Title
Extreme Quantum Memory Advantage for Rare-Event Sampling

Permalink
https://escholarship.org/uc/item/6ff0s7pd

Journal
Physical Review X, 8(1)

ISSN
2160-3308

Authors
Aghamohammadi, Cina
Loomis, Samuel P
Mahoney, John R
et al.

Publication Date
2018

DOI
10.1103/physrevx.8.011025

Peer reviewed

Extreme Quantum Memory Advantage for Rare-Event Sampling

Cina Aghamohammadi,[*] Samuel P. Loomis,[†] John R. Mahoney,[‡] and James P. Crutchfield[§]
Complexity Sciences Center and Physics Department, University of California at Davis,
One Shields Avenue, Davis, California 95616, USA

(Received 9 August 2017; revised manuscript received 28 December 2017; published 13 February 2018)

We introduce a quantum algorithm for memory-efficient biased sampling of rare events generated by classical memoryful stochastic processes. Two efficiency metrics are used to compare quantum and classical resources for rare-event sampling. For a fixed stochastic process, the first is the classical-to-quantum ratio of required memory. We show for two example processes that there exists an infinite number of rare-event classes for which the memory ratio for sampling is larger than r, for any large real number r. Then, for a sequence of processes each labeled by an integer size N, we compare how the classical and quantum required memories scale with N. In this setting, since both memories can diverge as $N \to \infty$, the efficiency metric tracks how fast they diverge. An extreme quantum memory advantage exists when the classical memory diverges in the limit $N \to \infty$, but the quantum memory has a finite bound. We then show that finite-state Markov processes and spin chains exhibit memory advantage for sampling of almost all of their rare-event classes.

DOI: 10.1103/PhysRevX.8.011025

Subject Areas: Quantum Physics,
Quantum Information,
Statistical Physics

I. INTRODUCTION

From earthquakes to financial market crashes, rare events are associated with catastrophe—from decimated social infrastructure and the substantial loss of life to global economic collapse. Though rare, their impact cannot be ignored. Prediction and modeling such rare events is essential to mitigating their effects. However, this is particularly challenging, often requiring huge data sets and massive computational resources, precisely because the events of interest are rare.

Ameliorating much of the challenge, biased or extended sampling [1,2] is an effective and now widely used method for efficient generation and analysis of rare events. The underlying idea is simple to state: transform a given distribution to a new one where previously rare events are now typical. This concept was originally proposed in 1961 by Miller to probe the rare events generated by discrete-time, discrete-value Markov stochastic processes [3]. It has since been extended to address non-Markovian processes [4]. The approach was also eventually adapted to continuous-time first-order Markov processes [5–7]. Today, the statistical analysis of rare events is a highly developed toolkit with broad applications in sciences and engineering [8]. Given this, it is perhaps not surprising that the idea and its related methods appear under different appellations, depending on the research arena. For example, large deviation theory refers to the s-ensemble method [9,10], the exponential tilting algorithm [11,12], or as generating twisted distributions.

In 1997, building on biased sampling, Torrie and Valleau introduced umbrella sampling into Monte Carlo simulation of systems whose energy landscapes have high energy barriers and so suffer particularly from poor sampling [13]. Since then, stimulated by computational problems arising in statistical mechanics, the approach was generalized to Ferrenberg-Swendsen reweighting, later still to weighted histogram analysis [14], and more recently to Wang-Landau sampling [15].

When generating samples for a given stochastic process, one can employ alternative types of algorithms. There are two main types—Monte Carlo or finite-state machine algorithms. Here, we consider finite-state machine algorithms based on Markov chains (MC) [16,17] and hidden Markov models (HMM) [18–20]. For example, if the process is Markovian, one uses MC generators and, in more general cases, one uses HMM generators.

When evaluating alternative approaches, the key questions that arise concern algorithm speed and memory

[*] caghamohammadi@ucdavis.edu
[†] sloomis@ucdavis.edu
[‡] jrmahoney@ucdavis.edu
[§] chaos@ucdavis.edu

efficiency. For example, it turns out that there are HMMs that are always equally or more memory efficient than MCs. There are many finite-state HMMs for which the analogous MC is infinite state [21]. And so, when comparing all HMMs that generate the same process, one is often interested in those that are most memory efficient. For a generic stochastic process, the most memory-efficient classical HMM known currently is the ϵ-machine of computational mechanics [22]. The memory it requires is called the process's "statistical complexity" C_μ [23].

Today, we have come to appreciate that several important mathematical problems can be solved more efficiently using a quantum computer. Examples include quantum algorithms for integer factorization [24], search [25], eigendecomposition [26], and linear system solutions [27]. Not long ago and for the first time, Ref. [28] provided a quantum algorithm that can perform stochastic process sample generation using less memory than the best-known classical algorithms. Recently, using a stochastic process's higher-order correlations, a new quantum algorithm—the q-machine—substantially improved this efficiency and extended its applicability [29]. More detailed analysis and a derivation of the closed-form quantum advantage of the q-machine was given in a sequel [30]. Notably, the quantum advantage has been verified experimentally for a simple case [31].

The following brings together techniques from large deviation theory, classical algorithms for stochastic process generation, computational complexity theory, and the newly introduced quantum algorithm for stochastic process generation to propose a new, memory-efficient quantum algorithm for the biased sampling problem. We show that there can be an extreme advantage in the quantum algorithm's required memory compared to the best-known classical algorithm where the required memory for the classical algorithm grows unboundedly with problem size, but is bounded from above for the quantum algorithm. Three examples are analyzed here. The first is the simple, but now well-studied perturbed coin process. The second is a more physical example—a stochastic process that arises from the Ising next-nearest-neighbor spin system in contact with a thermal reservoir. The third is a sequence of processes generated by a series of Ising N-nearest-neighbor Hamiltonians.

Today, we know of several different sampling problems for which their best quantum algorithm has an advantage compared to the best classical algorithm. These sampling problems fall into two categories. First are those in which the problem is quantum in nature, such as boson sampling [32]. Second are the ones in which the target problem is classical. Function sampling [33] and mixing [34] are in this category.

On the one hand, the advantage for both boson sampling and mixing appears in shorter run times for the quantum algorithm. For the problem of rare-event sampling we study here, the run times for both classical and quantum algorithms are the same. On the other hand, for function sampling, the advantage appears in the smaller required memory; this is similar to our problem. In both boson sampling and function sampling, the advantage appears as an increasing function of problem size. For mixing, it is function of the spectral gap—that is, a property of the problem input—the Markov chain of interest, in that case. The quantum memory advantage we introduce here is both a function of problem size and a property of the input instance.

In the boson sampling problem, a linear system scatters N individual bosons into $M \gg N$ output modes. The goal then is to sample from the output distribution. It is known that, for large N and M, the run time for the quantum algorithm is much smaller than for the classical algorithm, while both algorithms need memory on the order of the required sample size. In mixing, a Markov chain and an initial state are given and the goal is to sample from the stationary distribution over Markov chain's states with some acceptable error margin. Denoting the spectral gap for the Markov chain's transition matrix by δ, the run time for the best-known classical algorithm increases faster than the quantum algorithm when $\delta \to 0$ [35]. As a result, the notion of an advantage is captured by a function of δ. In function sampling, a function $f: X \times Y \to \{0, 1\}$ and a probability distribution $\mathbb{P}(X, Y)$ over $X \times Y$ are given. Alice and Bob start with no inputs. The goal then is to sample X, Y, and Z from the distribution $(\mathbb{P}, f(\mathbb{P}))$, where Alice ends up with X and Bob with Y and Z. Algorithm efficiency is then defined by how much information Alice and Bob must communicate during the algorithm. It turns out that the best-known quantum algorithm has markedly smaller communication costs than the classical. In the function sampling problem, as in many other similar problems, communication cost can be framed as a memory cost, since Alice can always write the message in a memory that Bob reads.

II. CLASSICAL ALGORITHM

The object for which we wish to generate samples is a discrete-time, discrete-value stochastic process [18,36]: a probability space $\mathcal{P} = \{\mathcal{A}^\infty, \Sigma, \mathbb{P}(\cdot)\}$, where $\mathbb{P}(\cdot)$ is a probability measure over the bi-infinite chain $\ldots X_{-2} X_{-1} X_0 X_1 X_2 \ldots$; where each random variable X_i takes values in a finite, discrete alphabet $\mathcal{A}$; and where Σ is the σ-algebra generated by the cylinder sets in $\mathcal{A}^\infty$. For simplicity, we consider only ergodic stationary processes: that is, $\mathbb{P}(\cdot)$ is invariant under time translation $[\mathbb{P}(X_{i_1} X_{i_2} \cdots X_{i_m}) = \mathbb{P}(X_{i_1+n} X_{i_2+n} \cdots X_{i_m+n})$ for all n, $m]$ and over successive realizations.

Sampling or generating a given stochastic process refers to producing a finite realization that comes from the process's probability distribution. There are two main generation (sampling) problems: sequential generation

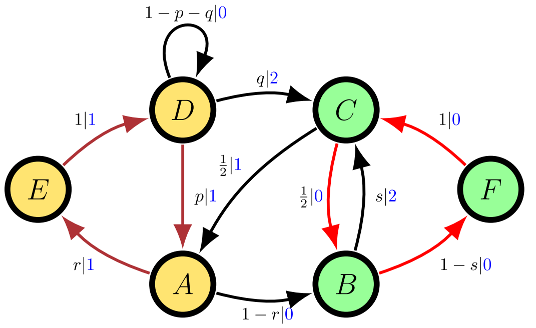

FIG. 1. Hidden Markov model generator of a stochastic process with infinite-range statistical dependencies that requires a HMM with only six states. To generate the same process via a Markov chain requires one with an infinite number of states and so infinite memory.

and simultaneous generation [37]. In sequential generation or one-shot sampling, the goal is to generate one long sample from the given process. However, in simultaneous generation, the goal is to generate $M \gg 1$ realizations of a process simultaneously, each of which is statistically independent of the others.

Generally, generating a process via its probability measure $\mathbb{P}(\cdot)$ is impossible because of the vast number of allowed realizations and, as a result, this prosaic approach requires an unbounded amount of memory. Fortunately, there are more compact ways than specifying in full the probability measure on the sequence sigma algebra. This recalls the earlier remark that HMMs can be arbitrarily more compact than alternative algorithms for the task of generation.

A HMM is specified by a tuple $\{\boldsymbol{\mathcal{S}}, \mathcal{A}, \{T^{(x)}, x \in \mathcal{A}\}\}$. In this, $\boldsymbol{\mathcal{S}}$ is a finite set of states; $\mathcal{A}$ is a finite alphabet; and $\{T^{(x)}, x \in \mathcal{A}\}$ is a set of $|\boldsymbol{\mathcal{S}}| \times |\boldsymbol{\mathcal{S}}|$ substochastic symbol-labeled transition matrices whose sum $T = \sum_{x\in\mathcal{A}} T^{(x)}$ is a stochastic matrix.

As an example, consider the HMM state-transition diagram shown in Fig. 1, where $\boldsymbol{\mathcal{S}} = \{A, B, C, D, E, F\}$; $\mathcal{A} = \{0, 1, 2\}$; and we have three 6×6 substochastic matrices $T^{(0)}$, $T^{(1)}$, and $T^{(2)}$. Each edge is labeled $p|x$, denoting the transition probability p and a symbol $x \in \mathcal{A}$, which is emitted during the transition. In this HMM, of the two edges exiting state C, one enters state B and the other enters state A. The edges from C to A and C to B are labeled by $\frac{1}{2}|1$ and $\frac{1}{2}|0$. This simply means that, if the HMM is in the state C, then, with probability $\frac{1}{2}$, it goes to the state A and emits the symbol 1, and, with probability $\frac{1}{2}$, it goes to the state B and emits symbol 0. Following these transition rules in succession generates realizations in the HMM's process.

How does this generation method compare to generating realizations of the same process via a finite Markov chain? (Recall that states in a MC are not hidden: $\mathcal{A} = \boldsymbol{\mathcal{S}}$.) It turns out that this cannot be implemented, since generating a symbol can depend on the infinite history. That is, the process has infinite Markov order. As a result, to generate a realization using a Markov chain, one needs an infinite number of Markovian states. In other words, implementing the Markov chain algorithm to generate process samples on a conventional computer requires an infinite amount of memory.

To appreciate the reason behind the process's infinite Markov order, refer to the HMM in Fig. 1. There are two length-3 state loops consisting of the edges colored red (the right side of the state-transition diagram) and those colored maroon (the left side). Note that, if the HMM generates n 1's in a row, we will not know the HMM's current state, only that it is A, D, or E. This state uncertainty (entropy) is bounded away from 0. The observation holds for the other loop and its sequences of symbol 0 and the consequent ambiguity among states B, C, and F. Thus, there exist process realizations from which we cannot determine the future statistics, independent of the number of symbols seen. This means that the process statistics depend on infinite past sequences—the process has infinite Markov order. We repeat, for emphasis, that implementing a MC algorithm for this requires infinite memory. The contrast with the finite HMM method is an important lesson: HMMs are strictly more powerful generators, as a class of algorithms, than Markov chain generators.

For any given process $\mathcal{P}$, there are an infinite number of HMMs that generate it. Therefore, one is compelled to ask, which algorithm requires the least memory for implementation? To appreciate the answer, let us first address how much state memory one needs to run a HMM.

Consider sequential generation, in which the goal is to produce a very long realization of a process. For this, we use one computer with a code that runs the algorithm (HMM). At each step, the computer must memorize the current HMM state. Assuming the HMM has N states, this requires $\log_2(N)$ bits of memory. As a result, if one wishes to implement one-shot sampling using the minimum required memory, then, over all the process's HMM generators, one needs to find the generator with the minimum number of states.

Here, though, we are interested in simultaneous generation, for which the goal is to simultaneously generate $M \gg 1$ process realizations, each of which is statistically independent of the others. The effective implementation uses M computers, each with the above code. Similar to the sequential problem, each computer must memorize the current state of its HMM. If each computer uses its own memory, each needs $\log_2(N)$ bits of memory, as before. The total memory is then $M\log_2(N)$ bits. However, we can reduce the required memory by using one large memory shared among the computers. Figure 2 depicts this schematically. In this way, according to Shannon's coding theorem [38], we can encode HMM states to reduce the amount of memory down to $MH(\mathcal{S}) \le M\log_2(N)$ bits, where $H(\mathcal{S})$ is the Shannon entropy of the probability

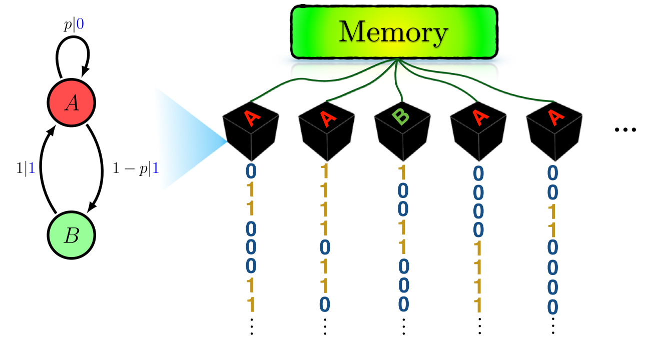

FIG. 2. Left: Even process ϵ-machine. Right: Schematic of simultaneous generation problem. Each black box contains an even process generator. They all share the same memory for tracking the individual generator states.

distribution over HMM's states. The memory per sample is then just $H(\mathcal{S})$. As a result, if one needs to do simultaneous sampling of a given process using the minimum required memory, over all its HMM generators, one needs to find the HMM whose state distribution has the minimum Shannon entropy.

For both one-shot and simultaneous sampling, the best-known implementation, and provably the optimal predictor, is known as the ϵ-machine M [22,39]. Its states are called causal states; we denote this as set $\boldsymbol{\mathcal{S}}$. The average memory required for $M(\mathcal{P})$ to sequentially sample process $\mathcal{P}$ is given by the process's statistical complexity $C_\mu(\mathcal{P})$ [23]. To calculate it,

(1) Compute the stationary distribution π over causal states. π is the left eigenvector of the state-transition matrix T with eigenvalue 1: $\pi T = \pi$.

(2) Calculate the state's Shannon entropy $H[\boldsymbol{\mathcal{S}}] = -\sum_{\sigma\in\boldsymbol{\mathcal{S}}}\pi(\sigma)\log_2 \pi(\sigma)$.

$C_\mu = H[\boldsymbol{\mathcal{S}}]$ measures the (ensemble average) memory required for the simultaneous sampling of the process.

Another important, companion measure is h_μ, the process's metric entropy (or Shannon entropy rate) [40]:

$$h_\mu(\mathcal{P}) = -\lim_{n\to\infty}\frac{1}{n}\sum_{w\in\mathcal{A}^n}\mathbb{P}(w)\log_2\mathbb{P}(w).$$

Although it is sometimes confusing, it is important to emphasize that h_μ describes randomness in realizations, while C_μ is the average memory required to generate them.

III. QUANTUM MEMORY ADVANTAGE

Recently, it was shown that a quantum algorithm for process generation can use less memory than the best-known classical algorithm (ϵ-machine) [28]. By accounting for a process's higher-order correlations, a new quantum algorithm—the q-machine—was introduced that substantially improved the original and is, to date, the most memory-efficient quantum algorithm known for process generation [29]. We refer to the ratio of required classical memory C_μ to quantum memory C_q as the "quantum memory advantage." Closed-form expressions for the quantum memory advantage are given in Ref. [30].

Importantly, the quantum advantage was recently verified experimentally for the simple perturbed coins process [31]. It was also discovered that the q-machine can confer an extreme quantum memory advantage. For example, when generating configurations in a Dyson-type spin model with N-nearest-neighbor interactions at temperature T, the quantum advantage scales as $NT^2/\log_2 T$ [41,42]. Another example of an extreme quantum memory advantage has appeared recently in the simulation of continuous-time stochastic processes [43].

One consequence of quantum advantage arises in model selection, where the basic question "Which process is simpler?" no longer has a well-defined answer [44]. Recently, this phenomenon has been experimentally observed [45]. Statistical inference of models for stochastic systems often involves controlling for model size or memory. So, quantum statistical inference may see large improvements. Another consequence of this advantage in the context of simulations is that reduced memory can reduce the heat dissipation of simulation [46].

The following determines the quantum advantage in biased sampling of a process's rare events. In particular, we develop tools to determine how the memory requirements of classical and quantum algorithms vary over rare-event classes.

IV. QUANTUM ALGORITHM

We define a stochastic process $\mathcal{P}$'s q machine by $\mathrm{Q}(\mathcal{P}) = \{\mathcal{H}, \mathcal{A}, \{K_x, x\in\mathcal{A}\}\}$, where $\mathcal{H}$ denotes the Hilbert space with dimension $|\boldsymbol{\mathcal{S}}|$ in which quantum states reside; $\mathcal{A}$ is the same alphabet as that of the given process; and $\{K_x, x\in\mathcal{A}\}$ is a set of Kraus operators that we use to specify the measurement protocol for states [47]. (We adopt a particular form for the Kraus operators. In general, they are not unique.) Assume that we have the quantum state (density matrix) $\rho_0 \in \mathcal{B}(H)$ in hand. We perform a measurement by applying Kraus operators and, as a result, measure X. The probability of yielding symbol $X = x$ is

$$\mathbb{P}(X = x_0|\rho_0) = \mathrm{tr}(K_{x_0}\rho_0 K_{x_0}^\dagger).$$

After measurement with outcome $X = x_0$, the new quantum state is

$$\rho_1 = \frac{K_{x_0}\rho_0 K_{x_0}^\dagger}{\mathrm{tr}(K_{x_0}\rho_0 K_{x_0}^\dagger)}.$$

Repeating these measurements generates a stochastic process. The process potentially could be nonergodic, depending on the initial state ρ_0. However, starting the q-machine in the stationary state defined by

$$\rho_s = \sum_{x\in\mathcal{A}} K_x \rho_s K_x^\dagger$$

and repeatedly measuring generates a stationary stochastic process over $x \in \mathcal{A}$. For any given process, ρ_s can be calculated by the method introduced in Ref. [30].

Our immediate goal is to design a quantum generator of a given classical process. (Section VI will then take the given process to represent a rare-event class of another process.) For now, we start with the process' ϵ-machine. The construction consists of three steps, as follows.

First step: Map every causal state $\sigma_i \in \boldsymbol{\mathcal{S}}$ to a pure quantum state $|\eta_i\rangle$. Each signal state $|\eta_i\rangle$ encodes the set of length-R sequences that may follow σ_i, as well as each corresponding conditional probability:

$$|\eta_i\rangle \equiv \sum_{w\in\mathcal{A}^R} \sqrt{\mathbb{P}(w|\sigma_i)}|w\rangle,$$

where w denotes a length-R sequence, $\mathbb{P}(w|\sigma_i) = \mathbb{P}(X_0 \cdots X_{R-1} = w|\mathcal{S}_0 = \sigma_i)$, and R is the process' Markov order. The resulting Hilbert space is $\mathcal{H}_w$ with size $|\mathcal{A}|^R$, the number of length-R sequences, with basis elements $|w\rangle = |x_0\rangle \otimes \cdots \otimes |x_{R-1}\rangle$, where the states $\{|x\rangle : x \in \mathcal{A}\}$ form an orthonormal basis. From here on out, assume all $|\eta_i\rangle$'s are linearly independent. (The procedure requires only slight modification for linearly dependent signal states and, in any case, does not affect the results.) As a result, one defines $|\boldsymbol{\mathcal{S}}|$ new states $|\xi_i\rangle$ that reside in a Hilbert space of size $|\boldsymbol{\mathcal{S}}|$. (This is much smaller than the $|\eta_i\rangle$'s Hilbert space, which has size $|\mathcal{A}|^R$.) Moreover, the $|\xi_i\rangle$'s have the same pairwise overlaps as the $|\eta_i\rangle$'s. That is, for all i, j,

$$\langle \xi_i|\xi_j\rangle = \langle \eta_i|\eta_j\rangle.$$

The authors of Ref. [30] developed a method to calculate all the overlaps $\langle \eta_i|\eta_j\rangle$ for a given process in closed form. Since the $|\eta_i\rangle$'s are linearly independent, one can use the overlaps to construct the $|\xi_i\rangle$'s.

Second step: Form a matrix Ξ by assembling the signal states:

$$\Xi = [\,|\xi_0\rangle \quad |\xi_1\rangle \quad \cdots \quad |\xi_{|\boldsymbol{\mathcal{S}}|-1}\rangle\,].$$

Define $|\boldsymbol{\mathcal{S}}|$ new bra states $|\widetilde{\xi_i}\rangle$:

$$\begin{bmatrix} \langle\widetilde{\xi_0}| \\ \langle\widetilde{\xi_1}| \\ \cdots \\ \langle\widetilde{\xi_{|\boldsymbol{\mathcal{S}}|-1}}| \end{bmatrix} = \Xi^{-1}.$$

That is, we design the new bra states such that we obtain the identity

$$\begin{bmatrix} \langle\widetilde{\xi_0}| \\ \langle\widetilde{\xi_1}| \\ \cdots \\ \langle\widetilde{\xi_{|\boldsymbol{\mathcal{S}}|-1}}| \end{bmatrix} [\,|\xi_0\rangle \quad |\xi_1\rangle \quad \cdots \quad |\xi_{|\boldsymbol{\mathcal{S}}|-1}\rangle\,] = \mathrm{I}.$$

Third step: Define $|\mathcal{A}|$ Kraus operators K_x for all $x \in \mathcal{A}$ via

$$K_x = \sum_{i,j} \sqrt{T^x_{ij}}\,|\xi_j\rangle\langle\widetilde{\xi_i}|.$$

(Appendix A proves the completeness of these operators.)

By applying Kraus operators repeatedly, one generates the target stochastic process. For example, consider the case in which the q-machine is in state $|\xi_i\rangle\langle\xi_i|$ and we apply the Kraus operators. Then, if we do not make a measurement, the next state is $\sum_{j,x} T^x_{ij}|\xi_j\rangle\langle\xi_j|$. Let us say, though, that we make a measurement and the result is x. The next state is $|\xi_j\rangle\langle\xi_j|$, where j is the unique index and where T^x_{ij} is nonzero. Appendix B shows that the stationary state is

$$\rho_s = \sum_i \pi_i |\xi_i\rangle\langle\xi_i|.$$

The Hilbert space in which the algorithm operates has dimension $|\boldsymbol{\mathcal{S}}|$. Since the operation is not unitary in this space and measurements are not projective, one may argue that $|\boldsymbol{\mathcal{S}}|$ is not the actual size of the Hilbert space needed to physically implement the algorithm. However, it was recently shown that the algorithm presented here can be implemented by unitary operations and projective measurements in a Hilbert space with dimension $|\boldsymbol{\mathcal{S}}||\mathcal{A}|$ [48]. This new result gives an experimental implementation of our algorithm.

Using the quantum generator $Q(\mathcal{P})$, the required average memory for simultaneous generation of process $\mathcal{P}$ is $C_q(\mathcal{P}) = S(\rho_s)$, where $S(\rho) = -\mathrm{tr}(\rho \log \rho)$ denotes the von Neumann entropy [47]. (Von Neumann entropy is a well-accepted measure of quantum advantage in communication games [49].) Since the algorithm can be implemented by unitary operations and projective measurements, the entropy of the quantum model stays constant at C_q at all times during the simulation process. This confirms that C_q is a valid measure of memory, while the minimal dimensions needed for sequential (one-shot) generation of $\mathcal{P}$ are at most $|\boldsymbol{\mathcal{S}}||\mathcal{A}|$.

Comparing memory efficiencies of classical and quantum algorithms requires an efficiency metric. Depending on the setting, there are two that one can use. In the single-process case, $\mathcal{P}$ is given, and memory efficiency is defined as the ratio of required memory for the classical algorithm to the quantum algorithm. Here, since we only explore finite-size stochastic processes, both memories are finite, and the ratio is a good quantitative efficiency measure.

In the multiprocess case, a series of stochastic processes is given, with each process labeled by an integer N that measures process size. Then, memory efficiency is defined by how the memory scales in N for the classical algorithm compared to the quantum. This metric is closer to relative-complexity definitions familiar in computation complexity theory. In the present case, since both memories are allowed to diverge when $N \to \infty$, the quantitative measure of efficiency tracks how fast they diverge. We say we have extreme quantum memory advantage when the classical memory diverges as $N \to \infty$, but quantum memory converges to a finite quantity.

V. TYPICAL REALIZATIONS

At this point, we established classical and quantum representations of processes and characterized their respective memory requirements. Using this, our purpose now shifts to monitor classical and quantum resources required to generate rare-event classes of a process's realizations. Our first task is to review the theory of typical events and their complement—rare events.

The concept of a stochastic process is quite general. Any physical system that exhibits stochastic dynamics in time or space may be thought of as generating a stochastic process. In the spatial setting, one considers not time evolution, but rather the spatial "dynamic." For example, consider a one-dimensional noninteracting Ising spin-$\frac{1}{2}$ chain with classical Hamiltonian $H = -\sum_{i=1}^{n} h\sigma_i$ in contact with a thermal reservoir at temperature T. After thermalizing, a spin configuration at one instant of time may be thought of as having been generated by a process that scans the lattice left to right (or, equivalently, right to left). The probability distribution over these spatial-translation-invariant configurations defines a stationary stochastic process—a simple Markov random field.

For $n \gg 1$, one can ask for the probability of seeing k up-spins. The Strong Law of Large Numbers [50] guarantees that for large n, the ratio k/n almost surely converges to $p_\uparrow = \frac{1}{2}[1 + \tanh(h/k_\mathrm{B}T)]$. That is,

$$\mathbb{P}\left(\lim_{n\to\infty} \frac{k}{n} = p_\uparrow\right) = 1.$$

Informally, a typical sequence is one that has close to $p_\uparrow n$ up-spins. However, this does not preclude seeing other kinds of long runs, e.g., all up-spins or all down-spins. It simply means that the latter are rare events, compared to the typical ones.

Now, let us formally define the concept of typical realizations and, consequently, rare ones. Consider a given process $\mathcal{P}$ and let $\mathcal{A}^n$ denote the set of all possible length-n realizations. Then, for an arbitrary $0 < \epsilon \ll 1$, the process' typical set [38,51,52] is defined:

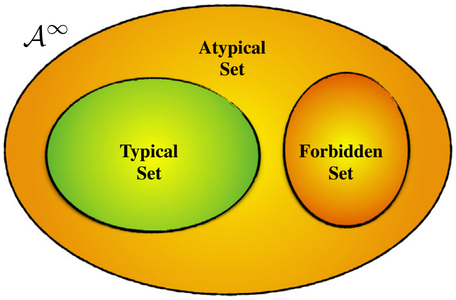

FIG. 3. For a given process, the space $\mathcal{A}^\infty$ of all sequences is partitioned into forbidden sequences, sequences in the typical set, and sequences neither forbidden nor typical—the atypical or rare sequences.

$$A^n_\epsilon \equiv \{w : 2^{-n(h_\mu+\epsilon)} \le \mathbb{P}(w) \le 2^{-n(h_\mu-\epsilon)}, w \in \mathcal{A}^n\}, \quad (1)$$

where h_μ is the process' Shannon entropy rate, introduced above.

According to the Shannon-McMillan-Breiman theorem [53–55], for a given $\epsilon \ll 1$, sufficiently large n^*, and $w \in \mathcal{A}^n$,

$$\mathbb{P}(w \notin A^n_\epsilon) \le \epsilon, \quad (2)$$

for all $n \ge n^*$. There are two important lessons here. First, from Eq. (1) we see that all sequences in the typical set have approximately the same probability. More precisely, the probability of typical sequences decays at the same exponential rate. The following adapts this to use decay rates to identify distinct sets of rare events. Second, coming from Eq. (2), for large n, the probability of sequences falling outside the typical set is close to zero—these are the sets of rare sequences.

Another important consequence of the theorem is that sequences generated by a stationary ergodic process fall into one of three partitions; see Fig. 3. The first contains sequences that are never generated; they fall into the forbidden set. For example, the HMM in Fig. 1 never generates sequences that have consecutive 2s. The second partition consists of those in the typical set—the set with probability close to 1, as in Eq. (1). The last contains sequences in a family of atypical sets—realizations that are rare to different degrees. We now refine this classification by dividing the atypical sequences into identifiable subsets, each with their own characteristic rarity.

Mirroring the familiar Boltzmann weight in statistical physics [56], in the $n \to \infty$ limit, we define the subsets $\Lambda^\mathcal{P}_u \subset \mathcal{A}^\infty$ for a process $\mathcal{P}$ as

$$\Lambda^\mathcal{P}_{u,n} = \left\{w : -\frac{\log_2 \mathbb{P}(w)}{n} = u, w \in \mathcal{A}^n\right\} \quad \text{and}$$
$$\Lambda^\mathcal{P}_u = \lim_{n\to\infty} \Lambda^\mathcal{P}_{u,n}. \quad (3)$$

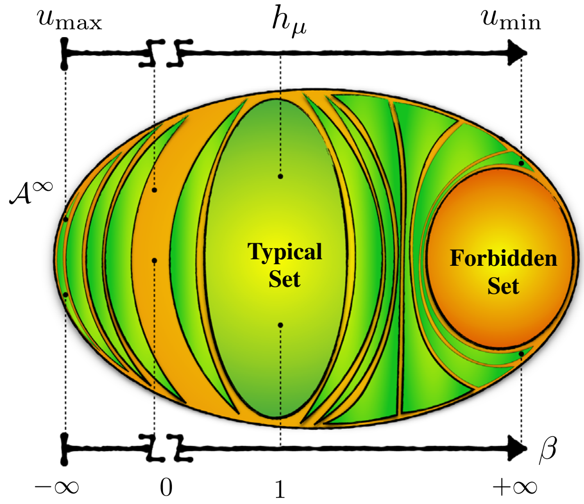

FIG. 4. Space of all sequences $\mathcal{A}^\infty$ partitioned into $\Lambda_u^{\mathcal{P}}$'s—isoenergy density or bubbles with equal probability-decay rates—in which all sequences in the same $\Lambda_u^{\mathcal{P}}$ have the same energy density u. The typical set is one such bubble with energy equal to the Shannon entropy rate: $u = h_\mu$. Another important class is the forbidden set, in which all sequences do not occur. The forbidden set can also be interpreted as the subset of sequences with infinite positive energy. By applying the map $\mathcal{B}_\beta$ to the process and changing β continuously from $-\infty$ to $+\infty$ (excluding $\beta = 0$), one can generate any rare class of interest $\Lambda_u^{\mathcal{P}}$. $\beta \to -\infty$ corresponds to the most probable sequences with the largest energy density $u_{\max}$, $\beta = 1$ corresponds to the typical set, and $\beta \to +\infty$ corresponds to the least probable sequences with the smallest energy density $u_{\min}$.

This partitions $\mathcal{A}^\infty$ into disjoint subsets $\Lambda_u^{\mathcal{P}}$, in which all $w \in \Lambda_u^{\mathcal{P}}$ have the same probability decay rate u. Physics vernacular would speak of the sequences having the same energy density u. (u, considered as a random variable, is sometimes called a "self-process" [57].) Figure 4 depicts these subsets as "bubbles" of equal energy density. Equation (1) says that the typical set is that bubble with energy equal to the process' Shannon entropy rate: $u = h_\mu$. All the other bubbles contain rare events, with some rarer than others. They exhibit faster or slower probability decay rates.

Employing a process' HMM to generate realizations produces sequences in the typical set with probability close to 1 and, rarely, atypical sequences. Imagine that one is interested in a particular class $\Lambda_u^{\mathcal{P}}$ of rare sequences, say, those with energy u. (One might be concerned about the class of large-magnitude earthquakes or the emergence of major instabilities in the financial markets, for example.) How can one efficiently generate these rare sequences? We now show that there is a new process $\mathcal{P}^u$ whose typical set is $\Lambda_u^{\mathcal{P}}$, and this returns us directly to the challenge of biased sampling.

VI. BIASED SAMPLING

Consider a finite set of configurations $\{c_i\}$ with probabilities specified by distribution $\mathbb{P}(\cdot)$ and an associated set $\{\omega_i\}$ of weighting factors. Consider the procedure of reweighting that introduces a new distribution $\tilde{\mathbb{P}}(\cdot)$ over configurations where

$$\tilde{\mathbb{P}}(c_i) = \frac{\mathbb{P}(c_i)\exp(\omega_i)}{\sum_i \mathbb{P}(c_i)\exp(\omega_i)}.$$

Given a process $\mathcal{P}$ and its ϵ-machine $M(\mathcal{P})$, how do we construct an ϵ-machine $M(\mathcal{P}^u)$ that generates $\mathcal{P}$'s atypical sequences at some energy density $u \neq h_\mu$ or, as we denoted it, the set $\Lambda_u^{\mathcal{P}}$? Here, we answer this question by constructing a map $\mathcal{B}_\beta : \mathcal{P} \to \mathcal{P}_\beta$ from the original process $\mathcal{P}$ to a new one $\mathcal{P}_\beta$. The map is parametrized by $\beta \in \mathbb{R}/\{0\}$, which indexes the rare set of interest. (We use β for convenience here, but it is related to u by a function introduced shortly.) Both processes $\mathcal{P} = \{\mathcal{A}^\infty, \Sigma, \mathbb{P}(\cdot)\}$ and $\mathcal{P}_\beta = \{\mathcal{A}^\infty, \Sigma, \mathbb{P}_\beta(\cdot)\}$ are defined on the same measurable sequence space. The measures differ, but their supports (allowed sequences) are the same. For simplicity we refer to $\mathcal{B}_\beta$ as the "β-map."

Assume we are given $M(\mathcal{P}) = \{\boldsymbol{\mathcal{S}}, \mathcal{A}, \{T^{(x)}, x \in \mathcal{A}\}\}$. We showed that for every probability decay rate or energy density u, there exists a particular β such that $M(\mathcal{P}_\beta)$ typically generates the words in $\Lambda_{u,n}^{\mathcal{P}}$ for large n [37]. The β-map that establishes this is calculated by a construction that relates $M(\mathcal{P})$ to $M(\mathcal{P}_\beta) = \{\boldsymbol{\mathcal{S}}, \mathcal{A}, \{\mathbf{S}_\beta^{(x)}, x \in \mathcal{A}\}\}$—the HMM that generates $\mathcal{P}_\beta$:

(1) For each $x \in \mathcal{A}$, construct a new matrix $\mathbf{T}_\beta^{(x)}$ for which $[\mathbf{T}_\beta^{(x)}]_{ij} = [\mathbf{T}^{(x)}]_{ij}^\beta$.
(2) Form the matrix $\mathbf{T}_\beta = \sum_{x\in\mathcal{A}} T_\beta^{(x)}$.
(3) Calculate $\mathbf{T}_\beta$'s maximum eigenvalue $\hat{\lambda}_\beta$ and corresponding right eigenvector $\hat{\mathbf{r}}_\beta$.
(4) For each $x \in \mathcal{A}$, construct new matrices $\mathbf{S}_\beta^{(x)}$ for which

$$(\mathbf{S}_\beta^{(x)})_{ij} = \frac{(\mathbf{T}_\beta^{(x)})_{ij}(\hat{\mathbf{r}}_\beta)_j}{\hat{\lambda}_\beta(\hat{\mathbf{r}}_\beta)_i}.$$

Having constructed the new process $\mathcal{P}_\beta$ by introducing its generator, we use the latter to produce some rare set of interest $\Lambda_{u,n}^{\mathcal{P}}$.

Theorem 1. In the limit $n \to \infty$, within the new process $\mathcal{P}_\beta$, the probability of generating realizations from the set $\Lambda_{u,n}^{\mathcal{P}}$ converges to 1:

$$\lim_{n\to\infty} \mathbb{P}_\beta(\Lambda_{u,n}^{\mathcal{P}}) = 1,$$

where

$$u = \beta^{-1}\left(h_\mu(\mathcal{P}_\beta) - \log_2 \hat{\lambda}_\beta\right). \tag{4}$$

In addition, in the same limit, the process $\mathcal{P}_\beta$ assigns equal energy densities over all the members of the set $\Lambda_{u,n}^{\mathcal{P}}$.

Proof.—See Ref. [37].

As a result, for large n, the process $\mathcal{P}_\beta$ typically generates the set $\Lambda^{\mathcal{P}}_{u,n}$ with the specified energy u. The process $\mathcal{P}_\beta$ is sometimes called the auxiliary, driven, or effective process [58–60]. Examining the form of the energy in Eq. (4), one can prove that there is a one-to-one relationship between β and u. So, we can equivalently denote the process $\mathcal{P}_\beta$ by $\mathcal{P}^u$. More formally, every word in $\Lambda^{\mathcal{P}}_u$ with probability measure 1 is in the typical set of process $\mathcal{P}_\beta$.

The β-map construction guarantees that the HMMs $M(\mathcal{P})$ and $M(\mathcal{P}_\beta)$ have the same states and transition topology: $(\mathbf{T}^{(x)}_\beta)_{ij} \neq 0 \Leftrightarrow (\mathbf{S}^{(x)}_\beta)_{ij} \neq 0$. The only difference is in their transition probabilities. $M(\mathcal{P}_\beta)$ is not necessarily an ϵ-machine—the most memory-efficient classical algorithm that generates the process. Typically, though, $M(\mathcal{P}_\beta)$ is an ϵ-machine. Moreover, there are only finitely many βs for which it is not. (More detailed development along these lines will appear in a sequel.)

VII. QUANTUM AND CLASSICAL MEMORY FOR BIASED SAMPLING

Having introduced the necessary background to compare classical versus quantum models and to appreciate typical versus rare realizations, we are ready to investigate the quantum memory advantage when generating a given process' rare events.

The last section concluded that the memory required by the classical algorithm to generate $\mathcal{P}$'s rare sequences with energy density u is

$$C_\mu(\mathcal{P}_\beta) = C_\mu[\mathcal{B}_\beta(\mathcal{P})],$$

where u and β are related via $u = \beta^{-1}(h_\mu(\mathcal{P}_\beta) - \log_2 \hat{\lambda}_\beta)$. Similarly, the memory required by the quantum algorithm to generate the rare class with energy density u is

$$C_q[\mathcal{B}_\beta(\mathcal{P})].$$

For simplicity, we denote these two quantities as $C_\mu(\beta) \equiv C_\mu(\mathcal{P}_\beta)$ and $C_q(\beta) \equiv C_q(\mathcal{P}_\beta)$.

The following analyzes the advantage for three example processes—two in the single-process setting and one in the multiprocess setting. For the first two, we consider particular given stochastic processes and study the advantage (memory ratio) as the metric of memory efficiency. In the third example, we consider a series of stochastic processes labeled by their size N and compare how both classical and quantum memories scale with N. Comparing the scaling is the metric for memory efficiency.

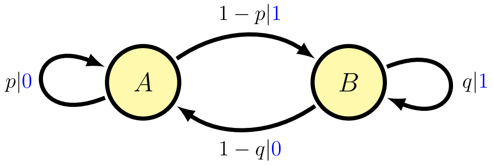

FIG. 5. ϵ-Machine generator of the Perturbed Coins Process. Edges are labeled with conditional transition probabilities and emitted symbols. For example, for the self-loop on state A, $p|0$ indicates that the transition is taken with probability $\mathbb{P}(0|A) = p$, and the symbol 0 is emitted.

A. Quantum memory advantage for a simple Markov process

Let us start in the single-process setting in which an individual stochastic process is given. Consider the case where we have two biased coins, call them A and B, and each has a different bias p and $1-q$ both for heads (symbol 0), respectively. When we flip a coin, if the result is heads, then on the next flip we choose coin A. If the result is tails, we choose coin B. Flipping the coins over and over again results in a process $\mathcal{P}^{\mathrm{pc}}$ called the Perturbed Coins Process [28]. Figure 5 shows the process' ϵ-machine generator $M(\mathcal{P}^{\mathrm{pc}})$, where $\boldsymbol{\mathcal{S}} = \{A, B\}$ and $\mathcal{A} = \{0, 1\}$.

One can also generate this process with a q-machine $Q(\mathcal{P}^{\mathrm{pc}})$. Using the construction introduced in Sec. IV, we see that the q-machine has two Kraus operators corresponding to symbols 0 and 1:

$$K_0 = \begin{bmatrix} \sqrt{p} & 0 \\ \sqrt{1-p} & 0 \end{bmatrix}$$
$$K_1 = \begin{bmatrix} 0 & \sqrt{1-q} \\ 0 & \sqrt{q} \end{bmatrix}.$$

For its stationary state distribution, we have

$$\rho_s = \frac{1}{2-p-q}\begin{bmatrix} 1-q & \alpha \\ \alpha & 1-p \end{bmatrix},$$

where $\alpha = (1-q)\sqrt{p(1-p)} + (1-p)\sqrt{q(1-q)}$. Calculation details are given in Appendix C.

Figure 6 shows the classical and quantum memory costs to generate rare realizations: $C_\mu(\beta)$ and $C_q(\beta)$ versus β for different β-classes. Surprisingly, the two costs exhibit completely different behaviors. For example, $\lim_{\beta\to 0} C_q = 0$, while $\lim_{\beta\to 0} C_\mu = 1$. More interestingly, as the inset demonstrates, even though both $C_\mu(\beta)$ and $C_q(\beta)$ vanish exponentially fast, in the limit of $\beta \to \infty$, $C_q(\beta)$ goes to zero noticeably faster.

We define the memory efficiency $\eta(\beta)$ of biased sampling in the single-process setting as the ratio of classical to quantum memory:

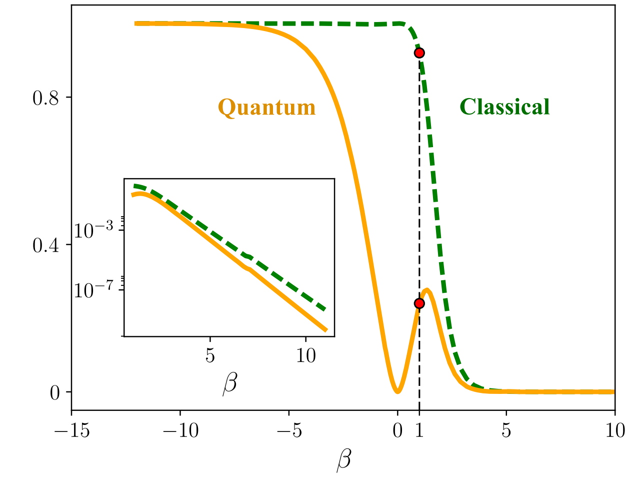

FIG. 6. Classical memory $C_\mu(\beta)$ and quantum memory $C_q(\beta)$ versus β for biased sampling of the Perturbed Coins Process' rare sequence classes: See Fig. 5, with $p = 0.6$ and $q = 0.8$. As the inset shows, for large β, both classical and quantum memories decay exponentially with β, but the quantum memory decays faster. The vertical dashed black line and two red markers at $\beta = 1$ show the memory costs to generate typical sequences.

$$\eta(\beta) \equiv \frac{C_\mu(\beta)}{C_q(\beta)}.$$

Figure 7 graphs $\eta(\beta)$, revealing how it divides into three distinct scaling regimes.

First, for small $|\beta|$, the memory ratio scales as $\mathcal{O}(\beta^{-2})$. Second, for large positive β, the memory ratio scales exponentially $\mathcal{O}[\exp(c\beta)]$ as one increases β, where c is a function of p and q. Third, for large negative β, there is no quantum advantage. Since we are analyzing finite-state processes, this regime appears and is the analog of population inversion. So, formally, there are β-class events with negative β.

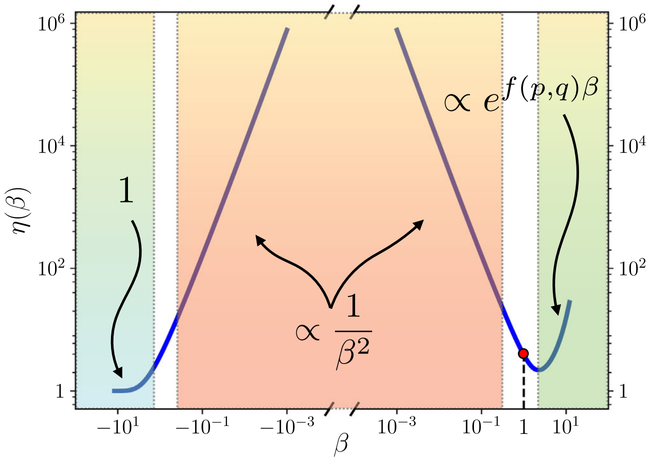

FIG. 7. The classical-to-quantum memory ratio for generating rare realizations of the Perturbed Coins Process with $p = 0.6$ and $q = 0.8$ when employing its q-machine instead of its (classical) ϵ-machine. Three different advantages occur: (i) near $\beta = 0$ the ratio scales as $\mathcal{O}(\beta^{-2})$; (ii) for large positive β, it scales exponentially with β, $\mathcal{O}\{\exp[f(q,p)\beta]\}$; and (iii) no advantage occurs at large negative β. The vertical dashed black line and red marker at $\beta = 1$ show the memory advantage for generating typical sequences.

Such is the quantum advantage for the perturbed coins process at $p = 0.6$ and $q = 0.8$. The features exhibited—the different scaling regimes—are generic for any $p > 1 - q$. Moreover, for Perturbed Coins Processes with $p < 1 - q$, the positive and negative low-temperature behaviors switch.

B. Quantum memory advantage for next-nearest-neighbor spin systems

Again, consider the single-process setting in which an individual stochastic process is given. Let us, however, analyze the quantum advantage in a more familiar physical system. Consider a general one-dimensional ferromagnetic next-nearest-neighbor Ising spin-$\frac{1}{2}$ chain [61,62] defined by the Hamiltonian

$$\mathcal{H} = -\sum_i \left(s_i s_{i+1} + \frac{1}{4} s_i s_{i+2} \right), \qquad (5)$$

in contact with the thermal bath at temperature $k_B T = 1$. The spin s_i at site i takes on values $\{+1, -1\}$.

After thermalizing, a spin configuration at one instant of time may be thought of as having been generated left to right (or, equivalently, right to left) along the lattice. The probability distribution over these spatial-translation-invariant configurations defines a stationary stochastic process. Equations (84)–(91) in Ref. [63] showed that for any finite and nonzero temperature T, this process has Markov order $R = 2$. More to the point, the ϵ-machine that generates this process has four causal states and those states are in one-to-one correspondence with the set of length-2 spin configurations.

Figure 8 displays the parametrized ϵ-machine that generates this family of spin-configuration processes. To simulate the process, the generator only needs to remember the last two spins generated. This means the ϵ-machine has four states, $\downarrow\downarrow$, $\downarrow\uparrow$, $\uparrow\downarrow$, and $\uparrow\uparrow$. If the last two observed spins are $\uparrow\uparrow$, for example, then the current state is $\uparrow\uparrow$. We denote the probability of generating a $\downarrow$ spin given that the previous two spins were $\uparrow\uparrow$ by $p^{\downarrow}_{\uparrow\uparrow}$. If the generator is in the $\uparrow\uparrow$ state and generates a $\downarrow$ spin, then the generator state changes to $\uparrow\downarrow$.

To determine the ϵ-machine transition probabilities $\{T^{(x)}\}_{x\in\mathcal{A}}$, we first compute the transfer matrix V for the Hamiltonian of Eq. (5) at temperature T and then extract conditional probabilities, following Ref. [63] and the appendix of Ref. [41].

What are the classical and quantum memory costs for bias sampling of the rare spin-configuration class with

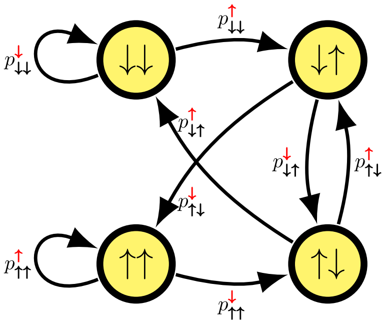
$p^{\downarrow}_{\downarrow\downarrow}$
↓↓
$p^{\uparrow}_{\downarrow\downarrow}$
↓↑
$p^{\uparrow}_{\downarrow\uparrow}$
$p^{\downarrow}_{\downarrow\uparrow}$
$p^{\uparrow}_{\uparrow\downarrow}$
$p^{\downarrow}_{\uparrow\downarrow}$
$p^{\uparrow}_{\uparrow\uparrow}$
↑↑
↑↓
$p^{\downarrow}_{\uparrow\uparrow}$

FIG. 8. ϵ-Machine that generates the spin configurations occurring in the one-dimensional ferromagnetic next-nearest-neighbor Ising spin chain with the Hamiltonian in Eq. (5).

decay rate u, as defined in Eq. (3)? First, note that u is not a configuration's actual energy density E. If we assume that the system is in thermal equilibrium and, thus, exhibits a Boltzmann distribution over configurations, then u and E are related via

$$u = \frac{\log_2(e)}{k_B T}[E - \mathcal{F}(T)],$$

where

$$\mathcal{F}(T) = -k_B T \lim_{n\to\infty} \frac{1}{n} \ln\left(\sum_{\{w\in\mathcal{A}^n\}} e^{-\frac{nE(w)}{k_B T}}\right).$$

This simply tells us that if a stochastic process describes thermalized configurations of a physical system with some given Hamiltonian, then every rare-event bubble in Fig. 4 can be labeled with β, u, or E. Moreover, there is a one-to-one mapping between every such variable pair.

Figure 9 plots $\eta(u)$ versus u—the memory ratio for generating rare configurations with decay rate u. To calculate $\eta(u)$ for a given process $\mathcal{P}$, first we determine the process' classical generator $M(\mathcal{P})$ using the method introduced in Ref. [39]. Second, for every $\beta \in \mathbb{R}/\{0\}$, using the map introduced in Sec. VI, we find the new classical generator $M(\mathcal{P}_\beta)$. Third, using the construction introduced in Sec. III, we find $Q(\mathcal{P}_\beta)$. Fourth, using Theorem 1, we find the corresponding u for the chosen β. Using these results gives $\eta[u(\beta)] = C_\mu(\beta)/C_q(\beta)$. By varying β in the range $\mathbb{R}/\{0\}$ we cover all the energy density us. Practically, to calculate $\eta(u)$ in Fig. (9), we choose 2000 $\beta \in [-10, 7.5]$.

As pointed out earlier, $\beta = 1$ always corresponds to the process itself. Also, one obtains its typical sequences. As one sees in Fig. 9, the memory ratio $\eta(1) < 2$. This simply means that, although there is a quantum advantage generating typical sequences, it is not that notable. However, the figure highlights four other interesting regimes.

First, there is the $\beta \to \infty$ limit corresponding to the rare class with minimum decay rate equal to $u_{\min} = -\log_2(p^{\downarrow}_{\downarrow\downarrow}) = -\log_2(p^{\uparrow}_{\uparrow\uparrow})$. From Eq. (5), it is easy to see that this rare bubble only has two configurations as members: all up-spins or all down-spins. Let us consider a finite but large $\beta \gg 1$ that corresponds to the rare class with a low energy density close to $u_{\min}$. Figure 10 (top right) shows a general ϵ-machine for this process. Low color intensity for both edges and states means that the process rarely visits them during generation. This means, in turn, that a typical realization consists of large blocks of all up-spins and all down-spins. These large blocks are joined by short segments.

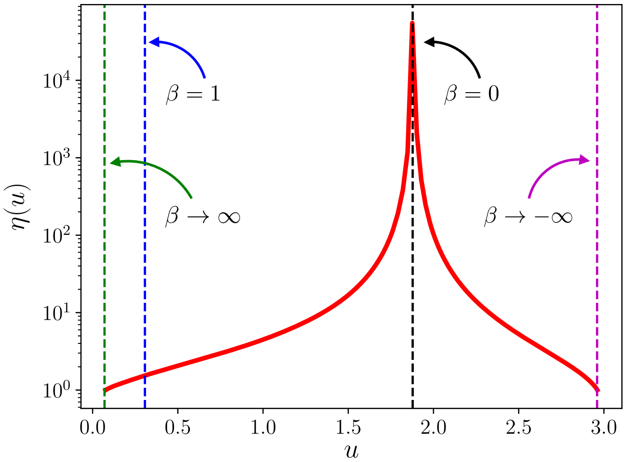

FIG. 9. Classical-to-quantum memory ratio for biased sampling of Ising spin configurations: $\eta(u)$ versus decay rate u for bias sampling of equal-energy density spin configurations. Vertical lines locate β's corresponding to particular us. Note that the memory ratio $\eta(u)$ diverges at $u = u_0 \approx 1.878$, corresponding to $\beta = 0$. The quantum memory advantage is not limited to $\beta = 0$, but occurs in a large neighborhood around it. Quantitatively, for any arbitrary large number r, there exists a number ϵ for which the rare class $\beta_0 = \epsilon$ has the memory ratio $\eta(\beta_0) > r$.

Second, there is the $\beta \to -\infty$ limit that corresponds to the rare class with a maximum decay rate equal to $u_{\max} = -\frac{1}{2}\log_2(p^{\downarrow}_{\downarrow\uparrow} p^{\uparrow}_{\uparrow\downarrow})$. From Eq. (5), it is easy to see that this rare bubble only has one configuration as a member: a periodic repetition of spin down and spin up. Consider a finite $\beta \ll -1$, corresponding to a rare class with a high energy density close to $u_{\max}$. Figure 10 (top left) shows the general ϵ-machine for the associated process. The typical configuration consists of large blocks tiled with spin-up and spin-down pairs that are connected by other short segments.

Third, there is the $\beta \to 0^+$ limit. In this limit, we expect to see completely random spin-up/spin-down configurations. Figure 10 (bottom right) shows the ϵ-machine for this class labeled with nonzero small β. The transition probability for the edges labeled $+$ is $\frac{1}{2} + \epsilon$ and for the edges labeled $-$ is $\frac{1}{2} - \epsilon$, where ϵ is a small positive number. As one can see, even though each transition probability is close to one half, the self-loops are slightly favored.

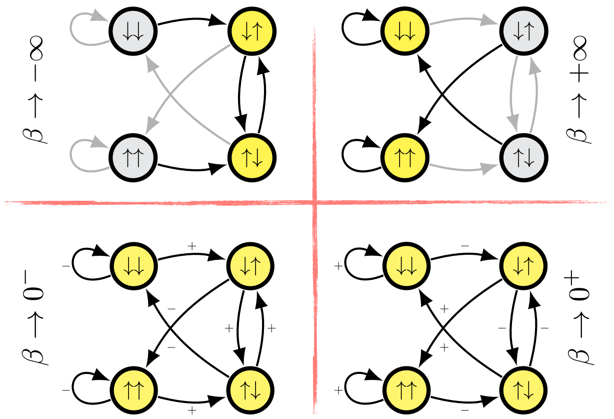

FIG. 10. Classical generators of four important rare classes. Top left: Negative infinite temperature limit. Top right: Positive temperature limit. Bottom left: Negative zero-temperature limit. Bottom right: Positive zero temperature limit. Gray edges and states denote them being rarely visited.

Fourth and finally, there is the $\beta \to 0^-$ limit. The generator here, Fig. 10 (bottom left), is similar to that at positive infinite temperature, except that the edge-sign labels are reversed. This means that the self-loops are slightly less favored.

Appendix D explains the meaning of the β regimes and why each is important.

Remarkably, the memory ratio $\eta(u)$ diverges at $u = u_0 \approx 1.878$, where $u_0 = \lim_{\beta\to 0} u$, that is, at both the positive and negative high-temperature limit. Moreover, the memory ratio $\eta(u)$ diverges as $(u - u_0)^{-2}$ in both limits.

C. Extreme quantum memory advantage for N-nearest-neighbor spin systems

Now, consider the multiprocess setting in which we specify a series of stochastic processes labeled by an integer N that determines the size of each. In this, efficiency is defined by how the memory scales in N for the quantum algorithm compared to the classical.

We explore a general one-dimensional ferromagnetic N-nearest-neighbors Ising spin-$\frac{1}{2}$ chain defined by the Hamiltonian

$$\mathcal{H}_N = -\sum_i \sum_{k=1}^{N} \frac{1}{k^\delta} s_i s_{i+k},$$

in contact with a thermal bath at temperature $k_B T = 1$ and for which there is monopole-dipole coupling ($\delta = 2$).

As in the nearest-neighbor spin system, after thermalizing, the probability over configurations at one instant of time defines a spatially stationary stochastic process. We denote the process generated by this Hamiltonian by $\mathcal{P}(N)$. $\mathcal{P}(N)$ has Markov order $R = N$. More to the point, the ϵ-machine that generates this process has 2^N causal states and those states are in one-to-one correspondence with the set of length-N spin configurations. To determine the ϵ-machine transition probabilities $\{T^{(x)}\}_{x\in\mathcal{A}}$, one can use Ref. [63] and the appendix of Ref. [41].

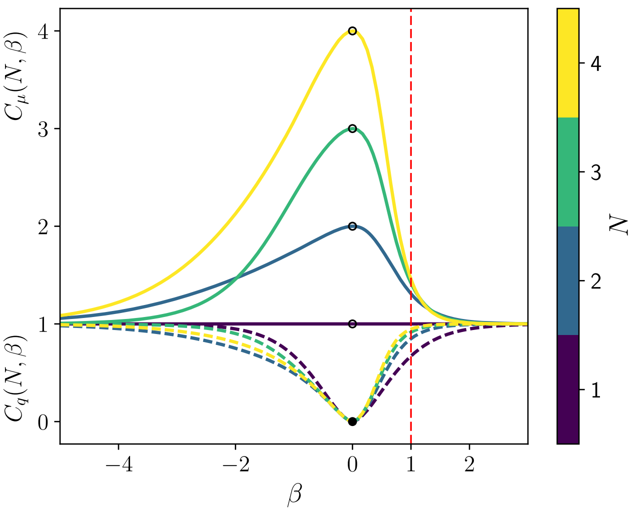

FIG. 11. Classical memory $C_\mu(N,\beta)$ (solid lines) and quantum memory $C_q(N,\beta)$ (dashed lines) required for generating process $\mathcal{P}(N,\beta)$ for interaction ranges $N = 1,\ldots,4$; a range of $\beta \in [-5, 3]$; and $\delta = 2$. At both limits, $\beta \to 0^+$ and $\beta \to 0^-$, $C_\mu(N,\beta)$ scales linearly with N, while $C_q(N,\beta)$ vanishes. For any finite β, for sufficiently large N, $C_\mu(N,\beta)$ is an increasing function of N, while $C_q(N,\beta)$ is bounded above by 1. The vertical dashed red line at $\beta = 1$ shows the memory costs for generating typical sequences.

Let $\mathcal{P}(N,\beta)$ denote the process that typically generates the rare β-class of process $\mathcal{P}(N)$. Now, for an arbitrary fixed β, one can ask how the required classical memory $C_\mu(N,\beta)$ and quantum memory $C_q(N,\beta)$ for generating $\mathcal{P}(N,\beta)$ scale with N.

Figure 11 shows $C_\mu(N,\beta)$ and $C_q(N,\beta)$ versus β for different N's. C_μs are plotted by solid lines and C_q's by dashed lines. To make them distinguishable, curves at different N's are displayed with different colors. Interestingly, in both the $\beta \to 0^+$ and $\beta \to 0^-$ limits, $C_\mu(N,\beta)$ scales linearly with N, while $C_q(N,\beta)$ goes to zero. More importantly, one can also check that for any finite nonzero β and sufficiently large N, $C_\mu(N,\beta)$ is an increasing function of N. Surprisingly, it can be shown that for any nonzero β and any N, $C_q(N,\beta)$ is bounded above by 1. The result is an extreme quantum memory advantage for rare-event sampling of this series of stochastic processes.

VIII. CONCLUSIONS

We introduced a new quantum algorithm for simultaneous sampling rare events in classical stochastic processes. We showed that it confers a significant memory advantage when compared to the best-known classical algorithm.

We explored two settings: single-process and multiprocess sampling. For single processes, an individual stochastic process is given and memory efficiency is defined as the ratio of memory required by the classical algorithm compared to that by the quantum one. For two example systems, we showed that there exist infinite classes of rare events for which the classical-quantum memory ratio for sampling is larger than r, for any large real number r. In the multiprocess setting, a series of stochastic processes, each labeled by an integer size N, is given. There, the memory efficiency is defined by how the required memory scales in N for the classical algorithm compared to the quantum algorithm. In this setting, we demonstrated an extreme quantum memory advantage in which the classical memory grows with N unboundedly, but the quantum memory is bounded.

ACKNOWLEDGMENTS

The authors thank Leonardo Duenas-Osorio for stimulating discussions on risk estimation in networked infrastructure and the referees for valuable suggestions. J. P. C. thanks the Santa Fe Institute for its hospitality during visits as an External Faculty member. This material is based upon work supported by, or in part by, the John Templeton Foundation Grant No. 52095, the Foundational Questions Institute Grant No. FQXi-RFP-1609, and the U.S. Army Research Laboratory and the U.S. Army Research Office under Contracts No. W911NF-13-1-0390 and No. W911NF-13-1-0340.

APPENDIX A: COMPLETENESS CONDITION

To establish Kraus operator completeness, $\sum_x K_x^\dagger K_x = I$, it is enough to show that, for all l and k,

$$\langle \xi_l | \left(\sum_x K_x^\dagger K_x \right) | \xi_k \rangle = \langle \xi_l | \xi_k \rangle .$$

Using the definition of Kraus operators, we have

$$\sum_x K_x^\dagger K_x = \sum_{\substack{i,j\\m,n\\x}} \sqrt{T_{mn}^{(x)}} \sqrt{T_{ij}^{(x)}} |\widetilde{\xi_m}\rangle \langle \xi_n | \xi_j \rangle \langle \widetilde{\xi_i} | .$$

As a result, for all l and k,

$$\langle \xi_l | \left(\sum_x K_x^\dagger K_x \right) | \xi_k \rangle = \sum_{\substack{n,j\\x}} \sqrt{T_{ln}^{(x)}} \sqrt{T_{kj}^{(x)}} \langle \xi_n | \xi_j \rangle = \langle \xi_l | \xi_k \rangle .$$

This completes the proof.

APPENDIX B: STATIONARY DISTRIBUTION

We calculate the stationary distribution over causal states. First, we have

$$\begin{aligned} \sum_x K_x \rho_s K_x^\dagger &= \sum_{\substack{r,i,j\\m,n\\x}} \pi_r \sqrt{T_{mn}^{x}} \sqrt{T_{ij}^{x}} |\xi_j\rangle \langle \widetilde{\xi_i} | \xi_r \rangle \langle \xi_r | \widetilde{\xi_m} \rangle \langle \xi_n | \\ &= \sum_{\substack{n,j,r\\x}} \pi_r \sqrt{T_{rn}^{(x)}} \sqrt{T_{rj}^{(x)}} |\xi_j\rangle \langle \xi_n | . \end{aligned}$$

Since ϵ-machines are unifilar [64], we always have $\sqrt{T_{rn}^{(x)}} \sqrt{T_{rj}^{(x)}} = T_{rn}^{x}$. From this, we find

$$\sum_x K_x \rho_s K_x^\dagger = \sum_{\substack{n,r\\x}} \pi_r T_{rn}^{(x)} |\xi_n\rangle \langle \xi_n | .$$

The stationary distribution over causal states satisfies $\pi T = \pi$ or, equivalently, $\pi_n = \sum_{x,r} \pi_r T_{rn}^{(x)}$. Replacing π_n in the above equation leads to

$$\sum_x K_x \rho_s K_x^\dagger = \sum_n \pi_n |\xi_n\rangle \langle \xi_n | = \rho_s .$$

APPENDIX C: PERTURBED COINS PROCESS QUANTUM MACHINE

First step: (Find the $|\eta_i\rangle$s): Map causal states A and B to two pure quantum states $|\eta_A\rangle$ and $|\eta_B\rangle$:

$$\begin{aligned} |\eta_A\rangle &= \sqrt{p}|0\rangle + \sqrt{1-p}|1\rangle \\ |\eta_B\rangle &= \sqrt{1-q}|0\rangle + \sqrt{q}|1\rangle , \end{aligned}$$

where $|0\rangle$ and $|1\rangle$ form orthogonal bases on Hilbert spaces of size 2.

Second step: (Find $|\xi_i\rangle$s and $|\tilde{\xi}_i\rangle$s): Since the size of alphabet $|\mathcal{A}| = 2$ and Markov order is $R = 1$, then $|\mathcal{A}|^R = 2$. The number of causal states is also $|\boldsymbol{\mathcal{S}}| = 2$. As a result $|\xi_A\rangle = |\eta_A\rangle$, $|\xi_B\rangle = |\eta_B\rangle$,

$$\Xi = [\, |\xi_A\rangle \quad |\xi_B\rangle \,] = \begin{bmatrix} \sqrt{p} & \sqrt{1-q} \\ \sqrt{1-p} & \sqrt{q} \end{bmatrix} ,$$

and

$$\Xi^{-1} = \begin{bmatrix} \langle \widetilde{\xi_A} | \\ \langle \widetilde{\xi_B} | \end{bmatrix} = \frac{\begin{bmatrix} \sqrt{q} & -\sqrt{1-q} \\ -\sqrt{1-p} & \sqrt{p} \end{bmatrix}}{\sqrt{pq} - \sqrt{(1-p)(1-q)}} .$$

Third step: (Find Kraus operators K_is):

$$K_0 = \sqrt{p}|\xi_A\rangle\langle\widetilde{\xi_A}| + \sqrt{1-q}|\xi_A\rangle\langle\widetilde{\xi_B}| = \begin{bmatrix} \sqrt{p} & 0 \\ \sqrt{1-p} & 0 \end{bmatrix}$$

$$K_1 = \sqrt{q}|\xi_B\rangle\langle\widetilde{\xi_B}| + \sqrt{1-p}|\xi_B\rangle\langle\widetilde{\xi_A}| = \begin{bmatrix} 0 & \sqrt{1-q} \\ 0 & \sqrt{q} \end{bmatrix}.$$

We can easily check the completeness condition $K_0^\dagger K_0 + K_1^\dagger K_1 = I$.

Fourth step: (Find stationary distribution ρ_s): For the stationary state distribution over causal states, we have $\pi T = \pi$, where

$$T = \begin{bmatrix} p & 1-p \\ 1-q & q \end{bmatrix}.$$

As a result,

$$\pi = \begin{bmatrix} \pi_A \\ \pi_B \end{bmatrix} = \frac{1}{2-p-q}\begin{bmatrix} 1-q \\ 1-p \end{bmatrix}.$$

Using $\rho_s = \pi_A|\xi_A\rangle\langle\xi_A| + \pi_B|\xi_B\rangle\langle\xi_B|$, we find

$$\rho_s = \frac{1}{2-p-q}\begin{bmatrix} 1-q & \alpha \\ \alpha & 1-p \end{bmatrix},$$

where $\alpha = (1-q)\sqrt{p(1-p)} + (1-p)\sqrt{q(1-q)}$.

APPENDIX D: MEANING OF THE β REGIMES

By Eq. (3), $\Lambda^{\mathcal{P}}_{u,n}$ is the subset of all words with length n, $\Lambda^{\mathcal{P}}_{u,n} \subseteq \mathcal{A}^n$, whose probability decay rate $u = -\frac{1}{n}\log_2 \mathbb{P}(w)$. This partitions $\mathcal{A}^n$ into disjoint subsets $\Lambda^{\mathcal{P}}_{u,n}$ shown by green bubbles in Fig. 12. By definition, all the words in the bubble $\Lambda^{\mathcal{P}}_{u,n}$ have the same probabilities 2^{-nu}, as Fig. 12 shows.

One important question is how the probability of the whole bubble $\mathbb{P}(\Lambda^{\mathcal{P}}_{u,n})$ decays with n. This quantity can be written as a product of two factors: $\mathbb{P}(\Lambda^{\mathcal{P}}_{u,n}) = 2^{-nu}|\Lambda^{\mathcal{P}}_{u,n}|$. The first is the probability of each word in the bubble, and the second is the number of words in it.

Since $|\Lambda^{\mathcal{P}}_{u,n}|$ grows exponentially with n, we define

$$S(u) \equiv \lim_{n\to\infty} \frac{\log_2(|\Lambda^{\mathcal{P}}_{u,n}|)}{n}.$$

Now, for the probability of the bubble with energy density u, one can write

$$\mathbb{P}(\Lambda^{\mathcal{P}}_{u,n}) = 2^{-n(u-S(u))}.$$

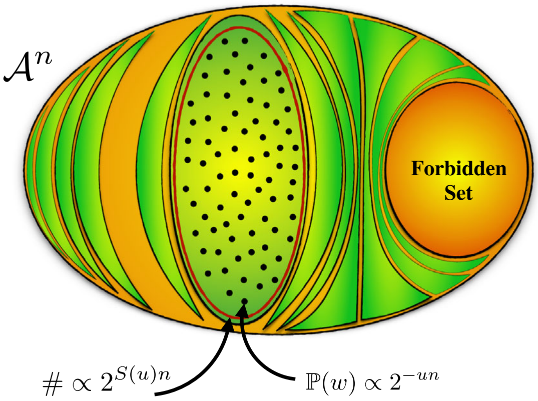

FIG. 12. $\mathcal{A}^n$, the set of words with length n, can be decomposed to different bubbles $\Lambda^{\mathcal{P}}_{u,n}$, each labeled with different u. The number of words in the bubble with the energy density u is $|\Lambda^{\mathcal{P}}_{u,n}| = 2^{nS(u)}$, and the probability of each word in this bubble is 2^{-nu}.

The quantity $I(u) = u - S(u)$ is often called the large deviation rate. $I(u)$ is always positive except for the typical set where it vanishes. This simply tells us that $\mathbb{P}(\Lambda^{\mathcal{P}}_{u,n})$, the probability of every bubble except the typical set, decays exponentially with n.

This provides the background needed to interpret the meaning behind different β regimes.

$\beta \ll -1$ *regime:* This regime includes bubbles with very high energy density u. Recalling that the probability of each word in the bubble with energy density u is 2^{-nu}, we see that this regime includes the bubble with the rarest words. Notice that this does not refer to the probability of an entire bubble but, rather, the probability of each word in the bubble.

$\beta \gg 1$ *regime:* This regime includes bubbles with very low energy density u. Again, since the probability of each word in the bubble is 2^{-nu}, then this regime includes the bubble with the most probable words. To emphasize, this is the probability of each word in the bubble. For example, the bubble that has the most probable word can potentially have one member and still be very rare.

$\beta = 1$ *class:* This points to a particular bubble, the typical set.

$\beta \in (-\epsilon, \epsilon)$ *regime:* This includes bubbles with βs near zero. One can show that $\partial_u S[u(\beta)] = \beta$. As a result, the maximum of $S[u(\beta)]$ happens at $\beta = 0$. Recalling the definition of $S(u)$, this means that the bubble with the maximum number of words is labeled by $\beta = 0$ or, equivalently, $u(\beta = 0)$. In other words, $\beta \in (-\epsilon, \epsilon)$ includes bubbles that have the largest number of words in them or, equivalently, the largest $S(u)$.

[1] A. R. Leach, *Molecular Modelling: Principles and Applications* (Pearson Education, Boston, Massachusetts, 2001).

[2] D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* 2nd ed (Academic Press, New York, 2007).

[3] H. D. Miller, *A Convexity Property in the Theory of Random Variables Defined on a Finite Markov Chain*, Ann. Math. Stat. **32**, 1260 (1961).

[4] K. Young and J. P. Crutchfield, *Fluctuation Spectroscopy*, Chaos Solitons Fractals **4**, 5 (1994).

[5] V. Lecomte, C. Appert-Rolland, and F. van Wijland, *Chaotic Properties of Systems with Markov Dynamics*, Phys. Rev. Lett. **95**, 010601 (2005).

[6] V. Lecomte, C. Appert-Rolland, and F. Van Wijland, *Thermodynamic Formalism for Systems with Markov Dynamics*, J. Stat. Phys. **127**, 51 (2007).

[7] R. Chetrite and H. Touchette, *Nonequilibrium Microcanonical and Canonical Ensembles and Their Equivalence*, Phys. Rev. Lett. **111**, 120601 (2013).

[8] S. R. S. Varadhan, *Large Deviations and Applications* (SIAM, Philadelphia, Pennsylvania, 1984).

[9] J. P. Garrahan, R. L. Jack, V. Lecomte, E. Pitard, K. van Duijvendijk, and F. van Wijland, *First-Order Dynamical Phase Transition in Models of Glasses: An Approach Based on Ensembles of Histories*, J. Phys. A **42**, 075007 (2009).

[10] L. O. Hedges, R. L. Jack, J. P. Garrahan, and D. Chandler, *Dynamic Order-Disorder in Atomistic Models of Structural Glass Formers*, Science **323**, 1309 (2009).

[11] J. Van Campenhout and T. M. Cover, *Maximum Entropy and Conditional Probability*, IEEE Trans. Inf. Theory **27**, 483 (1981).

[12] I. Csiszár, *Sanov Property, Generalized I-Projection and a Conditional Limit Theorem*, Ann. Probab. **12**, 768 (1984).

[13] G. M. Torrie and J. P. Valleau, *Nonphysical Sampling Distributions in Monte Carlo Free-Energy Estimation: Umbrella Sampling*, J. Comput. Phys. **23**, 187 (1977).

[14] S. Kumar, J. M. Rosenberg, D. Bouzida, R. H. Swendsen, and P. A. Kollman, *The Weighted Histogram Analysis Method for Free Energy Calculations on Biomolecules. I. The Method*, J. Comput. Chem. **13**, 1011 (1992).

[15] F. Wang and D. P. Landau, *Efficient, Multiple-Range Random Walk Algorithm to Calculate the Density of States*, Phys. Rev. Lett. **86**, 2050 (2001).

[16] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times* (American Mathematical Society, Providence, Rhode Island, 2009).

[17] J. R. Norris, *Markov Chains* (Cambridge University Press, Cambridge, United Kingdom, 1998), Vol. 2.

[18] D. R. Upper, *Theory and Algorithms for Hidden Markov Models and Generalized Hidden Markov Models*, Ph.D. thesis, University of California, Berkeley, 1997.

[19] L. R. Rabiner and B. H. Juang, *An Introduction to Hidden Markov Models*, IEEE ASSP Mag. **3**, 4 (1986).

[20] L. R. Rabiner, *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*, Proc. IEEE **77**, 257 (1989).

[21] J. P. Crutchfield and D. P. Feldman, *Regularities Unseen, Randomness Observed: Levels of Entropy Convergence*, Chaos **13**, 25 (2003).

[22] J. P. Crutchfield, *Between Order and Chaos*, Nat. Phys. **8**, 17 (2012).

[23] J. P. Crutchfield and K. Young, *Inferring Statistical Complexity*, Phys. Rev. Lett. **63**, 105 (1989).

[24] P. W. Shor, *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*, SIAM Rev. **41**, 303 (1999).

[25] L. K. Grover, *A Fast Quantum Mechanical Algorithm for Database Search*, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, Philadelphia, Pennsylvania, 1996* (ACM, New York, 1996), pp. 212–219.

[26] D. S. Abrams and S. Lloyd, *Quantum Algorithm Providing Exponential Speed Increase for Finding Eigenvalues and Eigenvectors*, Phys. Rev. Lett. **83**, 5162 (1999).

[27] A. W. Harrow, A. Hassidim, and S. Lloyd, *Quantum Algorithm for Linear Systems of Equations*, Phys. Rev. Lett. **103**, 150502 (2009).

[28] M. Gu, K. Wiesner, E. Rieper, and V. Vedral, *Quantum Mechanics Can Reduce the Complexity of Classical Models*, Nat. Commun. **3**, 762 (2012).

[29] J. R. Mahoney, C. Aghamohammadi, and J. P. Crutchfield, *Occam's Quantum Strop: Synchronizing and Compressing Classical Cryptic Processes via a Quantum Channel*, Sci. Rep. **6**, 20495 (2016).

[30] P. M. Riechers, J. R. Mahoney, C. Aghamohammadi, and J. P. Crutchfield, *Minimized State Complexity of Quantum-Encoded Cryptic Processes*, Phys. Rev. A **93**, 052317 (2016).

[31] M. S. Palsson, M. Gu, J. Ho, H. M. Wiseman, and G. J. Pryde, *Experimental Quantum Processing Enhancement in Modelling Stochastic Processes*, Sci. Adv. **3**, e1601302 (2017).

[32] S. Aaronson and A. Arkhipov, *The Computational Complexity of Linear Optics*, in *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing, San Jose, 2011* (ACM, New York, 2011), pp. 333–342.

[33] A. Ambainis, L. J. Schulman, A. Ta-Shma, U. Vazirani, and A. Wigderson, *The Quantum Communication Complexity of Sampling*, SIAM J. Comput. **32**, 1570 (2003).

[34] M. Szegedy, *Quantum Speed-Up of Markov Chain Based Algorithms*, in *Foundations of Computer Science, 2004. Proceedings. 45th Annual IEEE Symposium on* (IEEE, New York, 2004), pp. 32–41.

[35] P. Wocjan and A. Abeyesinghe, *Speedup via Quantum Sampling*, Phys. Rev. A **78**, 042336 (2008).

[36] N. F. Travers, *Exponential Bounds for Convergence of Entropy Rate Approximations in Hidden Markov Models Satisfying a Path-Mergeability Condition*, Stoch. Proc. Appl. **124**, 4149 (2014).

[37] C. Aghamohammadi and J. P. Crutchfield, *Minimum Memory for Generating Rare Events*, Phys. Rev. E **95**, 032101 (2017).

[38] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. (Wiley-Interscience, New York, 2006).

[39] C. R. Shalizi and J. P. Crutchfield, *Computational Mechanics: Pattern and Prediction, Structure and Simplicity*, J. Stat. Phys. **104**, 817 (2001).

[40] G. Han and B. Marcus, *Analyticity of Entropy Rate of Hidden Markov Chains*, IEEE Trans. Inf. Theory **52**, 5251 (2006).

[41] C. Aghamohammdi, J. R. Mahoney, and J. P. Crutchfield, *Extreme Quantum Advantage When Simulating Strongly Coupled Classical Systems*, Sci. Rep. **7**, 6735 (2017).

[42] A. J. P. Garner, Q. Liu, J. Thompson, V. Vedral, and M. Gu, *Provably Unbounded Memory Advantage in Stochastic Simulation Using Quantum Mechanics*, New J. Phys. **19**, 103009 (2017).

[43] T. J. Elliott and M. Gu, *Occam's Vorpal Quantum Razor: Memory Reduction When Simulating Continuous-Time Stochastic Processes with Quantum Devices*, arXiv:1704.04231.

[44] C. Aghamohammadi, J. R. Mahoney, and J. P. Crutchfield, *The Ambiguity of Simplicity in Quantum and Classical Simulation.*, Phys. Lett. A **381**, 1223 (2017).

[45] F. G. Jouneghani, M. Gu, J. Ho, J. Thompson, W. Y. Suen, H. M. Wiseman, and G. J. Pryde, *Observing the Ambiguity of Simplicity via Quantum Simulations of an Ising Spin Chain*, arXiv:1711.03661.

[46] A. J. P. Garner, J. Thompson, V. Vedral, and M. Gu, *Thermodynamics of Complexity and Pattern Manipulation*, Phys. Rev. E **95**, 042140 (2017).

[47] J. Preskill, Lecture Notes for Physics 229: Quantum Information and Computation (California Institute of Technology, Pasadena, California, 1998), Vol. **16**.

[48] F. C. Binder, J. Thompson, and M. Gu, *A Practical, Unitary Simulator for Non-Markovian Complex Processes*, arXiv:1709.02375.

[49] C Perry, R Jain, and J Oppenheim, *Communication Tasks with Infinite Quantum-Classical Separation*, Phys. Rev. Lett. **115**, 030504 (2015).

[50] R. Durrett, *Probability: Theory and Examples* (Cambridge University Press, Cambridge, United Kingdom, 2010).

[51] S. Kullback, *Information Theory and Statistics* (Dover, New York, 1968).

[52] R. W. Yeung, *Information Theory and Network Coding* (Springer, New York, 2008).

[53] C. E. Shannon, *A Mathematical Theory of Communication*, Bell Syst. Tech. J. **27**, 379, 623 (1948).

[54] B. McMillan, *The Basic Theorems of Information Theory*, Ann. Math. Stat. **24**, 196 (1953).

[55] L. Breiman, *The Individual Ergodic Theorem of Information Theory*, Ann. Math. Stat. **28**, 809 (1957).

[56] L. Boltzmann, *Lectures on Gas Theory* (University of California Press, Berkeley, California, 1964).

[57] H. Touchette, *The Large Deviation Approach to Statistical Mechanics*, Phys. Rep. **478**, 1 (2009).

[58] R. L. Jack and P. Sollich, *Large Deviations and Ensembles of Trajectories in Stochastic Models*, Prog. Theor. Phys. Suppl. **184**, 304 (2010).

[59] J. P. Garrahan and I. Lesanovsky, *Thermodynamics of Quantum Jump Trajectories*, Phys. Rev. Lett. **104**, 160601 (2010).

[60] R. Chetrite and H. Touchette, *Nonequilibrium Markov Processes Conditioned on Large Deviations*, Ann. Inst. Henri Poincaré **16**, 2005 (2015).

[61] R. J. Baxter, *Exactly Solved Models in Statistical Mechanics* (Academic Press, New York, New York, 2007).

[62] A. Aghamohammadi, C. Aghamohammadi, and M. Khorrami, *Externally Driven One-Dimensional Ising Model*, J. Stat. Mech. 2012, P02004.

[63] D. P. Feldman and J. P. Crutchfield, Technical Report No. 98-04-026, Santa Fe Institute, 1998, http://csc.ucdavis.edu/~cmg/compmech/pubs/DNCO.htm.

[64] J. R. Mahoney, C. J. Ellison, R. G. James, and J. P. Crutchfield, *How Hidden Are Hidden Processes? A Primer on Crypticity and Entropy Convergence*, Chaos **21**, 037112 (2011).