Space-Bounded Church-Turing Thesis and Computational Tractability of Closed Systems

Mark Braverman, Jonathan Schneider, and Cristóbal Rojas

Phys. Rev. Lett. **115**, 098701 — Published 27 August 2015

DOI: 10.1103/PhysRevLett.115.098701

Space-bounded Church-Turing thesis and computational tractability of closed systems

Mark Braverman and Jonathan Schneider
Computer Science Department, Princeton University

Cristóbal Rojas
Departamento de Matemáticas, Universidad Andres Bello
(Dated: July 17, 2015)

We report a new limitation on the ability of physical systems to perform computation – one that is based on generalizing the notion of memory, or storage space, available to the system to perform the computation. Roughly, we define memory as the maximal amount of information that the evolving system can carry from one instant to the next. We show that memory is a limiting factor in computation even in lieu of *any* time limitations on the evolving system — such as when considering its equilibrium regime. We call this limitation the Space-Bounded Church Turing Thesis (**SBCT**). The **SBCT** is supported by a Simulation Assertion (**SA**), which states that predicting the long-term behavior of bounded-memory systems is computationally tractable. In particular, one corollary of **SA** is an explicit bound on the computational hardness of the long-term behavior of a discrete-time finite-dimensional dynamical system that is affected by noise. We prove such a bound explicitly.

Can we use computers to predict the future of evolving physical systems? What are the computational capabilities of physical systems? The fundamental Church-Turing thesis (CTT) [1], and its physical counterparts [2, 3] assert that any computation that can be carried out in finite time by a physical device can be carried out by a Turing Machine. The thesis is sometimes paraphrased in the following way: provided all the initial conditions with arbitrarily good precision, and random bits when necessary, the Turing Machine can *simulate* the physical system S over any fixed period of time $[0, T]$ for $T < \infty$.

However, there exists conceivable situations which, while respecting all physical principles, would allow for nature to exhibit behavior that cannot be simulated by computers [4, 5]. Note that the power of a physical process which is being used as a computer, critically depends on our ability to prepare the system and take measurements of it. Therefore, the impossibility to simulate some natural processes does not immediately contradict CTT. In particular, it is not clear that the finite state spaces accessed by (quantum or classical) computers are sufficient to simulate, with arbitrary accuracy, all the processes one finds in nature, which may take place in infinite-dimensional spaces [4].

Moreover, even if we can simulate a system for any fixed period of time T, in many situations one would like to know more and predict the asymptotic properties of the system as $T \to \infty$, i.e. as it reaches its equilibrium regime. In this case, the computational unsolvability of problems like the Halting Problem — itself a long-term property of Turing Machines — implies that rich enough physical systems may exhibit non-computable asymptotic behavior [6–17].

As Feynman describes it [18], to simulate the statistical asymptotic behavior of a physical system (say its equilibrium regime) means to have a machine which, when provided with a sequence of uniform random bits as input, outputs a sequence of states of the system *with exactly the same* probability as nature does. Note that even if there is a finite number of states which are *distinguishable* for the physical measurement, the associated probability distribution may well be *continuous*. The non-computable examples mean that this infinite time horizon simulation is sometimes just not possible. For instance, there exist computable dynamical systems (e.g. maps on the unit interval [17] or cellular automata [19]) for which there is a positive measure set of initial conditions leading to the same equilibrium regime — so it is a "physical state" — that yet no Turing machine can simulate in this way.

On the other hand, it has also been observed that this analysis may be affected by restricting some of the features related to the physical plausibility of the systems considered, such as dimensionality, compactness, smoothness or robustness to noise — the long term behavior of such restricted systems may be easier to predict [4, 11, 15, 16, 20].

In this Letter, we report a new bound on the ability of physical systems to perform computation — one that is based on generalizing the notion of storage space from computational complexity theory to continuous physical systems. More precisely, we provide a formal definition of memory for physical systems and postulate an explicit quantitative bound on the computational complexity of their simulations. According to our postulate, bounded memory physical systems should not exhibit non-computable phenomena even in the infinite-time horizon. As evidence for our postulate, we rigorously prove that for compact noisy systems, the non-computable phenomenon is broken by the noise even in the infinite-dimensional case. Moreover, to substantiate the quantitative part of the thesis, we show that if the noise is not a source of additional complexity, then the additional space requirements for simulating the system below the noise threshold are minimal.

Consider a closed, stochastic system $\mathcal{S} = X_t$ over a state space $\mathcal{X}$. If the time t is discrete, define the *memory available to* $\mathcal{S}$ as

$$\mathcal{M}(\mathcal{S}) := \sup_t \sup_{\mu \text{ distribution on } \mathcal{X}} I_{X_t \sim \mu}(X_t; X_{t+1}). \quad (1)$$

Here $I(X_t; X_{t+1})$ is Shannon's mutual information [21]. If $f(x,y)$ is the PDF of the distribution of (X_t, X_{t+1}) where $X_t \sim \mu$ and $X_{t+1} \sim X_{t+1}|_{X_t}$, then

$$I_{X_t \sim \mu}(X_t; X_{t+1}) := \iint f(x,y) \log \frac{f(x,y)}{f(x)f(y)}\, dx\, dy. \quad (2)$$

We take the supremum over all possible distributions μ. Therefore $\mathcal{M}(\mathcal{S})$ measures the maximum amount of information the system can carry from one time step to the next. Note that if the space $\mathcal{X}$ is finite of size N then $\mathcal{M}(\mathcal{S})$ is bounded by the entropy $H(X_t) \leq \log_2 |\mathcal{X}| = \log_2 N$. As discussed below, in the presence of noise, all bounded finite-dimensional systems have finite memory available.

For continuous-time systems we define memory available at time lapse Δt as the amount of information that may be preserved for Δt time units:

$$\mathcal{M}_{\Delta t}(\mathcal{S}) := \sup_t \sup_{\mu \text{ distribution on } \mathcal{X}} I_{X_t \sim \mu}(X_t; X_{t+\Delta t}). \quad (3)$$

Information theoretic considerations imply that $\mathcal{M}_{\Delta t}(\mathcal{S})$ is a non-increasing function of Δt. The time-lapse Δt is chosen to be the highest among the values of Δt for which the behavior of the system at time scales below Δt is dynamically and computationally simple. It is possible to artificially construct an example where $\lim_{\Delta t \to 0} \mathcal{M}_{\Delta t}(\mathcal{S}) = \infty$, and where by encoding computation on a shrinking set of time intervals the computational power of the system is unbounded [22]. However, it has been pointed out [23, 24] that quantum mechanical considerations impose an ultimate lower bound $\Delta t \geq t^*$ [25] on the time it takes for a physical device to perform one logical operation.

We postulate that the memory $\mathcal{M}(\mathcal{S})$ is an intrinsic limitation on the ability of physical systems to perform computation. We call the limitation the Space-Bounded Church Turing thesis (**SBCT**):

[SBCT]: *If a physical system $\mathcal{S}$ has memory $s = \mathcal{M}(\mathcal{S})$ available to it, then it is only capable of performing computation in the complexity class* $\mathbf{SPACE}(s^{O(1)})$*, even when provided with unlimited time.*

SBCT is supported by the following assertion: **[Simulation Assertion, SA]:** *The problem of simulating the asymptotic behavior of a physical system $\mathcal{S}$ as in* **SBCT** *with n precision bits is in the complexity class* $\mathbf{SPACE}\left((s + \log n)^{O(1)}\right)$.

SA implies, in particular, that the long-term behavior of bounded-memory systems is computable. This covers a broad class of noisy systems. Interestingly, a number of low-dimensional systems with non-computable long-term behavior is known [11–17]. These examples require considerable care in their construction. As explained below, assuming the **SBCT** one should expect these constructions to be delicate, to the point of making them physically implausible.

It is clear that **SA** implies **SBCT**. While, logically speaking, the converse also (almost) holds, it is still useful to make a distinction between the two statements. A low-memory system $\mathcal{S}$ may be hard to simulate, for example, because of the hardness of the noise operator. Such a system would violate **SA**. However, it might still essentially satisfy **SBCT** — being incapable of performing computation outside the class $\mathbf{SPACE}(s^{O(1)})$ — save for the problem of simulating $\mathcal{S}$ itself.

SBCT can be considered in the context of other quantitative variants of the Church-Turing Thesis, notably the Extended Church-Turing Thesis (ECT) which asserts that physically-feasible computations are not only computable, but are *efficiently computable* in the sense of computational complexity theory [26]. Whereas previous discussions of efficiency focused on *time complexity* [27–31], we shift the discussion to *storage space complexity* (known as space complexity in the Computer Science literature). This shift has the benefit of allowing one to make assertions bounding the computational power of systems even when provided with unlimited time — we e.g. can allow the system to reach equilibrium at $t \to \infty$, and consider the outcome to be the output of the computation. We assert that this outcome will still not enhance the computational power of the system beyond its memory constraints.

In the theory of computational complexity, $\mathbf{SPACE}(S(n))$ is the complexity class of problems that can be solved by a Turing Machine which uses at most $S(n)$ bits of memory to solve instances of size n [32, 33]. Of particular interest are the classes of problems **PSPACE** and **LOGSPACE** where $S(n) = n^{O(1)}$ and $S(n) = O(\log n)$, respectively [34]. Putting these classes in the context of **P** and **NP**, the following chain of inclusions is known:

$$\mathbf{LOGSPACE} \subset \mathbf{P} \subset \mathbf{NP} \subset \mathbf{PSPACE}.$$

All of these inclusions are believed to be strict, although only the fact that $\mathbf{LOGSPACE} \subsetneq \mathbf{PSPACE}$ is known.

Space-bounded complexity classes exhibit several important robustness properties that do not have a parallel when considering time-bounded computation. For example, the space-bounded analogue of $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ has been resolved in the affirmative: $\mathbf{PSPACE} = \mathbf{NPSPACE}$ [35] — thus **PSPACE** is closed under the use of non-determinism. The question of whether quantum computation speeds up computation time in some cases, i.e.

whether $\mathbf{P} \subsetneq \mathbf{BQP}$, remains open, but likely the answer is that it does [36, 37]. In the case of *space* limitations, it is known that **BQPSPACE** = **PSPACE**, and thus quantum computing is not particularly useful [38] (suggesting that, unlike the ECT, the **SBCT** has a good chance of holding in a quantum world).

A bound of $S(n)$ on the amount of memory used by a computation means that the machine may be in at most $2^{S(n)}$ distinct states. If the computation is deterministic, this imposes a natural hard limit of $2^{S(n)}$ on its computation time: the computation either terminates in $2^{S(n)}$ steps, or ends up in an infinite loop. If the computation is randomized, then it naturally translates into a Markov chain on its $2^{S(n)}$ states. The stationary distribution(s) of the chain, which can be computed in $\mathrm{poly}(S(n))$ space, characterize the infinite-time horizon behavior of the machine. We assert that more generally, the ability of physical systems to remember information is the limiting factor for their computational power.

While in many cases the complexity of the system falls below the bound provided by **SBCT**, the power of **SBCT** partially arises from the fact that it is generally much easier to estimate the memory available to a system than its computational power/hardness.

The non-computability constructions mentioned earlier mean that while analytic methods can prove *some* long-term properties of *some* dynamical systems, for "rich enough" systems, one cannot hope to have a general closed-form *analytic* algorithm, i.e. one that is not based on simulations, that computes the properties of its long-term behavior. This fundamental phenomenon is qualitatively different from chaotic behavior, and has even led some researchers to claim [39] that the enterprise of theoretical physics itself is doomed from the outset; rather than attempting to construct solvable mathematical models of physical processes, computational models should be built, explored, and empirically analyzed.

However, it is a notable fact that in all the specific low-dimensional examples the non-computability phenomenon is *not robust to noise*: all these constructions are based on a fine structure responsible for Turing simulation which is destroyed once one introduces even a small amount of noise into the system. This has been explicitly observed e.g. for neural networks [40] and reachability problems [41]. This is consistent with the **SBCT**: a low-dimensional compact system affected by noise becomes a bounded-memory system, and is therefore explicitly limited in its computational power, and cannot serve as a universal computer.

In [42], an interesting example of a constant-dimensional analytic system capable of robustly performing universal computation is constructed. However, this system acts on an unbounded domain, and has therefore infinitely many robustly distinguishable states; i.e. infinite memory. This is again consistent with the **SBCT**.

We now turn to the rigorous analysis of discrete-time dynamical systems over continuous spaces, affected by random noise. In such models, the evolution is governed by a deterministic map T acting on phase space $\mathcal{X}$, together with a small random noise p^ε. The noisy system $\mathcal{S}_\varepsilon$ jumps, in one unit of time, from state x to $T(x)$ and then disperses randomly around $T(x)$ with distribution $p^\varepsilon_{T(x)}$. The parameter ε controls the "magnitude" of the noise, so that $p^\varepsilon_{T(x)}(\cdot) \to T(x)$ as $\varepsilon \to 0$ [43]. For example, $p^\varepsilon_{T(x)}(\cdot)$ could be taken to be uniform on an ε-ball around $T(x)$ or a Gaussian with mean $T(x)$ and variance ε. In all what follows we will assume, for the sake of simplicity, that the underlying system is one-dimensional and size($\mathcal{X}$)=1. That is, $\mathcal{X}$ can be thought of as the interval $[0, 1]$.

By expressing mutual information in terms of entropy and conditional entropy, it is not hard to estimate the memory of the system $\mathcal{S}_\varepsilon$ for each of these types of noise (uniform on an ε-ball or Gaussian). Indeed, if f_X stands for the PDF of a random variable X, then the entropy of X is defined by

$$H(X) = -\int f_X(x) \log(f_X(x))\, dx,$$

and mutual information can be expressed as

$$I(X_t; X_{t+1}) = H(X_{t+1}) - H(X_{t+1}|X_t).$$

On the one hand, since $H(p^\varepsilon) = \Theta(\log(\varepsilon))$ for both uniform on an ε-ball and Gaussian distributions and since $H(X_{t+1}) \leq 0$ and $X_{t+1}|X_t \sim p^\varepsilon$, we obtain that $I(X_{t+1}; X_t) \leq O(\log 1/\varepsilon)$. On the other hand, $H(X_{t+1})$ is maximized by the uniform distribution on $\mathcal{X}$, having a value of $\log(\mathrm{size}(\mathcal{X})) = 0$. It follows that $I(X_t; X_{t+1})$ is maximized by this distribution as well, and therefore $\mathcal{M}(\mathcal{S}_\varepsilon) = \Theta(\log(1/\varepsilon))$. The **SBCT** then predicts that the computational power of the system $\mathcal{S}_\varepsilon$ is in the complexity class $\mathbf{SPACE}\left(\log^{O(1)}(1/\varepsilon)\right)$.

How can the actual computational power of these systems be estimated? In order to give an upper bound one would have to give a generic algorithm for the noisy system that computes its long-term features. This would establish the **SA** for the system, and thus imply the **SBCT**. In order to give a lower bound one would have to show that even in the presence of noise the system is capable of simulating a Turing Machine subject to memory restrictions. We now explain how to prove such bounds.

Since the evolution of these systems is stochastic, only the statistical properties can be studied — instead of asking whether the system will ever fall in a given region B, we shall ask what is the probability of the system being in such a region, as $t \to \infty$.

These properties are mathematically described by the *invariant measures* of the system — the possible statistical behaviors once the system has converged to a "steady state" distribution. Quantities such as Lyapunov exponents or escape rates can be computed from the relevant invariant measure. Standard references on this material are [44–46].

Here, by computing a probability distribution μ over $[0,1]$ we mean to have a finite algorithm A that can produce arbitrarily good rational approximations to the probability of any interval with rational endpoints. That is, the algorithm A, upon input $(a,b,\delta)\in\mathbb{Q}^3$, must output a rational number $A(a,b,\delta)$ satisfying $|A(a,b,\delta)-\mu[a,b]|\leq\delta$. See for instance [47]. This definition is equivalent to the existence of a probabilistic machine producing a sequence of states distributed exactly according to μ [48].

Our first result, which can be seen as supporting the qualitative part of **SBCT**, shows that the addition of any amount of noise to a system is sufficient to destroy any non-computable behavior, even in the infinite-dimensional case.

Statement A: *If a compact system is affected by small random ε-noise as described above, then all its ergodic invariant measures are computable.*

Intuitively, this theorem says noise turns asymptotic statistical properties from non-computable to computable. Its proof essentially follows from the fact that the presence of noise forces the system to have only "well separated" ergodic measures. An exhaustive search can then be performed, and compactness guarantees that all such measures will be eventually found (there can only be finitely many of them). We note that the result holds even if the state space is infinite dimensional. We refer to [49] for a complete proof.

Thus, we know that in presence of ε-noise, ergodic measures are all computable. In addition, according to the **SBCT**, their computational power should be bounded in terms of their dimension and size. In order to give an upper bound, we prove a version of the **SA** by exhibiting an algorithm that computes the invariant measure to arbitrary accuracy using very little space. Specifically, we show:

Statement B: *Let $\mathcal{S}$ be a compact, constant-dimensional system affected by ε-Gaussian noise. Suppose that the transition function f is uniformly analytic and can be computed to within precision 2^{-m} using $O(\log m)$ space. Then the invariant measure of the noisy system S_ε can be computed with a given precision 2^{-n} in* $\mathbf{SPACE}(\mathrm{poly}(\log 1/\varepsilon)+\mathrm{poly}(\log n))$.

This statement implies, in particular, that the long-term behavior of noisy systems at scales below the noise level is computable in time quasipolynomial in n. Intuitively, this means that, at the right scale, the behavior of the system is governed by the efficiently predictable micro-analytic structure of the noise, rather than by the macro-dynamic structure of the system that can be computationally difficult to predict.

The formal proof of the above statement can be found in the accompanying paper [50]. Moreover, up to the polynomial factors, the statement can be shown to be tight: we can robustly separate $1/\varepsilon$ states of $\mathcal{S}_\varepsilon$, and thus simulate a computation that uses $\sim\log 1/\varepsilon$ bits of memory. Therefore, simulation using less than $\log 1/\varepsilon$ bits of memory is impossible due to the Space Hierarchy Theorems [33]. Note that the output of a precision-2^{-n} calculation requires $\geq n$ bits to write down. In the context of space-bounded computation, the output is stored in a write-only memory that is not part of the computation space. Still, in order to be able to write to a size-n outside memory, one needs to at least store indexes using $\log n$ bits, and thus the dependence on n is also optimal up to polynomial factors.

The algorithm establishing Statement B and its analysis consists of two main parts. The first idea is to exploit the mixing properties of the transition operator $\mathcal{P}$ of the perturbed system $\mathcal{S}_\varepsilon$. The transition operator contains Gaussian noise, and it thus has a spectral gap of at least $\exp(-1/\varepsilon^2)$, and will mix in time on the order of $T\approx\exp(-1/\varepsilon^2)$. We represent density functions of measures using piece-wise analytic functions with each piece of size $\approx\varepsilon$. On each piece we approximate the corresponding analytic function using $\approx n$ terms of its Taylor expansion, so that the density function is represented by a point in $\mathbb{R}^D$ where $D\approx n/\varepsilon$. When we consider the action of the transition operator $\mathcal{P}$ on these coefficients, we obtain a linear map $\mathcal{M}_\mathcal{P}$ whose coefficients can be computed in space $\mathrm{poly}(\log 1/\varepsilon)+\mathrm{poly}(\log n)$. By the mixing property, to approximate the invariant measure of $\mathcal{P}$ it suffices to raise $\mathcal{M}_\mathcal{P}$ to the T-th power.

The second part of the argument deals with raising a $D\times D$ matrix $\mathcal{M}_\mathcal{P}$ to power $T\approx 2^D$ using only $\mathrm{poly}(\log D)$ space. To the best of our knowledge, this problem has been previously addressed when T is polynomial but not exponential in D. The proof in [50] uses a number of techniques in space-efficient computation to obtain a degree-$O(D)$ polynomial $p(\cdot)$ such that the entries of $p(\mathcal{M}_\mathcal{P})-\mathcal{M}_\mathcal{P}^T$ have magnitude $\leq 2^{-n}$.

In conclusion, we postulated a principle that allows us to quantitatively bound the computational power of any device built out of a closed physical system — even when the device is allowed to run for an unlimited amount of time — in terms of the memory of the system. We have shown that this bound is tight for systems modeled by randomly perturbed dynamical processes, which account for a large part of physics. Additionally, we have shown that the asymptotic behavior of these systems can be computed at arbitrary precision, and that when computing below the noise level, the simulation can be achieved using an extremely limited amount of memory. Concerning quantum systems, the fact that general models like Topological Field Theories can be efficiently simulated by quantum computers [51] which, in turn, can be simulated by classical ones with only a quadratic increase in memory [38], suggests that our results apply in the quantum world as well.

Acknowledgments

MB was partially supported by an Alfred P. Sloan Fellowship, an NSF CAREER award (CCF-1149888), NSF CCF-0832797, a Turing Centenary Fellowship, a Packard Fellowship in Science and Engineering, and the Simons Collaboration on Algorithms and Geometry. CR was partially supported by projects Fondecyt 1150222, DI-782-15/R Universidad Andrés Bello and Basal PFB-03 CMM-Universidad de Chile.

We would like to thank José Aliste, Eric Allender, Cameron Freer, Andy Gomberoff, Aram Harrow, Avinatan Hassidim, Giorgio Krstulovic, Cristopher Moore, and the anonymous referees for their advice and comments on earlier versions of the manuscript.

[1] A. M. Turing, Proceedings of the London Mathematical Society **2**, 161 (1939).
[2] R. Gandy, in *The Kleene Symposium*, Studies in Logic and the Foundations of Mathematics, Vol. 101, edited by H. J. K. Jon Barwise and K. Kunen (Elsevier, 1980) pp. 123 – 148.
[3] B. Copeland and O. Shagrir, Minds and Machines **17**, 217 (2007).
[4] M. A. Nielsen, Physical Review Letters **79**, 2915 (1997).
[5] T. D. Kieu, International Journal of Theoretical Physics **42**, 1461 (2003).
[6] J. V. Neumann, *Theory of Self-Reproducing Automata*, edited by A. W. Burks (University of Illinois Press, Champaign, IL, USA, 1966).
[7] S. Wolfram, Communications in Mathematical Physics **96**, 15 (1984).
[8] M. Minsky, *Computation: finite and infinite machines*, edited by N. J. Englegood Cliffs (Prentice-Hall, Inc., 1967).
[9] H. Siegelmann and E. Sontag, Applied Mathematics Letters **4**, 77 (1991).
[10] E. Fredkin and T. Toffoli, in *Collision Based Computing*, edited by A. Adamatzky (Springer London, 2002) pp. 47–81.
[11] C. Moore, Physical Review Letters **64**, 2354 (1990).
[12] P. Koiran, M. Cosnard, and M. Garzon, Theoretical Computer Science **132**, 113 (1994).
[13] J. Reif, J. Tygar, and A. Yoshida, Discrete and Computational Geometry **11**, 265 (1994).
[14] E. Asarin, O. Maler, and A. Pnueli, Theoretical Computer Science **138**, 35 (1995).
[15] M. Braverman and M. Yampolsky, Journal of the AMS **19**, 551 (2006).
[16] M. Braverman and M. Yampolsky, in *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, STOC '07 (ACM, New York, NY, USA, 2007) pp. 709–716.
[17] S. Galatolo, M. Hoyrup, and C. Rojas, Discrete and Continuous Dynamical Systems **29**, 193 (2011).
[18] R. P. Feynman, International Journal of Theoretical Physics **21**, 467 (1982).
[19] B. Hellouin de Menibus and M. Sablik, Ergodic Theory and Dynamical Systems (To appear), arXiv:1301.1998.
[20] N. Israeli and N. Goldenfeld, Physical Review Letters **92**, 074105 (2004).
[21] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley series in telecommunications (J. Wiley and Sons, New York, 1991).
[22] B. J. Copeland, Minds and machines **12**, 461 (2002).
[23] S. Lloyd, Nature **406**, 1047 (2000).
[24] S. Lloyd, Physical Review Letters **88**, 237901 (2002).
[25] $t^* = h/4E$, where h is Plank's constant and E is the average energy of the system.
[26] I. Parberry, SIGACT News **18**, 54 (1986).
[27] A. Vergis, K. Steiglitz, and B. Dickinson, Mathematics and computers in simulation **28**, 91 (1986).
[28] H. T. Siegelmann and S. Fishman, Physica D: Nonlinear Phenomena **120**, 214 (1998).
[29] H. Siegelmann, A. Ben-Hur, and S. Fishman, Physical Review Letters **83**, 1463 (1999).
[30] D. S. Abrams and S. Lloyd, Physical Review Letters **81**, 3992 (1998).
[31] S. Aaronson, Physical Review A **71**, 032325 (2005).
[32] M. Sipser, *Introduction To The Theory Of Computation*, Advanced Topics Series (Thomson Course Technology, 2006).
[33] S. Arora and B. Barak, *Computational complexity: a modern approach* (Cambridge University Press, 2009).
[34] We recall that the notations $f = O(g)$ and $f = \Theta(g)$ mean that, up to a multiplicative constant, f is bounded by g and f is the same as g, respectively.
[35] W. J. Savitch, Journal of Computer and System Sciences **4**, 177 (1970).
[36] P. W. Shor, SIAM journal on computing **26**, 1484 (1997).
[37] M. A. Nielsen and I. L. Chuang, *Quantum computation and quantum information* (Cambridge University Press, 2000).
[38] J. Watrous, Journal of Computer and System Sciences **59**, 281 (1999).
[39] S. Wolfram, *A new kind of science*, (Wolfram Media Inc., Champaign, Ilinois, US, United States, 2002).
[40] W. Maass and P. Orponen, Neural Computation **10**, 1071 (1997).
[41] E. Asarin and A. Bouajjani, in *In Proceedings of the Sixteenth Annual IEEE Symposium on Logic in Computer Science. IEEE* (IEEE Computer Society Press, 2001) pp. 269–278.
[42] D. S. Graça, M. L. Campagnolo, and J. Buescu, Advances in Applied Mathematics **40**, 330 (2008).
[43] Y. Kifer, *Random perturbations of dynamical systems*, Progress in probability and statistics, v. 16 (Birkhäuser, Boston., 1988).
[44] P. Walters, *An Introduction to Ergodic Theory*, Graduate Texts in Mathematics, Vol. 79 (Springer-Verlag, New York, 1982).
[45] K. Petersen, *Ergodic Theory* (Cambridge Univ. Press, 1983).
[46] R. Mañé, *Ergodic theory and differentiable dynamics*, [Results in Mathematics and Related Areas (3)], Vol. 8 (Springer-Verlag, Berlin, 1987) pp. xii+317.
[47] M. Hoyrup and C. Rojas, Information and Computation **207**, 830 (2009).
[48] D. E. Knuth and A. C. Yao, Algorithms and complexity: new directions and recent results , 357 (1976).
[49] M. Braverman, A. Grigo, and C. Rojas, in *Proceedings of the 3rd Innovations in Theoretical Computer Science*

Conference, ITCS '12 (ACM, New York, NY, USA, 2012) pp. 128–141.

[50] See Suplemental Material [`http://www.cs.princeton.edu/~mbraverm/pmwiki/uploads/BRS2015-noise.pdf`, available at `tiny.cc/BRS15`], which includes Refs. [52–60].

[51] M. H. Freedman, A. Kitaev, and Z. Wang, Communications in Mathematical Physics **227**, 587 (2002).

[52] H. Alt, in *Proceedings of the 16th Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1984, Washington, DC, USA* (1984) pp. 466–470.

[53] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions* (Dover Publications, 1965).

[54] I. Binder, M. Braverman, C. Rojas, and M. Yampolsky, Communications in Mathematical Physics. **308**, 743 (2011).

[55] G. Buntrock, C. Damm, U. Hertrampf, and C. Meinel, Mathematical Systems Theory **25**, 223 (1992).

[56] A. Chiu, G. Davida, and B. Litow, RAIRO-Theoretical Informatics and Applications-Informatique Théorique et Applications **35**, 259 (2001).

[57] G. E. Collins, Journal of Symbolic Computation **32**, 467 (2001).

[58] I. M. Gelfand, M. M. Kapranov, and A. V. Zelevinsky, *Discriminants, resultants, and multidimensional determinants*, Mathematics : theory & applications (Birkhäuser, Boston, Basel, Berlin, 1994).

[59] J. Kari and V. Lukkarila, in *Algorithmic Bioprocesses*, Natural Computing Series, edited by A. Condon, D. Harel, J. N. Kok, A. Salomaa, and E. Winfree (Springer Berlin Heidelberg, 2009) pp. 639–660.

[60] C. A. Neff and J. H. Reif, Journal of Complexity **12**, 81 (1996).