), 3421–3431.
35. J. Avigad, J. Hözl, and L. Serafin, *A formally verified proof of the central limit theorem*, Journal of Automated Reasoning **59** (2017), 389–423.
36. J. Ax, *The elementary theory of finite fields*, Annals of Mathematics **85** (1968), 239–271.
37. L. Babai, *Trading group theory for randomness*, STOC '85: Proceedings of the seventeenth annual ACM symposium on Theory of Computing, 1985, pp. 421–429.
38. A. Baker, *Transcendental number theory*, Cambridge Mathematical Library, Cambridge, 1990.
39. J. T. Baldwin and S. Shelah, *Randomness and semigenericity*, Transactions of the American Mathematical Society **349** (1997), 1359–1376.
40. S. Banach, *Sur le problème de la measure*, Fundamenta Mathematicae **4** (1923), 7–33.
41. A.-L. Barabási and R. Albert, *Emergence of scaling in random networks*, Science **286** (1999), 509–512.
42. G. Barmpalias, D. Cenzer, and C. P. Porter, *The probability of a computable output from a random oracle*, preprint, 2016.
43. ———, *Random numbers as probabilities of machine behavior*, Theoretical Computer Science **673** (2017), 1–18.
44. George Barmpalias and Andrew Lewis-Pye, *Differences of halting probabilities*, preprint, 2016.
45. L. Barreira, *Dimension and recurrence in hyperbolic dynamics*, Progress in Mathematics, no. 272, Birkhäuser, 2008.
46. L. Bartholdi, *Counting paths in groups*, L'Enseignement Mathématique **45** (1999), 83–131.
47. N. A. Bazhenov and B. S. Kalmurzaev, *On dark computably enumerable equivalence relations*, Siberian Mathematical Journal **59** (2018), 22–30.
48. V. Becher, Y. Bugeaud, and T. A. Slaman, *On simply normal numbers to different bases*, Mathematische Annalen **364** (2016), 125–150.
49. V. Becher, O. Carton, and P. A. Heiber, *Normality and automata*, Journal of Computer and System Sciences **81** (2015), 1592–1613.
50. V. Becher and P. A. Heiber, *Normal numbers and finite automata*, Theoretical Computer Science **477** (2013), 109–116.
51. J. Beck, *An algorithmic approach to the Lovász local lemma I*, Random Structures and Algorithms **2** (1991), 343–365.
52. H. Becker and A. S. Kechris, *Borel actions of Polish groups*, Bulletin of the American Mathematical Society **28** (1993), 334–341.
53. O. Becker, A. Lubotzky, and A. Thom, *Stability and invariant random subgroups*, Duke Mathematical Journal **168** (2019), 2207–2234.
54. S. Ben-David, D. Pál, and S. Shalev-Shwartz, *Agnostic online learning*, Conference on Learning Theory (COLT), 2009.
55. I. Ben Yaacov, *Schrödinger's cat*, Israel Journal of Mathematics **153** (2006), 157–191.
56. ———, *Continuous and random Vapnik-Chervonenkis classes*, Israel Journal of Mathematics **173** (2009), 309–333.

57. ———, *On theories of random variables*, Israel Journal of Mathematics **194** (2013), 957–1012.
58. I. Ben Yaacov, A. Berenstein, C. W. Henson, and A. Usvyatsov, *Model theory for metric structures*, Model theory with applications to algebra and analysis, vol. 2, London Mathematical Socieity Lecture Note Series, no. 350, Cambridge, 2008, pp. 315–429.
59. I. Ben Yaacov and H. J. Keisler, *Randomizations of models as metric structures*, Confluentes Mathematici **1** (2009), 197–223.
60. I. Ben Yaacov and A. P. Pedersen, *A proof of completeness for continuous first-order logic*, Journal of Symbolic Logic (2010), 168–190.
61. I. Ben Yaacov and A. Usvyatsov, *Continuous first order logic and local stability*, Transactions of the American Mathematical Society **362** (2010), 5213–5259.
62. C. Bernardi and A. Sorbi, *Classifying positive equivalence relations*, The Journal of Symbolic Logic **48** (1983), 529–538.
63. A. Beros, *Learning theory in the arithmetic hierarchy*, Journal of Symbolic Logic **79** (2014), 908–927.
64. Ö. Beyarslan, *Random hypergraphs in pseudofinite fields*, Journal of the Institute of Mathematics of Jussieu **9** (2010), 29–47.
65. L. Bienvenu, *Game-theoretic approaches to randomness: unpredictability and stochasticity*, Ph.D. thesis, Université de Provence, 2008.
66. L. Bienvenu, A. Day, M. Hoyrup, I. Mezhirov, and A. Shen, *A constructive version of Birkhoff's ergodic theorem for Martin-Löf random points*, Information and Computation **210** (2012), 21–30.
67. C. M. Bishop, *Pattern recognition and machine learning*, Information Science and Statistics, Springer, 2006.
68. L. Blum and M. Blum, *Toward a mathematical theory of inductive inference*, Information and control **28** (1975), 125–155.
69. A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, *Learnability and the Vapnik-Chervonenkis dimension*, Journal of the ACM **36** (1989), 929–965.
70. B. Bollobás, *Random graphs*, 2nd ed., Cambridge Studies in Advanced Mathematics, no. 73, Cambridge, 2001.
71. G. Boole, *An investigation of the laws of thought, on which are founded the mathematical theories of logic and probabilities*, Macmillan, 1851.
72. W. W. Boone, *Certain simple, unsolvable problems of group theory V, VI*, Indagationes Mathematicae **60** (1957), 22–27, 227–232.
73. ———, *The word problem*, Proceedings of the National Academy of Sciences of the USA **44** (1958), 1061–1065.
74. M. Borda, *Fundamentals in information theory and coding*, Springer, 2011.
75. A. Borel, *Density properties for certain subgroups of semi-simple groups wihtout compact components*, Annals of Mathematics **72** (1960), 179–188.
76. M. E. Borel, *Les probabilités dénombrables et leurs applications arithmétiques*, Rendiconti del Circolo Matematico di Palermo (1884–1940) **27** (1909), 247–271.
77. C. Borgs, J. T. Chayes, L. Lov'asz, V. T. Sos, and K. Vesztergombi, *Convergent sequences of dense graphs I: Subgraph frequencies, metric properties and testing*, Advances in Mathematics **219** (2008), 1801–1851.
78. C. Borgs, J. T. Chayes, L. Lovász, V. T. Sós, and K. Vesztergombi, *Convergent sequences of dense graphs I: Subgraph frequencies, metric properties and testing*, Advances in Mathematics **219** (2008), 1801–1851.
79. K. H. Borgwardt, *The simplex method: A probabilistic analysis*, Algorithms and Combinatorics, no. 1, Springer, 1987.
80. L. Bowen, *Invariant random subgroups of the free group*, Groups, Geometry, and Dynamics **9** (2015), 891–916.
81. G. Boxall, *NIP for some pair-like theories*, Archive for Mathematical Logic **50** (2011), 353–359.
82. M. Braverman, *Parabolic Julia sets are polynomial time computable*, Nonlinearity **19** (2006), 1383–1401.
83. M. Braverman and M. Yampolsky, *Computability of julia sets*, Algorithms and Computation in Mathematics, no. 23, Springer, 2009.

84. J. Brody and M. C. Laskowski, *Rational limits of Shelah-Spencer graphs*, The Journal of Symbolic Logic **77** (2012), 580–592.
85. Y. Bugeaud, *Distribution modulo one and Diophantine approximation*, Cambridge Tracts in Mathematics, no. 193, Cambridge, 2012.
86. S. Buss and M. Minnes, *Probabilistic algorithmic randomness*, The Journal of Symbolic Logic **78** (2013), 579–601.
87. D. Cai, N. Ackerman, and C. Freer, *An iterative step-function estimator for graphons*, preprint, 2015.
88. W. Calvert, *Metric structures and probabilistic computation*, Theoretical Computer Science **412** (2011), 2766–2775.
89. ———, *PAC learning, VC dimension, and the arithmetic hierarchy*, Archive for Mathematical Logic **54** (2015), 871–883.
90. W. Calvert, V. Harizanov, and A. Shlapentokh, *Turing degrees of isomorphism types of algebraic objects*, The Journal of the London Mathematical Society **75** (2007), 273–286.
91. ———, *Random fields*, preprint, 2020.
92. W. Calvert and J. F. Knight, *Classification from a computable viewpoint*, Bulletin of Symbolic Logic **12** (2006), 191–219.
93. P. J. Cameron, *Transitivity of permutation groups on unordered sets*, Mathematische Zeitschrift **148** (1976), 127–139.
94. C. Carathéodory, *Über das lineare Mass von Punktmengen — eine Verallgemeinerung des Längenbegriffs*, Nachrichten von der Königlichen Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-phisikalisch Klasse **1914** (1914), 404–426.
95. J. Case, S. Jain, and F. Stephan, *Effectivity questions for Kleene's recursion theorem*, Theoretical Computer Science **733** (2018), 55–70.
96. J. Case and C. Lynes, *Machine inductive inference and language identification*, International Colloquium on Automata, Languages, and Programming, 1982, pp. 107–115.
97. T. Ceccherini-Silberstein and M. Coornaert, *Cellular automata and groups*, Springer Monographs in Mathematics, Springer, 2010.
98. D. Cenzer, Π^0_1 *classes in computability theory*, Handbook of Computability, Studies in Logic and the Foundations of Mathematics, no. 140, Elsevier, 1999, pp. 37–85.
99. C. Chabauty, *Limite d'ensembles et géométrie des nombres*, Bulletin de la S. M. F. **78** (1950), 143–151.
100. G. J. Chaitin, *A theory of program size formally identical to information theory*, Journal of the Association for Computing Machinery **22** (1975), 329–340.
101. ———, *Algorithmic information theory*, Cambridge Tracts in Theoretical Computer Science, no. 1, Cambridge, 1987.
102. A. S. Charles, *Interpreting deep learning: The machine learning Rorschach test?*, preprint, 2018.
103. H. Chase and J. Freitag, *Model theory and machine learning*, preprint, 2018.
104. Z. Chatzidakis, *Théorie des modèles des corps valués*, lecture notes, 2008.
105. Z. Chatzidakis, L. van den Dries, and A. Macintyre, *Definable sets over finite fields*, Journal für die reine und angewandte Mathematik **427** (1992), 107–135.
106. G. Cherlin and E. Hrushovski, *Finite structures with few types*, Annals of Mathematics Studies, no. 152, Princeton University Press, 2003.
107. A. Chernikov and S. Starchenko, *Definable regularity lemmas for NIP hypergraphs*, Quarterly Journal of Mathematics **72** (2021), 1401–1433.
108. N. Chomsky, *Three models for the description of language*, IRE Transactions on Information Theory **2** (1956), 113–124.
109. ———, *Knowledge of language: Its nature, origin, and use*, Convergence, Praeger Scientific, 1986.
110. F. Chung, *On concentrators, superconcentrators, generalizers, and nonblocking networks*, The Bell System Technical Journal **58** (1978), 1765–1777.
111. F. Chung and L. Lu, *Complex graphs and networks*, CBMS Regional Conference Series in Mathematics, no. 107, American Mathematical Society, 2006.
112. F. Chung, L. Lu, T. G. Dewey, and D. J. Galas, *Duplication models for biological networks*, Journal of Computational Biology **10** (2003), 677–687.

113. B. Cisma, D. D. Dzhafarov, D. R. Hirschfeldt, C. G. Jockusch, R. Solomon, and L. B. Westrick, *The reverse mathematics of Hindman's theorem for sums of exactly two elements*, Computability **8** (2019), 253–263.
114. K. J. Compton, *Laws in logic and combinatorics*, Algorithms and Order, NATO ASI Series C, no. 255, Kluwer, 1989, pp. 353–383.
115. G. Conant and A. Pillay, *Pseudofinite groups and VC-dimension*, preprint, 2018.
116. A. Condon, *The complexity of stochastic games*, Information and Computation **96** (1992), 203–224.
117. C. T. Conley, A. S. Kechris, and B. D. Miller, *Stationary probability measures and topological realizations*, Israel Journal of Mathematics **198** (2013), 333–345.
118. C. T. Conley and B. D. Miller, *Measure reducibility of countable Borel equivalence relations*, Annals of Mathematics **185** (2017), 347–402.
119. D. Conlon and J. Fox, *Bounds for graph regularity and removal lemmas*, Geometric and Functional Analysis **22** (2012), 1191–1256.
120. A. Connes and B. Weiss, *Property T and asymptotically invariant sequences*, Israel Journal of Mathematics **37** (1980), 209–210.
121. S. D. Conte and C. de Boor, *Elementary numerical analysis*, 3rd ed., International Series in Pure and Applied Mathematics, McGraw-Hill, 1980.
122. O. Cooley, W. Fang, D. Del Giudice, and M. Kang, *Subcritical random hypergraphs, high-order components, and hypertrees*, 2019 Proceedings of the Sixteenth Workshop on Analytic Algorithmics and Combinatorics (ANALCO), 2019, pp. 111–118.
123. M. Coornaert, *Topoligcal dimension and dynamical systems*, Universitext, Springer, 2015.
124. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*, 3rd ed., MIT Press, 2009.
125. R. T. Cox, *Probability, frequency, and reasonable expectation*, American Journal of Physics **14** (1946), 1–13.
126. B. F. Csima, V. S. Harizanov, R. Miller, and A. Montalban, *Computability of Fraïssé limits*, Journal of Symbolic Logic **76** (2011), 66–93.
127. R. W. R. Darling and J. R. Norris, *Structure of large random hypergraphs*, The Annals of Applied Probability **15** (2005), 125–152.
128. S. Ben David, P. Hrubes, S. Moran, A. Shpilka, and A. Yehudayoff, *Learnability can be undecidable*, Nature Machine Intellignece **1** (2019), 44–48.
129. A. P. Dawid, *Probability, causality, and the empirical world: a Bayes-de Finetti-Popper-Borel synthesis*, Statistical Science **10** (2004), 44–57.
130. B. de Finetti, *Funzione caratteristica di un fenomeno aleatorio*, Memorie della R. Accademia dei Lincei **4** (1930), 86–133.
131. ———, *Foresight: its logical laws, its subjective sources*, Breakthroughs in Statisitcs, Springer, 1992, Translated by H. E. Kyberg, Jr.; Original published 1937, pp. 134–174.
132. ———, *A translation of 'The characteristic function of a random phenomenon' by Bruno de Finetti*, D. Alvarez-Melis and T. Broderick, translators. arXiv:1512.01229, 2015.
133. K. de Leeuw, E. F. Moore, C. E. Shannon, and N. Shapiro, *Computability by probabilistic machines*, Automata Studies, Annals of Mathematics Studies, no. 34, Princeton, 1956, pp. 183–212.
134. M. Dehn, *Über unendliche diskontinuierliche Gruppen*, Mathematische Annalen **71** (1911), 116–144.
135. ———, *Transformation der kurven auf zweiseitigen flächen*, Mathematische Annalen **72** (1912), 413–421.
136. A. P. Dempster, *Upper and lower probabilities induced by a multivalued mapping*, The Annals of Mathematical Statistics **38** (1967), 325–339.
137. O. Demuth, *On constructive pseudonumbers*, Commentationes Mathematicae Universitatis Carolinae **16** (1975), 315–331.
138. M. Detlefsen and A. Arana, *Purity of methods*, Philosophers' Imprint **11** (2011), 1–20.
139. P. Diaconis and S. Janson, *Graph limits and exchangeable random graphs*, Rendiconti di Matematica, Serie VII **28** (2008), 33–61.
140. A. Ditzen, *Definable equivalence relations on Polish spaces*, Ph.D. thesis, California Institute of Technology, 1992.
141. A. Dolich, D. Lippel, and J. Goodrick, *dp-minimal theories: basic facts and examples*, Notre Dame Journal of Formal Logic **52** (2011), 267–288.

142. M. D. Donsker, *An invariance principle for certain probability limit theorems*, Memoirs of the American Mathematical Society **6** (1951), 1–12.
143. R. Dougherty, S. Jackson, and A. S. Kechris, *The structure of hyperfinite Borel equivalence relations*, Transactions of the American Mathematical Society **341** (1994), 193–225.
144. R. G. Downey and E. J. Griffiths, *Schnorr randomness*, The Journal of Symbolic Logic **69** (2004), 533–554.
145. R. G. Downey and D. R. Hirschfeldt, *Algorithmic randomness and complexity*, Theory and Applications of Computability, Springer, 2010.
146. R. G. Downey, C. G. Jockusch Jr., and P. E. Schupp, *Asymptotic density and computably enumerable sets*, Journal of Mathematical Logic **13** (2013), 1350005.
147. A. Dudko and M. Yampolsky, *On computational complexity of Cremer Julia sets*, Fundamenta Mathematicae **252** (2021), 343–353.
148. R. M. Dudley, *Central limit theorems for empirical measures*, The annals of probability **6** (1978), 899–929.
149. J.-L. Duret, *Les corps faiblement algebriquement clos non separablement clos ont la properiete d'independance*, Model Theory of Algebra and Arithmetic, Lecture Notes in Mathematics, no. 834, Springer, 1980, pp. 136–162.
150. M. Džamonja and I. Tomašić, *Graphons arising from graphs definable over finite fields*, preprint, 2017.
151. P. D. Eastman, *Are you my mother?*, Random House, 1960.
152. G. Edgar, *Measure, topology, and fractal geometry*, second ed., Undergraduate Texts in Mathematics, Springer, 2008.
153. H. G. Eggleston, *Sets of fractional dimensions which occur in some problems of number theory*, Proceedings of the London Mathematical Society **54** (1952), 42–93.
154. K. Eickmeyer and M. Grohe, *Randomisation and derandomisation in descriptive complexity theory*, Logical Methods in Computer Science **7** (2011), 1–24.
155. G. Elek and B. Szegedy, *A measure-theoretica approach to the theory of dense hypergraphs*, Advances in Mathematics **231** (2012), 1731–1772.
156. R. Elwes, *Asymptotic classes of finite structures*, Journal of Symbolic Logic **72** (2007), 418–438.
157. H. B. Enderton, *A mathematical introduction to logic*, Academic Press, 1972.
158. I. Epstein, *Orbit inequivalent actions of non-amenable groups*, preprint, 2008.
159. P. Erdős, D. J. Kleitman, and B. L. Rothschild, *Asymptotic enumeration of K_n-free graphs*, Colloquio Internazionale sulle Teorie Combinatorie (Rome, 1973), vol. 2, Acad. Naz. Lincei, 1976, pp. 19–27.
160. P. Erdős and L. Lovász, *Problems and results on 3-chromatic hypergraphs and some related questions*, Infinite and Finite Sets, vol. II, Colloq. Math. Soc. János Bolyai, no. 10, North-Holland, 1975, pp. 609–627.
161. P. Erdős and A. Rényi, *On the evolution of random graphs*, Matematikai Kutató Intézet Közelményei **A** (1960), 17–60.
162. P. Erdős and A. Rényi, *On random graphs I*, Publicationes Mathematicae Debrecen **6** (1959), 290–297.
163. Ю. Л. Ершов, Позитивные Еквивалентности, Алгебра и Логика **10** (1971), 620–650.
164. R. Fagin, *Generalized first-order spectra and polynomial-time recognizable sets*, SIAM–AMS Proceedings, vol. 7, 1974.
165. ———, *Probabilities on finite models*, The Journal of Symbolic Logic **41** (1976), 50–58.
166. S. Fajardo and H. J. Keisler, *Model theory of stochastic processes*, Lecture Notes in Logic, no. 14, A K Peters, 2002.
167. K. Falconer, *Fractal geometry: Mathematical foundations and applications*, 2nd ed., Wiley, 2003.
168. U. Felgner, *Pseudo-endliche gruppen*, Jahrbuch der Kurt-Gödel-Gesellschaft **3** (1990), 94–108.
169. E. Fokina, V. Harizanov, and D. Turetsky, *Computability-theoretic categoricity and Scott families*, preprint, 2019.
170. G. B. Folland, *Real analysis*, 2nd ed., Pure and Applied Mathematics, Wiley, 1999.
171. M. Foreman, D. J. Rudolph, and B. Weiss, *The conjugacy problem in ergodic theory*, Annals of Mathematics **173** (2011), 1529–1586.

172. M. Foreman and B. Weiss, *An anti-classification theorem for ergodic measure presrving transormations*, Journal of the European Mathematical Society **6** (2004), 277–292.
173. W. Fouché, *Arithmetical representations of Brownian motion I*, The Journal of Symbolic Logic **65** (2000), 421–442.
174. ———, *Martin-Löf randomness, invariant measures and countable homogeneous structures*, Theory of Computing Systems **52** (2013), 65–79.
175. W. L. Fouché, *Algorithmic randomness and Ramsey properties of countable homogeneous structures*, Logic, language, information, and computation, Lecture Notes in Computer Science, no. 7456, Springer, 2012, pp. 246–256.
176. R. Fraïssé, *Sur l'extension aux relations de quelques propriétés des ordres*, Annales scientifiques de l'É.N.S. **71** (1954), 363–388.
177. J. N. Y. Franklin, N. Greenberg, J. S. Miller, and K. M. Ng, *Martin-Löf random points satisfy Birkhoff's ergodic theorem for effectively closed sets*, Proceedings of the American Mathematical Society **140** (2012), 3623–3628.
178. J. N. Y. Franklin and C. P. Porter (eds.), *Algorithmic randomness*, Lecture Notes in Logic, no. 50, Cambridge University Press, 2020.
179. C. Freer, *Computable de Finetti measures*, Annals of Pure and Applied Logic **163** (2012), 530–546.
180. M. D. Fried and M. Jarden, *Field arithmetic*, 2nd ed., Ergebnisse der Mathematik und ihrer Grenzgebiete, vol. 11, Springer, 2005.
181. H. Friedman and L. Stanley, *A Borel reducibility theory for classes of countable structures*, Journal of Symbolic Logic **54** (1989), 894–914.
182. A. Frieze and R. Kannan, *Quick approximation to matrices and applications*, Combinatorica **19** (1999), 175–220.
183. A. Furman, *What is a stationary measure?*, Notices of the American Mathematical Soceity **58** (2011), 1276–1277.
184. H. Furstenberg, *A Poisson formula for semi-simple Lie groups*, Annals of Mathematics **77** (1963), 335–386.
185. ———, *Disjointness in ergodic theory, minimal sets, and a problem in Diophantine approximation*, Mathematical Systems Theory **1** (1967), 1–49.
186. ———, *A note on Borel's density theorem*, Proceedings of the American Mathematical Society **55** (1976), 209–212.
187. ———, *Ergodic behavior of diagonal measures and a theorem of Szemerédi on arithmetic progressions*, Journal D'Analyse Mathématique **31** (1977), 204–256.
188. S. Gaal and L. Gál, *The discrepancy of the sequence* $\{(2^n x)\}$, Indagationes Mathematicae **26** (1964), 129–143.
189. H. Gaifman, *Concerning measures in first order calculi*, Israel Journal of Mathematics **2** (1964), 1–18.
190. S. Gao, *Invariant descriptive set theory*, Pure and Applied Mathematics, CRC Press, 2009.
191. S. Gao and P. Gerdes, *Computably enumerable equivalence relations*, Studia Logica **67** (2001), 27–59.
192. W. I. Gasarch, *The P=?NP poll*, ACM SIGACT News **33** (2002), 34–47.
193. H. Geffner and J. Pearl, *A framework for reasoning with defaults*, Tech. Report R-94, Cognitive Systems Laboratory, UCLA, 1987.
194. D. Geiger and J. Pearl, *Logical and algorithmic properties of conditional independence and graphical models*, Annals of Statistics **21** (1993), 2001–2021.
195. T. Gelander, *Lecture notes on invariant random subgroups and lattices in rank one and higher rank*, preprint, 2015.
196. ———, *Kazhdan-Margulis theorem for invariant random subgroups*, Advances in Mathematics **327** (2018), 47–51.
197. G. Ghoshal, V. Zlatić, G. Caldarelli, and M. E. J. Newman, *Random hypergraphs and their applications*, Physical Review E **79** (2009), 066118–1–066118–10.
198. J. Gill, *Computational complexity of probabilistic Turing machines*, SIAM Journal on Computing **6** (1977), 675–695.
199. E. Glasner and B. Wiess, *Minimal actions of the group* $\mathbb{S}(\mathbb{Z})$ *of permutations of the integers*, Geometric and Functional Analysis **12** (2002), 964–988.
200. Yu. V. Glebskii, D. I. Kogan, M. I. Kogonkii, and V. A. Talanov, *Volume and fraction of satisfiability of formulas of the lower predicate calculus*, Kibernetika (Kiev) (1969), 17–27.

201. E. Mark Gold, *Language identification in the limit*, Information and Control **10** (1967), 447–474.
202. I. Goldbring and B. Hart, *Computability and the Connes Embedding Problem*, Bulletin of Symbolic Logic **22** (2016), 238–248.
203. I. Goldbring and V. C. Lopes, *Pseudofinite and pseudocompact metric structures*, Notre Dame Journal of Formal Logic **56** (2015), 493–510.
204. I. Goldbring and H. Towsner, *An approximate logic for measures*, Israel Journal of Mathematics **199** (2014), 867–913.
205. O. Goldreich, *A primer on pseudorandom generators*, University Lecture Series, no. 55, American Mathematical Society, 2010.
206. O. Goldreich, S. Micali, and A. Wigderson, *Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems*, Journal of the Association for Computing Machinery **38** (1991), 691–729.
207. S. Goldwasser, S. Micali, and C. Rackoff, *The knowledge complexity of interactive proof systems*, STOC '85: Proceedings of the seventeenth annual ACM symposium on Theory of Computing, 1985, pp. 291–304.
208. D. Gorenstein, *Classifying the finite simple groups*, Bulletin of the American Mathematical Society **14** (1986), 1–98.
209. N. Greenberg, J. S. Miller, A. Shen, and L. Brown Westrick, *Dimension 1 sequences are close to randoms*, Theoretical Computer Science **705** (2018), 99–112.
210. M. Gromov, *Random walk in random groups*, Geometric and Functional Analysis **13** (2003), 73–146.
211. Y. Guivarc'h, *Sur la loi des grands nombres et le rayon spectral d'une marche aléatoire*, Astérisque **74** (1980), 47–98.
212. A. Günaydin and P. Hieronymi, *Dependent pairs*, Journal of Symbolic Logic **76** (2011), 377–390.
213. Y. Gurevich and P. H. Schmitt, *The theory of ordered Abelian groups does not have the independence property*, Transactions of the American Mathematical Society **284** (1984), 171–182.
214. I. Hacking, *The emergence of probability*, 2nd ed., Cambridge, 2006.
215. R. Haenni and N. Lehmann, *Probabilistic artumentation systems: a new perspecive on the Dempster-Shafer theory*, International Journal of Intelligent Systems **18** (2003), 93–106.
216. R. Haenni, J.-W. Romeijn, G. Wheeler, and J. Williamson, *Probabilistic logics and probabilistic networks*, Synthese Library, vol. 350, Springer, 2011.
217. T. Hailperin, *Sentential probability logic*, Lehigh University Press, 1996.
218. J. Y. Halpern, *A counterexample to theorems of Cox and Fine*, Journal of Artificial Intelligence Research **10** (1999), 67–85.
219. ———, *Cox's theorem revisited*, Journal of Artifical Intelligence Research **11** (1999), 429–435.
220. ———, *Reasoning about uncertainty*, MIT Press, 2003.
221. V. S. Harizanov, *Inductive inference systems for learning classes of algorithmically generated sets and structures*, Induction, Algorithmic Learning Theory, and Philosophy (M. Friend, N. B. Goethe, and V. S. Harizanov, eds.), Logic, Epistemology, and the Unity of Science, no. 9, Springer, 2007, pp. 27–54.
222. V. S. Harizanov and F. Stephan, *On the learnability of vector spaces*, Journal of Computer and System Sciences **73** (2007), 109–122.
223. L. A. Harrington, A. S. Kechris, and A. Louveau, *A Glimm-Effros dichotomy for Borel equivalence relations*, Journal of the American Mathematical Society **3** (1990), 903–928.
224. M. Harrison-Trainor, B. Khoussainov, and D. Turetsky, *Effective aspects of algorithmically random structures*, Computability **8** (2019), 359–375.
225. H. Hatami and S. Norine, *The entropy of random-free graphons and properties*, Combinatorics, Probability, and Computing **22** (2013), 517–526.
226. F. Hausdorff, *Dimension und äußeres Maß*, Mathematische Annalen **79** (1918), 157–179.
227. C. W. Henson, *A family of countable homogeneous graphs*, Pacific Journal of Mathematics **38** (1971), 69–83.
228. P. Hieronymi and T. Nell, *Distal and non-distal pairs*, Journal of Symbolic Logic **82** (2017), 375–383.

229. D. Hilbert, *Lectures on the foundations of geometry, 1891–1902*, vol. 1, Springer, 2004, translation due to V. Pambuccian, Fragments of Euclidean and hyperbolic geometry, *Scientia mathematicae Japonicae* 53 (2001) pp. 361–400.
230. J. Hintikka and G. Sandu, *Game-theoretical semantics*, Handbook of Logic and Language, Amsterdam, 1997, pp. 361–410.
231. D. R. Hirschfeldt, C. G. Jockusch Jr., T. H. McNicholl, and P. E. Schupp, *Asymptotic density and the coarse computability bound*, Computability **5** (2016), 13–27.
232. D. R. Hirschfeldt, B. Khoussainov, R. A. Shore, and A. M. Slinko, *Degree spectra and computable dimensions in algebraic structures*, Annals of Pure and Applied Logic **115** (2002), 71–113.
233. J. M. Hitchcock, *Correspondence principles for effective dimensions*, Theory of Computing Systems **38** (2005), 559–571.
234. J. M. Hitchcock and J. H. Lutz, *Why computational complexity requires stricter martingales*, Theory of Computing Systems **39** (2006), 277–296.
235. G. Hjorth, *Classification and orbit equivalence relations*, Mathematical Surveys and Monographs, vol. 75, American Mathematical Society, 2000.
236. ———, *On invariants for measure preserving transformations*, Fundamenta Mathematicae **169** (2001), 51–84.
237. ———, *A converse to Dye's Theorem*, Transactions of the American Mathematical Society **357** (2005), 3083–3103.
238. ———, *Glimm-Effros for coanalytic equivalence relations*, Journal of Symbolic Logic **74** (2009), 402–422.
239. G. Hjorth and A. S. Kechris, *Analytic equivalence relations and Ulm-type classifications*, Journal of Symbolic Logic **60** (1995), 1273–1300.
240. W. Hodges, *What is a structure theory?*, Bulletin of the London Mathematical Society **19** (1987), 209–237.
241. ———, *Groups in pseudofinite fields*, Model theory of groups and automorphism groups, London Mathematical Society Lecture Note Series, no. 244, Cambridge University Press, 1997, pp. 90–109.
242. B. Host and B. Kra, *Nilpotent structures in ergodic theory*, Mathematical Surveys and Monographs, vol. 236, American Mathematical Society, 2018.
243. E. Hrushovski, *Pseudo-finite fields and related structures*, Model Theory and Applications, Quaderni di Matematica, no. 11, Aracne, 2002, pp. 151–212.
244. ———, *Stable group theory and approximate subgroups*, Journal of the American Mathematical Society **25** (2012), 189–243.
245. E. Hrushovski, Y. Peterzil, and A. Pillay, *Groups, measures, and the NIP*, Journal of the American Mathematical Society **21** (2008), 563–596.
246. E. Hrushovski and A. Pillay, *Groups definable in local fields and pseudo-finite fields*, Israel Journal of Mathematics **85** (1994), 203–262.
247. ———, *Definable subgroups of algebraic groups over finite fields*, Journal für die reine und angewandte Mathematik **462** (1995), 69–91.
248. T. W. Hungerford, *Algebra*, Graduate Texts in Mathematics, no. 73, Springer, 1974.
249. N. Immerman, *Descriptive complexity*, Graduate Texts in Computer Science, Springer, 1999.
250. S. Jain, D. Osherson, J. Royer, and A. Sharma, *Systems that learn: An introduction to learning theory*, 2nd ed., Learning, Development, and Conceptual Change, MIT Press, 1999.
251. S. Janson, *Graphons, cut norm, and distance, couplings and rearrangements*, NYJM Monographs, no. 4, New York Journal of Mathematics, 2013.
252. R. C. Jeffrey, *The logic of decision*, 2nd ed., University of Chicago Press, 1983.
253. Z. Ji, A. Natarajan, T. Vidick, J. Wright, and H. Yuen, $\mathbf{MIP}^* = \mathbf{RE}$, arXiv:2001.04383, 2020.
254. C. G. Jockusch Jr and P. E. Schupp, *Generic computability, turing degrees, and asymptotic density*, Journalof the London Mathematical Society, 2nd Series **85** (2012), 472–490.
255. V. Kaimanovich, I. Kapovich, and P. Schupp, *The subadditive ergodic theorem and generic stretching factors for free group automorphisms*, Israel Journal of Mathematics **157** (2007), 1–46.
256. ———, *The subadditive ergodic theorem and generic stretching factors for free group automorphisms*, Israel Journal of Mathematics **157** (2007), 1–46.

257. O. Kallenberg, *Foundations of modern probability*, Probability and its Applications, Springer, 1997.
258. ———, *Probabilistic symmetries and invariance principles*, Probability and its Applications, Springer, 2005.
259. E. R. Kandel, J. H. Schwartz, T. M. Jessell, S. A. Siegelbaum, and A. J. Hudspeth, *Principles of neural science*, fifth ed., McGraw-Hill, 2013.
260. I. Kaplansky, *Infinite Abelian groups*, revised ed., University of Michigan Press, 1969.
261. I. Kapovich, A. Myasnikov, P. Schupp, and V. Shpilrain, *Generic-case complexity, decision problems in group theory, and random walks*, Journal of Algebra **264** (2003), 665–694.
262. M. Karpinski and A. Macintyre, *Approximating volumes and integrals in o-minimal and p-minimal theories*, Connections between model theory and algeraic and analytic geometry, Quaderni di Matematica, no. 6, Arcane, 2000, pp. 149–177.
263. S. Kauffman, *Homeostasis and differentiation in random genetic control networks*, Nature **224** (1969), 177–178.
264. ———, *Metabolic stability and epigenesis in randomly constructed genetic nets*, Journal of Theoretical Biology **22** (1969), 437–467.
265. ———, *The large scale structure and dynamics of gene control circuits: an ensemble approach*, Journal of Theoretical Biology **44** (1974), 167–190.
266. ———, *The origins of order*, Oxford University Press, 1993.
267. S. M. Kautz, *Degrees of random sets*, Ph.D. thesis, Cornell University, 1991.
268. D. Kazhdan and G. Margulis, *A proof of Selberg's hypothesis*, Matematicheskii Sbornik **75** (1968), 163–168.
269. M. J. Kearns and U. V. Vazirani, *An introduction to computational learning theory*, MIT Press, 1994.
270. A. S. Kechris and B. D. Miller, *Topics in orbit equivalence*, Lecture Notes in Mathematics, no. 1852, Springer, 2004.
271. A. S. Kechris, V. G. Pestov, and S. Todorcevic, *Fraïssé limits, Ramsey theory, and topological dynamics of automorphism groups*, Geometric and Functional Analysis **15** (2005), 106–189.
272. ———, *Fraïssé limits, Ramsey theory, and topological dynamics of automorphism groups*, Geometric and Functional Analysis **15** (2005), 106–189.
273. A. S. Kechris and R. D. Tucker-Drob, *The complexity of classification results in ergodic theory*, Appalachian Set Theory 2006–2012, London Mathematical Society Lecture Note Series, no. 406, Cambridge, 2013, pp. 265–299.
274. H. J. Keisler, *Randomizing a model*, Advances in Mathematics **143** (1999), 124–158.
275. ———, *Probability quantifiers*, Model-Theoretic Logics, Perspectives in Logic, Cambridge University Press, 2016, Originally published 1985, pp. 509–556.
276. J. M. Keynes, *A treatise on probability*, MacMillan, 1921.
277. A. I. Khinchine, *Mathematical foundations of information theory*, Dover, 1957.
278. B. Khoussainov, *A quest for algorithmically random infinite structures*, Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), ACM, 2014, Article No. 56.
279. ———, *A quest for algorithmically random infinite structures, II*, Logical foundations of computer science, Lecture Notes in Computer Science, no. 9537, Springer, 2016, pp. 159–173.
280. H. Ki and T. Linton, *Normal numbers and subsets of* $\mathbb{N}$ *with given densities*, Fundamenta Mathematicae **144** (1994), 163–179.
281. S. Kiefer, R. Mayr, M. Shirmohammadi, and D. Wojtczak, *Strong determinacy of countable stochastic games*, Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science, 2017.
282. J. H. Kim, O. Pikhurko, J. Spencer, and O. Verbitsky, *How complex are random graphs in first order logic?*, Random Structures & Algorithms **26** (2005), 119–145.
283. J. F. C. Kingman, *The ergodic theory of subadditive stochastic processes*, Journal of the Royal Statistical Soceity, Series B (Methodological **30** (1968), 499–510.
284. V. Klee and G. J. Minty, *How good is the simplex algorithm?*, Inequalities, III, Academic Press, 1972, pp. 159–175.
285. J. F. Knight, A. Pillay, and C. Steinhorn, *Definable sets in ordered structures II*, Transactions of the American Mathematical Society **295** (1986), 593–605.

286. D. E. Knuth, *The art of computer programming: Seminumerical algorithms*, 3rd ed., vol. 2, Pearson, 1998.
287. P. G. Kolaitis and M. Y. Vardi, *0–1 laws and decision problems for fragments of second-order logic*, Information and Computation **87** (1990), 302–338.
288. ———, *Infinitary logics and 0–1 laws*, Information and Computation **98** (1992), 258–294.
289. A. N. Kolmogorov, *Foundations of the theory of probability*, Chelsea, 1950.
290. А. Н. Колмогоров, Три подхода к определению понятия ⟨⟨ количество информатсии ⟩⟩, Проблемы передачи информации **1** (1965), 3–11.
291. B. Kra, *Commentary on 'Ergodic theory of amenable group actions': old and new*, Bulletin of the American Mathematical Society **55** (2018), 343–345.
292. P. N. Kryloff and N. Bogoliouboff, *La théorie générale de la mesure dans son application à l'étude des systémes dynamiques de la mécanique non linéaire*, Annals of Mathematics **38** (1937), 65–113.
293. M. H. Kutner, C. J. Nachtsheim, J. Neter, and W. Li, *Applied linear statistical models*, fifth ed., McGraw-Hill Irwin Series: Operations and Decision Sciences, McGraw-Hill, 2005.
294. A. Kučera and T. Slaman, *Randomnuess and recursive enumerability*, SIAM Journal on Computing **31** (2001), 199–211.
295. H. Lädesmäki, S. Hautaniemi, I. Shmulevich, and O. Yli-Harja, *Relationships between probabilistic Boolean networks and dynamic Bayesian networks as models of gene regulatory networks*, Signal Processing **86** (2006), 814–834.
296. H. Lädesmäki, I. Shmulevich, and O. Yli-Harja, *On learning gene regulatory networks under the Boolean network model*, Machine Learning **52** (2003), 147–167.
297. J. C. Lagarias, *The ultimate challenge: The* $3x+1$ *problem*, American Mathematical Society, 2010.
298. S. Lang and A. Weil, *Number of points of varieties in finite fields*, American Journal of Mthematics **76** (1954), 819–827.
299. M. C. Laskowski, *A simpler axiomatization of the Shelah-Spencer almost sure theories*, Israel Journal of Mathematics **161** (2007), 157–186.
300. S. L. Lauritzen and N. Wermuth, *Graphical models for associations between variables, some of which are qualitative and some quantitative*, The Annals of Statistics **17** (1989), 31–57.
301. Y. LeCun, Y. Bengio, and G. Hinton, *Deep learning*, Nature **521** (2015), 436–444.
302. M. Ledoux, *The concentration of measure*, Mathematical Surveys and Monographs, no. 89, American Mathematical Society, 2001.
303. Л. А. Левин, Законы сохранения (невозрастания) информации и вопросы обоснования теории вероятностей, Проблемы передачи информации **10** (1974), 30–35.
304. M. B. Levin, *On the discrepancy estimate of normal numbers*, Acta Arithmetica **88** (1999), 99–111.
305. M. Li, J. Tromp, and P. Vitányi, *Sharpening Occam's razor*, Information Processing Letters **85** (2003), 267–274.
306. Ming Li and Paul Vitányi, *An introduction to Kolmogorov complexity and its applications*, third ed., Texts in Computer Science, Springer, 2008.
307. N. Littlestone, *Learning quickly when irrelevant attributes abound: a new linear-threshold algorithm*, Machine Learning **2** (1988), 285–318.
308. L. Lovász, *Large networks and graph limits*, American Mathematical Society Colloquium Publications, vol. 60, American Mathematical Society, 2012.
309. L. Lovász and B. Szegedy, *Limits of dense graph sequences*, Journal of Combinatorial Theory, Series B **96** (2006), 933–957.
310. D. Loveland, *A new interpretation of the von Mises' concept of random sequence*, Zeitschrift fur mathematische Logik und Grundlagen dr Mathematik **12** (1966), 279–294.
311. A. Lubotzky and B. Weiss, *Groups and expanders*, Expanding Graphs, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 10, American Mathematical Society, 1993, pp. 95–109.
312. T. Luczak and J. Spencer, *When does the zero-one law hold?*, Journal of the American Mathematical Society **4** (1991), 451–468.
313. C. Lund, L. Fortnow, H. Karloff, and N. Nisan, *Algebraic methods for interactive proof systems*, Journal of the Association for Computing Machinery **39** (1992), 859–868.
314. J. H. Lutz, *Dimension in complexity classes*, SIAM Journal on Computing **32** (2003), 1236–1259.

315. R. Lyons and Y. Peres, *Probability on trees and networks*, Cambridge Series in Statistical and Probabilistic Mathematics, no. 42, Cambridge, 2016.
316. D. Macpherson and C. Steinhorn, *One-dimensional asymptotic classes of finite structures*, Transactions of the American Mathematical Society **380** (2008), 411–448.
317. ———, *Definability in classes of finite structures*, Finite and algorithmic model theory, London Mathematical Society Lecture Note Series, no. 379, Cambridge, 2011, pp. 140–176.
318. D. Macpherson and K. Tent, *Pseudofinite groups with NIP theory and definability in finite simple groups*, Groups and Model Theory, Contemporary Mathematics, no. 576, American Mathematical Society, 2012, pp. 255–267.
319. M. Malliaris and A. Pillay, *The stable regularity lemma revisited*, Proceedings of the American Mathematical Society **144** (2016), 1761–1765.
320. M. Malliaris and S. Shelah, *Regularity lemmas for stable graphs*, Transactions of the American Mathematical Society **366** (2014), 1551–1585.
321. ———, *Notes on the stable regularity lemma*, Bulletin of Symbolic Logic **27** (2021), 415–425.
322. V. W. Marek and M. Truszczyński, *Nonmonotonic logic*, Artificial Intelligence, Springer, 1993.
323. G. A. Margulis, *Discrete subgroups of semisimple Lie groups*, Ergebnisse der Mathematik un ihrer Grenzgebiete, 3. Folge, no. 17, Springer, 1991.
324. D. Marker, *Model theory*, Graduate Texts in Mathematics, no. 217, Springer, 2002.
325. A. Marks and S. Unger, *Baire measurable paradoxical decompositions via matchings*, Advances in Mathematics **289** (2016), 397–410.
326. R. Marshall, *Robust classes of finite structures*, Ph.D. thesis, University of Leeds, 2008.
327. P. Martin-Löf, *The definition of random sequences*, Information and Control **9** (1966), 602–619.
328. K. R. Matthews, *Generalized* $3x + 1$ *mappings: Markov cxhains and ergodic theory*, The Ultimate Challenge: The $3x + 1$ Problem (J. C. Lagarias, ed.), American Mathematical Society, 2010, pp. 79–.
329. K. R. Matthews and A. M. Watts, *A generalization of Hasse's generalization of the Syracuse algorithm*, Acta Arithmetica **43** (1984), 167–175.
330. P. Mattila, *Geometry of sets and measures in Euclidean spaces*, Cambridge Studies in Advanced Mathematics, no. 44, Cambridge University Press, 1995.
331. A. D. Matushkin, *Zero-one law for random uniform hypergraphs*, preprint, 2016.
332. A. D. Matushkin and M. E. Zhukovskii, *First order sentences about random graphs: small number of alternations*, Discrete Applied Mathematics **236** (2018), 329–346.
333. J. E. Maxfield, *Normal k-tuples*, Pacific Journal of Mathematics **3** (1953), 189–196.
334. E. Mayordomo, *A Kolmogorov complexity characterization of constructive Hausdorff dimension*, Information Processing Letters **84** (2002), 1–3.
335. W. Merkle, N. Mihalović, and T. A. Slaman, *Some results on effective randomness*, Theory of Computing Systems **39** (2006), 707–721.
336. J.-F. Mertens, *Stochastic games*, Handbook of Game Theory, vol. 3, Elsevier, 2002, pp. 1809–1832.
337. J.-F. Mertens and A. Neyman, *Stochastic games*, International Journal of Game Theory **10** (1981), 53–66.
338. P. Michel, *Busy beaver competition and Collatz-like problems*, Archive for Mathematical Logic **32** (1993), 351–367.
339. P. Michel and M. Margenstern, *Generalized* $3x+1$ *functions and the theory of computation*, The Ultimate Challenge: The $3x+1$ Problem (J. C. Lagarias, ed.), American Mathematical Society, 2010, pp. 105–.
340. C. F. Miller, III, *On group-theoretic decision problems and their classification*, Annals of Mathematics Studies, no. 68, Princeton University Press, 1971.
341. G. A. Miller and D. McNeill, *Psycholinguistics*, The Handbook of Social Psychology (G. Lindzey and E. Aronson, eds.), vol. 3, Addison-Wesley, 2nd ed., 1968, pp. 666–794.
342. G. L. Miller, *Riemann's hypothesis and tests for primality*, Journal of computer and system science **13** (1976), 300–317.
343. R. Miller, *Isomorphism and classification for countable structures*, Computability **8** (2019), 99–117.
344. J. Milnor, *Dynamics in one complex variable*, 3rd ed., Annals of Mathematics Studies, no. 160, Princeton University Press, 2006.

345. A. Montalbán, *Computable structure theory: Within the arithmetic*, preprint, 2019.
346. *****A. Montalban, *Computable structure theory, part ii*, not yet seen, 2021?
347. A. Montalbán and A. Nies, *Borel structures: a brief survey*, Effective Mathematics of the Uncountable, Lecture Notes in Logic, no. 41, Cambridge, 2013, pp. 124–134.
348. C. C. Moore, *Ergodicity of flows on homogeneous spaces*, American Journal of Mathematics **88** (1966), 154–178.
349. A. De Morgan, *Formal logic: or, the calculus of inference, necessary and probable*, Taylor and Walton, 1847.
350. R. A. Moser and G. Tardos, *A constructive proof of the general Lovász local lemma*, Journal of the Association for Computing Machinery **57** (2010), 1–15.
351. A. A. Muchnik, A. L. Semenov, and V. A. Uspensky, *Mathematical metaphysics of randomness*, Theoretical Computer Science **207** (1998), 263–317.
352. K. P. Murphy, *Dynamic Bayesian networks*, Ph.D. thesis, University of California, Berkeley, 2002.
353. T. Neary, *Small polynomial time universal Turing machines*, Proceedings of the 4th Irish Conference on the Mathematical Foundations of Computer Science and Information Technology (T. Hurley, et al., ed.), 2006, pp. 325–329.
354. A. Nies, *Computability and randomness*, Oxford Logic Guides, no. 51, Oxford, 2009.
355. N. Nisan and A. Wigderson, *Hardness vs. randomness*, Journal of Computer Systems and Sciences **49** (1994), 149–167.
356. P. S. Novikov, *On algorithmic unsolvability of the word problem in group theory*, Trudy Matimaticheskogo Instituta imeni V.A. Steklova, no. 44, Izdat. Akad. Nauk SSSR, 1955.
357. D. S. Ornstein and B. Weiss, *Ergodic theory of amenable group actions. I: The Rohlin Lemma*, Bulletin of the American Mathematical Society **2** (1980), 161–164.
358. D. N. Osherson and S. Weinstein, *Criteria of language learning*, Information and Control **52** (1982), 123–138.
359. D. Osin, *A topological zero-one law and elementary equivalence of finitely generated groups*, Annals of Pure and Applied Logic **172** (2021), 102915.
360. R. Pal, I. Ivanov, A. Datta, M. L. Bittner, and E. R. Dougherty, *Generating Boolean networks with a prescribed attractor structure*, Bioinformatics **21** (2005), 4021–4025.
361. J. Paris and A. Venkovská, *Pure inductive logic*, Perspectives in Logic, Cambridge, 2015.
362. J. Pearl, *Deciding consistency in inheritance networks*, Tech. Report 870053, Cognitive Systems Laboratory, UCLA, 1987.
363. ———, *Probabilistic reasoning in intelligent systems*, Morgan Kaufmann, 1988.
364. J. Pearl and A. Paz, *Graphoids: A graph-based logic for reasoning about relevance relations*, Tech. Report 850038, Cognitive Systems Laboratory, UCLA, 1985.
365. Y. B. Pesin, *Dimension theory in dynamical systems*, University of Chicago Press, 1997.
366. P. Petersen, *Riemannian geometry*, 3rd ed., Graduate Texts in Mathematics, no. 171, Springer, 2016.
367. F. Petrov and A. Vershik, *Uncountable graphs and invariant measures on the set of universal countable graphs*, Random Structures and Algorithms **37** (2010), 389–406.
368. R. R. Phelps, *Lectures on choquet's theorem*, Lecture Notes in Mathematics, no. 1757, Springer, 2001.
369. A. Pillay and C. Steinhorn, *Discrete o-minimal structures*, Annals of Pure and Applied Logic **34** (1987), 275–289.
370. M. S. Pinsker, *On the complexity of a concentrator*, 7th International Teletraffic Conference, 1973.
371. K. Popper, *The logic of scientific discovery*, Routledge, 2002, Original edition published in 1935; this edition dates from text of 1959.
372. E. L. Post, *Recursive unsolvability of a problem of Thue*, The Journal of Symbolic Logic **12** (1947), 1–11.
373. P. Potgieter, *Algorithmically random series and Brownian motion*, Annals of Pure and Applied Logic **169** (2018), 1210–1226.
374. M. O. Rabin, *Probabilistic algorithm for testing primality*, Journal of Number Theory **12** (1980), 128–138.
375. C. Radin and L. Sadun, *Phase transitions in a complex network*, Journal of Physics A **46** (2013), 305002.

376. F. P. Ramsey, *Truth and probability*, The Foundations of Mathematics and other Logical Essays (R. B. Braithwaite, ed.), Harcourt Brace, 1931, First published 1926, pp. 156–198.
377. J. Reimann, *Computability and fractal dimension*, Ph.D. thesis, Ruprecht-Karls-Universität Heidelberg, 2004.
378. R. Reiter, *A logic for default reasoning*, Artificial Intelligence **13** (1980), 81–132.
379. ———, *Nonmonotonic reasoning*, Annual Review of Computer Science **1987** (1987), 147–186.
380. L. J. Richter, *Degrees of structures*, The Journal of Symbolic Logic **46** (1981), 723–731.
381. A. Rivkind and O. Barak, *Local dynamics in trained recurrent neural networks*, Physical Review Letters **118** (2017), no. 258101, 1–5.
382. A. Robinson, *Complete theories*, Studies in logic and the foundations of mathematics, North-Holland, 1956.
383. F. Rosenblatt, *Principles of neurodynamics; perceptrons and the theory of brain mechansims*, Spartan, 1962.
384. S. Ross, *Introduction to probability models*, 12th ed., Academic Press, 2019.
385. J. J. Rotman, *The theory of groups*, Allyn and Bacon, 1965.
386. A. Rumyantsev, *Infinite computable version of Lovász local lemma*, preprint, 2010.
387. A. Rumyantsev and A. Shen, *Probabilistic constructions of computable objects and a computable version of Lovász local lemma*, Fundamental Informaticae **132** (2014), 1–14.
388. S. J. Russell and P. Norvig, *Artificial intelligence: A modern approach*, third ed., Prentice Hall Series in Artificial Intelligence, Prentice Hall, 2010.
389. J. Rute, *Algorithmic randomness and constructive/computable measure theory*, Algorithmic Randomness: Progress and Prospects (J. N. Y. Franklin and C. P. Porter, eds.), Lecture Notes in Logic, no. 50, Cambridge, 2020, pp. 58–114.
390. M. J. Ryten, *Model theory of finite difference fields and simple groups*, Ph.D. thesis, The University of Leeds, 2007.
391. N. Sauer, *On the density of families of sets*, Journal of Combinatorial Theory (A) **13** (1972), 145–147.
392. A. Saxe, Y. Bansal, J. Dapello, M. Advani, A. Kolchinsky, B. Tracey, and D. Cox, *On the information bottleneck theory of deep learning*, ICLR, 2018.
393. A.-M. Scheerer, *Computable absolutely normal numbers and discrepancies*, Mathematics of Computation (2017), 2911–2926.
394. J. Schmidhuber, *Deep learning in neural networks: An overview*, Neural Networks **61** (2015), 85–117.
395. K. Schmidt, *Asymptotically invariant sequences and an action of $SL(2, Z)$ on the 2-sphere*, Israel Journal of Mathematics **37** (1980), 193–208.
396. W. M. Schmidt, *Über die Normalität von Zahlen zu verschiedenen Basen*, Acta Arithmetica **VII** (1962), 299–309.
397. ———, *Irregularities of distribution, VII*, Acta Arithmetica **21** (1972), 45–50.
398. C. P. Schnorr, *A unified approach to the definition of random sequences*, Mathematical systems theory **5** (1971), 246–258.
399. ———, *Zufälligkeit und Wahrscheinlichkeit*, Lecture Notes in Mathematics, no. 218, Springer, 1971.
400. G. Shafer, *A mathematical theory of evidence*, Princeton University Press, 1976.
401. A. Shamir, **IP = PSPACE**, Journal of the Association for Computing Machinery **39** (1992), 869–877.
402. C. E. Shannon, *A mathematical theory of communication*, The Bell System Technical Journal **27** (1948), 379–423.
403. L. S. Shapley, *Stochastic games*, Proceedings of the National Academy of Sciences **39** (1953), 1095–1100.
404. S. Shelah, *A combinatorial problem; stability and order for models and theories in infinitary languages*, Pacific Journal of Mathematics **41** (1972), 247–261.
405. ———, *Classification theory and the number of non-isomorphic models*, revised ed., Studies in Logic and the Foundations of Mathematics, no. 92, North-Holland, 1990.
406. S. Shelah and J. Spencer, *Zero-one laws for sparse random graphs*, Journal of the American Mathematical Society **1** (1988), 97–115.
407. A. Shen, *IP = PSPACE: Simplified proof*, Journal of the Association for Computing Machinery **39** (1992), 878–880.

408. A. K. Shen, *On relations between different algorithmic definitions of randomness*, Sovient Mathematics Doklady **38** (1989), 316–319.
409. A. N. Shiryaev, *Probability*, second ed., Graduate Texts in Mathematics, no. 95, Springer, 1996.
410. I. Shmulevich and E. R. Dougherty, *Genomic signal processing*, Princeton Series in Applied Mathematics, Princeton University Press, 2007.
411. _____, *Probabilisitc boolean networks*, Society for Industrial and Applied Mathematics, 2010.
412. H. A. Simon, *On a class of skew distribution functions*, Biometrika **42** (1955), 425–440.
413. P. Simon, *A guide to NIP theories*, Lecture Notes in Logic, no. 44, Cambridge, 2015.
414. R. I. Soare, *Recursively enumerable sets and degrees*, Springer-Verlag, 1987.
415. J. Spencer, *Asymptotic lower bounds for Ramsey functions*, Discrete Mathematics **20** (1977), 69–76.
416. _____, *Threshold functions for extension statements*, Journal of Combinatorial Theory A **53** (1990), 286–305.
417. _____, *Threshold spectra via the Ehrenfeucht game*, Discrete Applied Mathematics **30** (1991), 235–252.
418. _____, *The strange logic of random graphs*, Algorithms and Combinatorics, no. 22, Springer, 2001.
419. J. Spencer and K. St. John, *The tenacity of zero-one laws*, The Electronic Journal of Combinatorics **8** (2001), R17.1–R17.14.
420. J. Spencer and M. E. Zhukovskii, *Bounded quantifier depth spectra for random graphs*, Discrete Mathematics **339** (2016), 1651–1664.
421. L. Staiger, *Kolmogorov complexity and Hausdorff dimension*, Information and Computation **103** (1993), 159–194.
422. _____, *A tight upper bound on Kolmogorov complexity and uniformly optimal prediction*, Theory of Computing Systems **31** (1998), 215–229.
423. C. I. Steinhorn, *Borel structures and measure and category logics*, Model Theoretic Logics, Perspectives in Logic, no. 8, Springer, 1985, pp. 579–596.
424. G. Stengle and J. E. Yukich, *Some new Vapnik-Chervonenkis classes*, The Annals of Statistics **17** (1989), 1441–1446.
425. V. E. Stepanov, *Phase transitions in random graphs*, Theory of Probability and its Applications **15** (1970), 187–203.
426. F. Stephan and Yu. Ventsov, *Learning algebraic structures from text*, Theoretical Computer Science **268** (2001), 221–273.
427. L. J. Stockmeyer, *The polynomial-time hierarchy*, Theoretical Computer Science **3** (1976), 1–22.
428. G. Stuck and R. Zimmer, *Stabilizers for ergodic actions of higher rank semisimple groups*, Annals of Mathematics **139** (1994), 723–747.
429. M. Studený, *Conditional independence relations have no finite complete characterization*, preprint, 1992.
430. D. Sussillo and O. Barak, *Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks*, Neural Computation **25** (2013), 626–649.
431. E. Szemerédi, *On sets of integers containing no k elements in arithmetic progression*, Acta Arithmetica **27** (1975), 199–245.
432. _____, *Regular partitions of graphs*, preprint, 9 pp., 1975.
433. T. Tao, *Expanding polynomials over finite fields of large characteristic, and a regularity lemma for definable sets*, Contributions to Discrete Mathematics **10** (2015), 22–98.
434. _____, *Szemerédi's proof of Szemerédi's theorem*, Acta Mathematica Hungarica **161** (2020), 443–487.
435. A. Tarski, *Algebraische Fassung des Maßproblems*, Fundamenta Mathematicae **31** (1938), 47–66.
436. _____, *Cardinal algebras*, Oxford University Press, 1949.
437. A. Taveneaux, *Randomness zoo*, preprint, 2012.
438. E. Thoma, *Die unzerlegbaren, positiv-definiten Klassenfunktionen der abzählbar unendlichen, symmetrischen Gruppe*, Mathematische Zeitschfrift **85** (1964), 40–61.
439. _____, *Über unitäre Darstellungen abzählbarer, diskreter Gruppen*, Mathematische Annalen **153** (1964), 111–138.

440. S. Thomas, *The classification problem for torsion-free Abelian groups of finite rank*, Journal of the American Mathematical Society **16** (2003), 233–258.
441. S. Thomas and R. Tucker-Drob, *Invariant random subgroups of strictly diagonal limits of finite symmetric groups*, Bulletin of the London Mathematical Society **46** (2014), 1007–1020.
442. ———, *Invariant random subgroups of inductive limits of finite alternating groups*, Journal of Algebra **503** (2018), 474–533.
443. A. Thue, *Probleme über Veränderungen von Zeichenreihen nach gegeben regeln*, Skrifter utgit av Videnskapsselskapet i Kristiania **1** (1914), no. 10, 1–34.
444. N. Tishby and N. Zaslavsky, *Deep learning and the information bottleneck principle*, IEEE Information Theory Workshop, 2015.
445. S. Toda, *PP is as hard as the polynomial-time hierarchy*, SIAM Journal of Computing **20** (1991), 878–880.
446. H. Towsner, *σ-algebras for quasirandom hypergraphs*, Random Structures and Algorithms **50** (2017), 114–139.
447. ———, *Algorithmic randomness in ergodic theory*, Algorithmic Randomness: Progress and Prospects (J. N. Y. Franklin and C. P. Porter, eds.), Lecture Notes in Logic, no. 50, Cambridge, 2020, pp. 40–57.
448. S. Vadhan, *Pseudorandomness*, Foundations and Trends in Theoretical Computer Science, vol. 7, Now, 2012.
449. L. G. Valiant, *The complexity of computing the permanent*, Theoretical Computer Science **8** (1979), 189–201.
450. ———, *A theory of the learnable*, Communications of the ACM **27** (1984), 1134–1142.
451. P. van der Hoorn, G. Lippner, and D. Krioukov, *Sparse maximum-entropy random graphs with a given power-law degree distribution*, Journal of Statistical Physics **173** (2018), 803–844.
452. V. N. Vapnik and A. Ya. Chervonenkis, *On the uniform convergence of relative frequencies of events to their probabilities*, Theory of probability and its applications **16** (1971), 264–280.
453. A. M. Vershik, *Totally nonfree actions and the infinite symmetric group*, Moscow Mathematical Journal **12** (2012), 193–212.
454. N. Vieille, *Stochastic games: Recent results*, Handbook of Game Theory, vol. 3, Elsevier, 2002, pp. 1833–1850.
455. Susan Vineberg, *Dutch Book Arguments*, The Stanford Encyclopedia of Philosophy (Edward N. Zalta, ed.), Metaphysics Research Lab, Stanford University, spring 2016 ed., 2016.
456. A. Visser, *Numerations, λ-calculus, & arithmetic*, To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism, Academic Press, 1980, pp. 259–284.
457. J. von Neumann, *Zur allgemeinen Theorie des Masses*, Fundamenta Mathematicae **13** (1929), 73–116.
458. P. Walters, *An introduction to ergodic theory*, Graduate Texts in Mathematics, no. 79, Springer, 1982.
459. R. Weber, *Computability theory*, Student Mathematical Library, no. 62, American Mathematical Society, 2012.
460. K. Weihrauch, *Computable analysis*, Texts in Theoretical Computer Science, Springer, 2000.
461. J. Williamson, *Probability logic*, Handbook of the Logic of Argument and Inference, Studies in Logic and Practical Reasoning, vol. 1, Elsevier, 2002, pp. 397–424.
462. J. S. Wilson, *On simple pseudofinite groups*, Journal of the London Mathematical Society **51** (1995), 471–490.
463. D. Xiao, *New perspectives on the complexity of computational learning, and other problems in theoretical computer science*, Ph.D. thesis, Princeton University, 2009.
464. ———, *On basing $\mathbf{ZK} \neq \mathbf{BPP}$ on the hardness of PAC learning*, 24th Annual IEEE Conference on Computational Complexity, 2009, pp. 304–315.
465. A. C. Yao, *Theory and applications of trapdoor functions*, 23rd Annual Symposium on Foundations of Computer Science, IEEE, 1982, pp. 80–91.
466. C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, *Understanding deep learning requires rethinking generalization*, ICLR, 2017.
467. X. Zheng and R. Rettinger, *On the extnesions of Solovay reducibility*, Computing and Combinatorics: 10th Annual International Conference, COCOON 2004, Lecture Notes in Computer Science, no. 3106, Springer, 2004, pp. 360–369.

468. M. E. Zhukovskii, *On infinite spectra of first order properties of random graphs*, Moscow Journal of Combinatorics and Number Theory **4** (2016), 73–102.
469. R. J. Zimmer, *Ergodic theory and semisimple groups*, Monographs in Mathematics, no. 81, Birkhäuser, 1984.