PNAS

Is stochastic thermodynamics the key to understanding the energy costs of computation?

David H. Wolpert[a,b,c,d,e,1] (ID), Jan Korbel[b,f] (ID), Christopher W. Lynn[g,h,i], Farita Tasnim[j], Joshua A. Grochow[k] (ID), Gülce Kardeş[a,k], James B. Aimone[l], Vijay Balasubramanian[a,m,n] (ID), Eric De Giuli[o] (ID), David Doty[p] (ID), Nahuel Freitas[q], Matteo Marsili[d] (ID), Thomas E. Ouldridge[r,s] (ID), Andréa W. Richa[c] (ID), Paul Riechers[t] (ID), Édgar Roldán[d], Brenda Rubenstein[u], Zoltan Toroczkai[v] (ID), and Joseph Paradiso[w] (ID)

The relationship between the thermodynamic and computational properties of physical systems has been a major theoretical interest since at least the 19th century. It has also become of increasing practical importance over the last half-century as the energetic cost of digital devices has exploded. Importantly, real-world computers obey multiple physical constraints on how they work, which affects their thermodynamic properties. Moreover, many of these constraints apply to both naturally occurring computers, like brains or Eukaryotic cells, and digital systems. Most obviously, all such systems must finish their computation quickly, using as few degrees of freedom as possible. This means that they operate far from thermal equilibrium. Furthermore, many computers, both digital and biological, are modular, hierarchical systems with strong constraints on the connectivity among their subsystems. Yet another example is that to simplify their design, digital computers are required to be periodic processes governed by a global clock. None of these constraints were considered in 20th-century analyses of the thermodynamics of computation. The new field of stochastic thermodynamics provides formal tools for analyzing systems subject to all of these constraints. We argue here that these tools may help us understand at a far deeper level just how the fundamental thermodynamic properties of physical systems are related to the computation they perform.

computation | stochastic thermodynamics | energy costs

Information technology accounted for roughly 6 to 10% of world electricity usage in 2018, resulting in a greater carbon footprint than all civil aviation (1, 2). This demand for computational power is only expected to grow, increasing energy costs to society and the environment. Unsurprisingly, reducing energy requirements has become a central concern for the engineering of computers (3).

How does all of this energy usage depend on the details of how the computation is physically implemented, as well as the computational problem itself? The relationship between the unavoidable energetic costs of carrying out a computation in a physical system and the details of that computation is a deep issue that has been of concern in the physics community for close to two centuries (4, 5). For a long time, it was thought that a lower limit on this cost is given by the original version of Landauer's bound (6). This stated that the "thermodynamic cost" of erasing a bit in a physical system was always $k_B T \ln 2$, where k_B is Boltzmann's constant, and T is the temperature of the system.

The modern, fully formal version of this bound applies to any physical system that evolves for a fixed period while in contact with one or more thermal reservoirs. Like all of statistical physics, this modern version of the bound involves the change in the probability distribution over the states of a physical system during its evolution. Specifically, the bound says that the sum over the reservoirs, of the net energy flow of energy to those reservoirs divided by that reservoir's temperature, is lower-bounded by the drop in Shannon entropy of the system.

Unfortunately, Landauer's bound is almost useless for analyzing real-world computers. The problem is that the bound is only saturated when there are no constraints on the processes that can be used to implement the computation, i.e. when we can consider any process whatsoever to achieve the desired change in Shannon entropy. However, real-world computers are almost all subject to constraints on the processes they can use. Such constraints

Author affiliations: ªSanta Fe Institute, Santa Fe, NM 87501; ᵇComplexity Science Hub Vienna, Vienna 1080, Austria; ᶜSchool of Computing and Augmented Intelligence, Arizona State University, Tempe, AZ 85287; ᵈThe Abdus Salam International Centre for Theoretical Physics, Trieste 34151, Italy; ᵉAlbert Einstein Institute for Advanced Study in the Life Sciences, New York, NY 10467; ᶠInstitute for the Science of Complex Systems, Center for Medical Data Science (CeDAS), Medical University of Vienna, Vienna 1090, Austria; ᵍCenter for the Physics of Biological Function, Princeton University, Princeton, NJ 08544; ʰCenter for the Physics of Biological Function, City University of New York, New York, NY 10017; ⁱDepartment of Physics, Yale University, New Haven, CT 06520; ʲBrilliant.org, New York, NY 10036; ᵏDepartment of Computer Science, University of Colorado Boulder, Boulder, CO 80309; ˡSandia National Laboratories, Albuquerque, NM 87185; ᵐDavid Rittenhouse Laboratory, University of Pennsylvania, Philadelphia, PA 19104; ⁿRudolf Peierls Centre for Theoretical Physics, University of Oxford, OX1 3PU, Oxford, United Kingdom; ᵒDepartment of Physics, Toronto Metropolitan University, M5B 2K3, Toronto, ON, Canada; ᵖDepartment of Computer Science, University of California, 95616, Davis, CA; ᑫDepartment of Physics, University of Buenos Aires, C1053, Buenos Aires, Argentina; ʳDepartment of Bioengineering, Imperial College London, SW7 2AZ, London, United Kingdom; ˢCentre for Synthetic Biology, Imperial College London, SW7 2AZ, London, United Kingdom; ᵗSchool of Physical and Mathematical Sciences, Nanyang Quantum Hub, Nanyang Technological University, Singapore 639798, Singapore; ᵘDepartment of Chemistry, Brown University, Providence, RI 02912; ᵛDepartment of Physics and Astronomy, University of Notre Dame, Notre Dame, IN 46556; and ʷMassachusetts Institute of Technology Media Lab, Massachusetts Institute of Technology, Cambridge, MA 02139

¹To whom correspondence may be addressed. Email: dhw@santafe.edu.

invariably mean that the process must have nonzero "irreversible entropy production (EP)," which is an energetic cost over and above the cost demanded by Landauer's bound.

For example, almost all real-world computations must finish in some given, finite time. The thermodynamic speed limit theorems (SLTs) provide nonzero lower bounds on the EP, which increases as the required time shrinks. Furthermore, almost all digital computers run in a periodic process, where an electronic clock on the chip is used to ensure that the exact same process is used to implement each iteration of any desired computation. However, as described below, any periodic process has a nonzero EP. Finally, consider two separate bits that are erased in parallel. Each such bit eraser, considered by itself, achieves Landauer's bound. However, suppose that the initial states of the bits are statistically coupled. In this case, even though each bit-eraser considered by itself is thermodynamically reversible, the joint system of the two bits is *not* thermodynamically reversible. The map between a computer's design features and performance is shown in Fig. 1.

The general theory underlying these examples is a set of revolutionary recent advances in nonequilibrium statistical physics, often called stochastic thermodynamics (7–10). This exploding field provides precisely the tools required to analyze the thermodynamics of far-from-equilibrium systems—like computers. The implications of stochastic thermodynamics for running computers are just starting to be explored.

These extra unavoidable costs mean Landauer's bound is far below the actual energy costs of computers.* This has been verified in numerous investigations of detailed physical models of computers, like complimentary metal-oxide semiconductor (CMOS)-based electronic circuits (11, 12), as well as biological systems like brains (13). These investigations also reveal deep relationships between the thermodynamics of different kinds of physical systems that we characterize as "performing computation." For example, ribosomes perform simple decoding computations at an energy cost within an order of magnitude of Landauer's bound (14). In contrast, typical artificial computers can only reach within multiple orders of magnitude of Landauer's bound (15).

Are there fundamental reasons for the yawning gap between the energetic efficiency of the computers we build and those found in nature? If so, can they be at least partially circumvented? Why do even biological computers require so much more energy than the minimum given by Landauer's bound, in light of the fact that energy use is one of the most important fitness costs in biological evolution?

We argue below that the time is ripe to use stochastic thermodynamics to investigate the relationship between i) the energy usage of physical computational systems, ii) their other performance metrics [e.g., their "space" and "time" complexity, in the sense of those terms investigated in computer science (CS) theory], and iii) constraints on the allowed physical processes in those computers. Such investigations can potentially generate breakthroughs in the design of artificial computers that expend less energy. They may also reveal insights into the relationship between artificial and biological computers.

For experts in stochastic thermodynamics, these investigations provide new kinds of dynamical systems to analyze (namely, those that implement computational machines). For CS theorists, they provide new kinds of resource costs to analyze (namely, those grounded in statistical physics). For scientists as a whole community, they may provide important new insights into physical systems of many types.

1. Background

The term "computation" is semiformally used in many fields (16–19). It is also sometimes called "information processing" in this literature. The common thread uniting the various approaches is that a physical system is "computing" if its dynamics modifies the distribution over the variables of the system, with some subset of those variables identified as "information-bearing degrees of freedom (IBDF)." [Sometimes certain subsets of these IBDF are referred to as input, output, or temporary data variables (6, 20).]

Unfortunately, neither the term "information processing" nor the term "computation" is ever defined in this literature in a way that is both formal and broadly applicable. More precisely, there is no consensus on how to define precisely what computation an arbitrary given dynamic system does.† At present, the only universal agreement of how to define "computation" is that whatever else computation is, it includes the digital systems analyzed in CS theory as special cases. In addition, these are the computational systems that have received the most attention in the stochastic
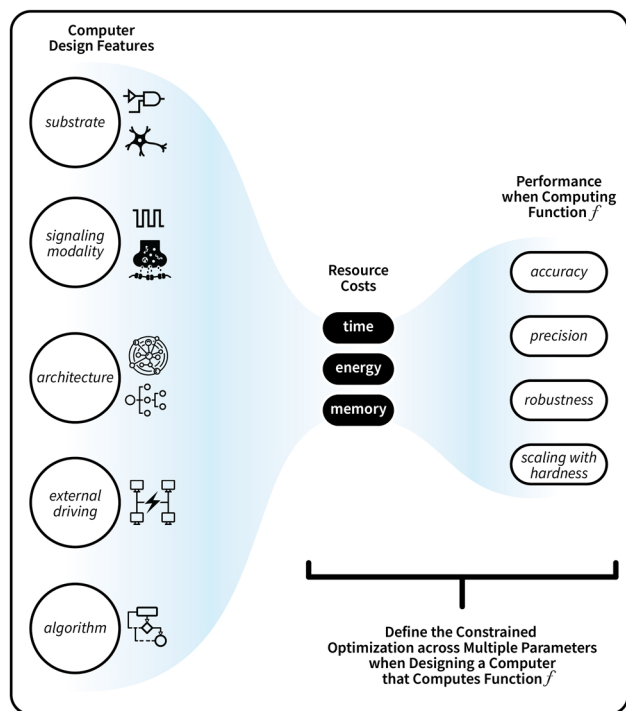
Fig. 1. The mapping between the design features of a computer and its performance when computing a function is mediated by its resource costs.

---

*Indeed if one considers a physical system that performs the computation in a nonequilibrium steady state, typically $\Delta S \approx 0$, in which case Landauer's bound is zero.

†See ref. 21 and references therein for previous literature on this problem. A soon-to-be-published focus issue on this problem is in ref. 22.

thermodynamics community. In light of this, we focus on the digital systems considered in CS theory in this perspective.[‡]

To begin, in the next subsection, we present some relevant background on CS theory. After that, we present some background on stochastic thermodynamics.

1.1. Theoretical Computer Science. Much of theoretical CS is concerned with the minimal resources needed to solve computational problems with some abstractly defined computational machine (23–25). Each such computational machine is a generic model of any entity that computes (transforms an input into an output). There is a set of one or more natural "resource costs" for all commonly studied computational machines. One of the core concerns of CS theory is how the minimal resource costs required to solve a given computational problem on a given computational machine depend on the size of the instance of the problem. In the rest of this subsection, we elaborate on this high-level description of CS theory, underscoring why it is of central importance in designing real computers.

An important example of a computational machine is the Turing machine (TM) (26). This is an abstract, nonphysical device that scans back and forth over one or more "tapes" containing infinite sequences of bits. The input to the TM (e.g., a list of integers to be sorted) specifies the initial string on a tape. The output (e.g., the sorted version of that list of integers) is given by the bit string on a tape of the TM when the TM enters a special "halt state." TMs have a natural notion of a resource cost involving "time" (the number of operations the TM runs before completing a computation) and a resource cost involving "space" (the number of tape cells used by the TM to complete the computation).

Another frequently studied computational machine is a Boolean circuit (24), which is a directed acyclic graph (DAG) where each node implements a Boolean operation (a "gate"). The full graph executes a Boolean function by mapping the input bit string (written onto the bits at the roots of the DAG) through the intermediate gates to an output bit string (sequence of bits produced at the leaves of the DAG). Natural resource costs for Boolean circuits include the total number of gates and the maximal number of directed links from any input node to any output node. These roughly correspond to the time cost and space cost, respectively.

Another set of computational models is designed for investigating distributed sets of interacting computers, e.g., multiple (perhaps very many) computers connected in a network. Associated costs include the total number of asynchronous messages sent among the computers and the number of "rounds" of communication required by the system.

In CS theory, a "computational problem" that a computational machine may be tasked with is defined as a set of instances of that problem that must be solved. For example, one of the most important computational problems is integer sorting (24). Each instance of this problem is specified by a list of integers, and the associated solution is that set of integers sorted into ascending order.

A well-studied issue is how the worst-case number of iterations required by a TM to sort a list of n integers scales with n. In particular, the class P is the set of all computational problems that can be solved using a TM that is guaranteed to halt on an instance of size n "in polynomial time," i.e., within a number of iterations that is a polynomial function of n. The class NP is instead the set of computational problems such that a solution to the problem can be verified by a TM in polynomial time. One of the deepest open questions in mathematics and theoretical CS is whether P = NP.

The main reason for analyzing any of the resource costs formalized in CS theory is that those costs approximate important operating costs that arise when solving computational problems on real physical systems like digital computers. For example, consider a problem that requires a TM to undergo a number of steps, which is roughly exponential as a function of the input size. This problem will also require exponentially many operations as a function of the input size to run on a real, digital computer.[§]

Most previous investigations in CS theory have ignored a primary resource cost of any real computer—the energy it uses to run. One notable exception is an attempt to formalize the energy complexity of algorithms (27). Unfortunately, the study uses Landauer's bound to guide their investigations, which leads to the misunderstanding of the thermodynamic costs of far-off equilibrium systems that we discussed in the Introduction. In hindsight, this is not surprising. Only recently have physicists had the framework (stochastic thermodynamics) to calculate energy usage of out-of-equilibrium systems, and so provide computer scientists with the theoretical tools to explore how energy scales with the size of a problem.

Note that computational complexity theory analyzes how the resource costs of computational problems scale with the size of the (instance of the) problem (24). Typically, these analyses concern worst-case resource costs, mostly due to mathematical tractability. However, other types of costs, including "average-case complexity", "derandomization" or "hardness amplification" are also studied (24). Most importantly, though, many real computation tasks do not typically operate on worst-case inputs but rather on a set of many different inputs with different occurring frequencies. By calculating the average quantities over the set of all possible inputs, we can more accurately estimate the performance of an algorithm when repeatedly used to perform a computation on many possible inputs.

Accordingly, all information theory considers systems with stochastic inputs (28). Similarly, statistical physics involves distributions, including stochastic thermodynamics. This is illustrated in the following subsection.

1.2. Statistical Physics. Throughout the 20th century, the thermodynamics of computation focused on questions of whether an intelligent being could violate the second law of thermodynamics through measurement and feedback (4, 5) and erasure (6, 20) of single bits. Although modern approaches have resolved the paradoxes brought up by these systems (29–31), they retain wide resonance throughout the

---

[‡]An important alternative interpretation of the term "computation" arises in signal processing, which we can view as a type of analog computation. Very little is known about the stochastic thermodynamics of such systems at present, though, so we do not focus on them here.

[§]It is well known in computational complexity theory that a TM with RAM can obtain at most polynomial savings in time cost compared to a regular TM (23, 24).

physics community and have remained a focus of investigation (32, 33), mostly due to mathematical tractability.

Statistical physics, which analyzes the behavior resulting from large collections of components, offers a potential pathway to widen the prior focus to more complex systems. Most of the research in statistical physics through the end of the 20th century considered systems modeled to be at (a perhaps local) thermodynamic equilibrium. These systems are either static, undergoing quasi-static evolution, or undergoing some first-order perturbation around such evolution, as, for example, they are in a local equilibrium that evolves quasistatically (i.e., infinitely slowly). Thus, they cannot complete a computation in finite time.

Accordingly, analyzing real computers requires extending equilibrium statistical physics to concern systems with many nonhomogeneous degrees of freedom evolving quickly while far from thermodynamic equilibrium. Over the past two decades, stochastic thermodynamics, an important development in the study of nonequilibrium systems, has provided a framework for addressing this need in both theoretical computer science and the physics of computation (7, 8).

Stochastic thermodynamics is the extension of thermodynamics which allows us to analyze heat, work, and entropy at the level of individual trajectories generated in a physical system that is evolving arbitrarily quickly while arbitrarily far from thermal equilibrium (7). For a brief introduction to stochastic thermodynamics, see ref. 8 and see refs. 9 and 10 for textbook-length introductions. For a physical system that is evolving while coupled to N thermal reservoirs at temperatures T_i, the entropy production can be expressed as

$$\sigma = \Delta S + \sum_{i=1}^{N} \frac{Q_i}{k_B T_i}. \qquad [1]$$

In this formula, ΔS is the change in entropy between the probability distributions of the input and output states and Q_i is the *heat flow* out of the system to the heat reservoir i. The second law of thermodynamics states that $\sigma \geq 0$, no matter what physical process transpires.

Crucially for the analysis of computers, all of this holds whether we consider an entire computer (e.g., a modern digital computer) or subsystems of that computer (e.g., individual gates on a circuit, or individual circuits in a digital computer). It also holds no matter how long the time interval we are considering, e.g., a single cycle of a clock in a synchronous computer, or the time for the computer to complete an entire run of calculating the output given by running an algorithm on an input.

To illustrate Eq. **1**, consider a physical system performing a computation task (e.g., sorting a list of inputs). Suppose we define a probability distribution over possible inputs and a particular algorithm for the task. For sorting, this would be the joint distribution over input lists. With these specifications, the joint distribution of input, intermediate, and output variables is fixed, and so is the entropy change ΔS. Applying the nonnegativity of EP to Eq. **1** gives the "generalized Landauer bound," which states that total entropy flow cannot be less than the entropy change $\sum_i \frac{Q_i}{k_B T_i} \geq -\Delta S$

Note, that the term ΔS in the generalized Landauer bound can be quite small. Indeed, in many models of biological computational systems like those that occur inside cells, the system is in a steady state—so there is no change in entropy in time. Similarly, consider a clocked, Boolean circuit of depth L that is run repeatedly, with IID inputs. In both cases, $\Delta S = 0$.

However, even when the change in entropy is strictly positive, the generalized Landauer bound is a tiny fraction of the heat generated by running the process in real-world computers. This is due to the fact that the factors of $k_B T_i$ in Eq. **1** are so small compared to the heat flow generated in real-world computers. Moreover, the difference between the heat flow and the change in entropy of a computer depends on the details of the physical system implementing that computer. [For example, they depend on whether a given sorting algorithm is implemented on a system that uses CMOS technology (34) or something else.] Since Eq. **1** is an exact equality, the total EP will vary with the details of the physical process.

Consequently, relying solely on the Landauer bound misses the effect of the large, EP in real computations, which often dominates the total thermodynamic cost. Even biological computers that are far more energetically efficient than our current artificial ones (14, 35) only come within an order of magnitude or two of the generalized Landauer bound. So, Eq. **1** tells us that almost all the dissipated heat (energy cost) of actual computers comes from the EP.

As technology continues to improve, the minimum achievable EP—subject to constraints on the materials involved and other restrictions on the physical process—may rapidly become the primary limitation on the energy costs of real computers. However, it is now known that essentially any constraint results in strictly positive EP (36, 37). In the rest of this section, we discuss some important special cases.

As a first example, one possible source of EP in computation is the mismatch cost (38, 39). Write the initial distribution over states of a system that minimizes EP as q_0.¶ We denote by $\sigma(q_0)$ the EP corresponding to a computational process starting with the distribution q_0. The EP $\sigma(p_0)$ associated with the same process starting at another distribution p_0 can be expressed as (38)

$$\sigma(p_0) = \sigma(q_0) + D(p_0||q_0) - D(p_f||q_f), \qquad [2]$$

where p_f and q_f are the final distributions in the process started at p_0 and q_0, respectively, and $D(p||q) = \sum_x p(x) \ln \frac{p(x)}{q(x)}$ denotes the Kullback–Leibler (KL) divergence. The difference $D(p_0||q_0) - D(p_f||q_f)$ is called the mismatch cost. Importantly, the formula for mismatch cost is very general, applicable to classical systems, quantum systems, and even systems undergoing non-Markovian dynamics.

Note that the difference between the initial and final KL divergences is always nonnegative by the data processing inequality for KL divergence. So even if the process is thermodynamically reversible for some q_0, the system generates a positive EP when initialized at any other initial distribution. We cannot optimize the system's EP for all possible p_0. Therefore, if $p_0 \neq q_0$, EP must be positive.

¶Often, a machine's intermediate variables are initialized (e.g., to 0) before processing input, so the "initial distribution" refers to the probability distribution of input states. For a TM about to compute, the head usually starts in an initialized state, with input tapes set by an IID-sampled probability distribution over possible inputs.

Mismatch cost is not the only source of EP. EP is also unavoidable when there are constraints on the overall dynamics. To illustrate these, in the rest of this paper we will restrict attention to the case where the state space of the system is digitized into a countable number of values, e.g., due to coarse-graining or due to quantum mechanical effects.[#] Given this restriction, then as described above, under certain technical conditions, if a computation has to be performed within a certain time interval τ [here, τ describes physical (clock) time], the EP is lower bounded (41) by the SLT:

$$\sigma(\tau) \geq \frac{\left(\sum_x |p_0(x) - p_\tau(x)|\right)^2}{2A_{\text{tot}}(\tau)}. \quad [3]$$

The numerator is the square of the distance between the initial (p_0) and final (p_τ) distributions. The term $A_{\text{tot}}(\tau)$ in the denominator is the total activity, which is the average number of physical state transitions[||] that occur during the computational process. Thus, the SLT states that the minimum EP increases as the process speeds up.

Another contribution to the EP arises if we need statistical precision in the dynamics of our computer. Fix some observable, real-valued function of state transitions $x' \rightarrow x$ that is antisymmetric under the interchange of the two states. The current J associated with such an "increment function" for a full trajectory is the value of the observable summed over all state transitions in that trajectory. Examples of currents include the total transport of a molecular motor and the total heat flow down a thermal gradient. As another example, we can define the current through a transistor as the difference between the input state and the output state. In general, the more statistically precise the value of the current, the more heat has to be dissipated. Intuitively, when the precision of digital computation is higher, the switching transistors require higher steepness of flanks and more precise rectangular voltage curves, which require increased energy consumption.

Write $\langle J(t) \rangle$ for the expected current over all trajectories in the time period $[0, t]$, and write $\text{Var}(J(t))$ for the variance of J. The ratio of those two quantities is called the (statistical) "precision" of the process. A quickly growing set of thermodynamic uncertainty relations (TUR) (42–44) is being derived that lower bound the statistical precision in terms of the EP generated by the underlying system. The historically first TUR applies in the special case that the system is in a nonequilibrium steady state (NESS):

$$\sigma(\tau) \geq \frac{2\langle J(\tau) \rangle^2}{\text{Var}(J(\tau))}. \quad [4]$$

This TUR says that the minimum EP in a computational process increases as the precision of any current of the process increases, assuming the system is in a NESS.

Many of these theoretical predictions of stochastic thermodynamics have been experimentally validated; see,

e.g., refs. 45–47. In addition, important generalizations of these results have been recently found, including extended versions of the TURs (48–50), and stronger SLTs (51–54). Other results of stochastic thermodynamics promising for the thermodynamics of computers include the fluctuation theorems (55–58), the kinetic uncertainty relations (59), and the thermodynamic correlation inequality (60).

2. Integrating Computer Science and Statistical Physics

2.1. Previous Work. Before discussing how to integrate CS theory and statistical physics, we comment on earlier work. As described in the previous section, real-world computers operate far from equilibrium and cannot be analyzed with equilibrium statistical physics. Before the advent of stochastic thermodynamics, physicists were forced to try to do precisely this since the primary tool they had was the equilibrium statistical physics. This resulted in some confusion in the literature.

In particular, there was some confusion in earlier literature about the relation between logical (ir)reversibility and thermodynamic (ir)reversibility. The difference is that logical reversibility is about the dynamics over a state space (specifically, it must be injective). In contrast, thermodynamic reversibility (i.e., zero EP) is about the associated dynamics of distributions over that state space. These two types of (ir)reversibility are, in fact, completely independent properties of a physical process (29, 61).

A related confusion arose regarding the consideration of computations performed in a logically reversible manner. In that case, it is at least possible that a single run of the computation can be performed without any thermodynamic cost (e.g., refs. 27 and 62). However, the complete cycle of computation, from generation of the input to computation of the output back to generation of a new input, overwriting the original input, results in unavoidable thermodynamic costs. (See section XI in ref. 61.)

While real-world computers cannot be directly analyzed as systems described by equilibrium statistical physics, there has been some important work applying mathematical techniques of equilibrium statistical physics to investigate CS theory problems without considering the underlying dynamics of a computer designed to solve those problems. For example, a recent study (63) presents a random language model, establishing the statistical physics of generative grammars. Furthermore, statistical physics increased attention to inference methods (see, e.g., ref. 64 for a recent review) and deep learning (65). Statistical physics models, such as the Ising model or the hard-core lattice gas model, have also permeated the design of Markov Chain Monte Carlo algorithms in CS (66), and more recently have been applied to swarm robotics systems (67).

Additionally, results from the statistical physics of disordered systems can help us understand how the difficulty of solving a randomly generated computational problem depends on the distribution generating those problems (68). Consider the example of a k-SAT (Boolean satisfiability) problem, the example of a NP-complete problem. (69). The goal of this problem is to determine whether there is an assignment to a set of K Boolean variables that can

---

[#]Extensions of all these results hold when considering the stochastic thermodynamics of systems with uncountably infinite state spaces, e.g., those evolving under an overdamped Langevin equation (40). However, for simplicity, we do not explicitly consider such physical systems in this paper.

[||]Note that the physical state transitions are not necessarily related to the number of transitions (or operations) in the CS sense. Even in the case of a single-bit erasure, when we transform the initial uniform distribution to the distribution where $p(0) = 1$, the system encoding the bit can randomly jump back and forth between the states 0 and 1. The total activity describes the average number of these random jumps per unit of time.

simultaneously satisfy N Boolean clauses (logical OR of K variables or their negations). Random instances of k-SAT are typically easy for a broad range of values of $\alpha = N/K$. However, for a fixed k, existing algorithms for solving k-SAT display a discontinuous phase transition in their efficiency as α increases (25).

Note, though, that these are all applications of statistical physics techniques to computer science. They do not address the issue of the thermodynamic costs associated with physically implementing that computation. In the rest of this section, we review some preliminary work on precisely this issue, and discuss how the unavoidable constraints faced by real computers contribute to their energy costs.

2.2. Stochastic Thermodynamics of Detailed Models of Artificial Computers. Recently, researchers have started to use stochastic thermodynamics to study concrete models of circuits. These studies analyze the behavior of small, linear circuits whose operational voltages are low enough that thermal fluctuations affect their behavior. Early theoretical endeavors used stochastic thermodynamics to characterize the fluctuations in the injected and dissipated energies of resistors (70), nanoscopic circuits (71), and generic electrical conductors (72). Other recent work has analyzed more complex, time-dependent RLC circuits (73). There has also been experimental work on information engines (74) constructed with single-electron circuits (quantum dots, tunnel junctions, etc.) (75, 76) as well as physical implementations of Maxwell's demon (77) and the Szilard engine (78).

In the past few years, in light of rapidly shrinking transistor size, researchers have also started to use stochastic thermodynamics to study the effect of noise in nonlinear electronic circuits, such as CMOS technology (11, 12), in which each gate operates in its subthreshold regime. A recent article (79) goes a step further and studies how different thermodynamic uncertainty relations constrain the efficiency of electronic devices as: an inverter, a memory, and an oscillator. Stochastic thermodynamics has proven useful enough in analyzing electronic circuits that it is starting to be used in the engineering community to model and optimize the operation of low-power electronic devices (80).

As electronic advances near the thermal fluctuation limit, interest in stochastic computing has resurged. Current computers, being deterministic, incur energy costs to prevent thermal fluctuations from affecting logical states and computations. However, many computational tasks actually use stochasticity. In those cases, deterministic computers rely on complicated algorithms to produce pseudorandom numbers. This situation is somewhat paradoxical, wasting energy unnecessarily avoiding stochasticity in the first place only to produce it artificially later. A more efficient alternative, known as stochastic computing, controls intrinsic thermal fluctuations instead of avoiding them to solve problems requiring stochasticity. Such circuits employ probabilistic bits (p-bits), which output 0 with a probability p and output 1 with a probability $1 - p$.

Some novel work has demonstrated the use of magnetic tunnel junctions to create probabilistic bits (81–83). Interestingly, (11) presented the design of a p-bit that uses only conventional CMOS gates operating in the subthreshold regime. Such a CMOS p-bit has not yet been built, but this study shows how stochastic thermodynamics can be used to analyze new computing paradigms that exploit, rather than fight, thermal noise.

Apart from aplication in stochastic algorithms (e.g., Monte Carlo methods), the main advantage of stochastic computing is that, by incorporating tunable p-bits into conventional logic, many tasks can be efficiently formulated as stochastic optimization problems that can be solved by Gibbs sampling (85). This might allow us to solve a broad set of optimization problems in a way that computes both solutions and their associated uncertainties together (82, 84).

2.3. Stochastic Thermodynamics of Abstract Artificial Computers. The optimization of key performance metrics also increases energy dissipation. For example, minimizing computational noise (i.e., having high accuracy in the map from a computation's inputs to its outputs) is a central concern for all computational systems. However, in general, reducing such noise also increases energy dissipation.

As reviewed in Section 1.1, the study of computational complexity aims to analyze computing costs with various abstract computational machines. The simplest nontrivial example is the deterministic finite automaton (DFA). DFAs process input strings sequentially, updating a computational state within a finite state space according to the previous state and the next input symbol.

Since the computational update of a DFA is the same at each iteration, it is natural to assume that any physical process implementing a DFA will also be the same at each iteration. (In fact, there is always such periodicity of the underlying physical process in any clocked, synchronous implementation of a computational machine—precisely the type of computer always used in modern digital systems.) In ref. 86, a strictly positive lower bound was derived on the total mismatch cost of running a DFA this way. The reason for this bound is that the distribution of the system at the start of each iteration, p_0 from Eq. **2**, evolves with the computation, whereas q_0 must remain constant; hence, it is not identical. This phenomenon illustrates a general law: Whenever any physical system repeats the exact same underlying physical process on a system that is not at (periodic) stationary state, each repetition of the process will undoubtedly generate EP. This is true no matter what the details of the physical process are (86, 87). This law applies to all modern digital computers operating synchronously and governed by an electronic clock.

Consider, for example, a DFA designed to recognize input strings consisting of the letters a and b, with no more than two b's in a row. This case is investigated in ref. 86, where it was found that the resultant per-iteration mismatch cost has highly nontrivial iteration dependence—dependence that has yet to be explained. Among their other findings, the paper also showed how to split the regular languages into two separate classes, defined by (lower bounds on) the "mismatch cost complexity" of those languages.

Another recent study investigated properties of the total EP (not just mismatch cost) generated by DFAs using the inclusive Hamiltonian approach of stochastic thermodynamics (88). The inclusive Hamiltonian approach allows the system of interest to operate deterministically, as is the

case for many computational complexity theory models. In applying this approach, the authors showed that the minimal DFA for a given language is also the DFA that incurs the minimum EP of any DFA implementing that language.

Aside from Boolean circuits, almost all of CS theory concerns computers that run for a number of iterations that vary depending on the input to that computer. Unfortunately, stochastic thermodynamics typically concerns physical processes that finish at fixed times; it does not apply when the "stopping time" of the process varies.

However, recent research has extended stochastic thermodynamics to apply to processes with random stopping times (89–92). Preliminary work is investigating the application of these extensions of stochastic thermodynamics to computational systems (87). As an example, we now know that the irreversible entropy produced when a DFA processes a randomly generated bit string obeys a modified "fluctuation theorem" (8, 57, 58),

$$\int ds\, P(s) e^{-\sigma(\boldsymbol{x}(s)) - \delta(\boldsymbol{x}(s))} = 1. \qquad [5]$$

In this expression, $P(s)$ is the distribution over input strings to the DFA, and $\boldsymbol{x}(s)$ is the resultant sequence of states of the DFA up to the time that it halts. The $\delta(\boldsymbol{x})$ term captures the effects of the constraints on the physical process that implements the computation, e.g., that the process be periodic (as it is in digital computers). This explicitly demonstrates how considering such constraints can lead to strengthened versions of the second law—and that these strengthened versions are particularly relevant for investigating computational systems.

In general, different algorithms can implement the same computation, whether the algorithm is represented by a pseudocode, as a program written in assembly code, or as a computational machine from the Chomsky hierarchy [e.g., a finite-state automaton or push-down automaton (23)].

Moreover, the EP generated by physically implementing an algorithm depends more than just on the details of that algorithm. It also depends on the statistical coupling of the states of the other logical variables at the time of the state transition. Consequently, the constraints embodied in the specific algorithm used to implement a given computation contribute to the overall energy costs of implementing the computation with that algorithm. These costs are higher energy costs that arise due to the constraints on the precise physical system that implements that algorithm—see refs. 61, 86, 88, and 93–95 for some preliminary work concerning the stochastic thermodynamics of such algorithms. Also, recent work has derived the minimal EP generated by any communication systems—one of the major sources of heat in modern, real-world computers that have multiple components communicating with one another (96).

2.4. Stochastic Thermodynamics of Biological Computers. Just as artificial systems consume energy to perform computations and process information, so do living systems (97). Perhaps the first example of the connections between biological information processing, energy consumption, and irreversible dynamics was kinetic proofreading (98, 99). In protein synthesis, the genetic code is read with an error rate on the order of 10^4, an accuracy far higher than

can be achieved through any one-step reversible process. However, by inputting energy to generate irreversible steps, Hopfield showed that kinetic proofreading can achieve the accuracy observed in nature (98), a theoretical prediction that has been verified experimentally in multiple settings (100–103).

Stochastic thermodynamics provides a consistent mathematical framework for investigating the energetic costs of biological computations. At the foundation of all biological processes are complex networks of chemical reactions. Using stochastic thermodynamics, one can derive the energy expended and entropy produced by these chemical reaction networks (104, 105).

At a larger scale, cells must perform computations to respond to environmental cues. A simple example, first considered by Berg and Purcell, is the ability of cells to determine the concentration of chemicals in the surrounding (106). It is now understood that learning about the environment requires breaking detailed balance and consuming energy, with greater learning requiring greater energy consumption (107). After the information is gathered, it must be transferred between spatially separated components, which can consume a substantial fraction of the cellular energy budget (108). The problems of cellular sensing and information transfer have received significant focus in the statistical mechanics of living systems, revealing the thermodynamic costs of simple biological computations (97, 109–111).

At an even larger scale, groups of cells (particularly in the brain) must communicate to sense, represent, and process information (112). To execute these functions, the human brain consumes 20% of our metabolic output despite only accounting for 2% of the body's mass (13, 16, 113, 114). Understanding how groups of neurons represent and transmit information while minimizing energy consumption remains a foundational question in neuroscience (115). Recent advances in stochastic thermodynamics have revealed that the irreversibility of neurons in the retina depends critically on the visual stimulus (116). At the whole-brain level in humans, recent studies have suggested that irreversibility and broken detailed balance may reflect increases in cognitive processing and consciousness (117, 118). Yet despite this progress, research at the intersection of stochastic thermodynamics and biology has only begun to scratch the surface of an ultimate understanding of the energetic and thermodynamic costs of computations in living systems.

3. Open Research Topics

As discussed in Section 1.1, to date, CS theory has considered the scaling properties of resource costs other than energy in various kinds of computational systems. Stochastic thermodynamics opens the possibility of integrating the energetic resource costs of computer systems in these analyses. A fascinating—and potentially practically very important—future research program is to investigate the scaling properties of the trade-off between the energy cost of computation and the kinds of computational resource cost traditionally investigated in CS theory.

Concretely, at present, we have a very limited understanding of why thermodynamic costs in both natural and

artificial computers are many orders of magnitude above the minimum possible. The reason for this must lie in the physical constraints that apply to how both artificial and natural computers can operate. To start to investigate those constraints, we need to consider the minimal thermodynamic costs of broad kinds of physical computers rather than the very abstract "computational machines" considered in CS theory.

For example, computational systems can be modeled as networks of distributed computational units that communicate with each other in limited ways. However, we do not know how the properties of a network affect its energy cost when used to implement a computation. To help guide our investigation, we can consider the properties that have been considered before in the literature. In particular, computer scientists (119, 120), neuroscientists (121), network scientists (122, 123), political scientists (124), roboticists (125), and biologists (126, 127) alike have repeatedly identified two key features of network topology to the versatility, robustness, and efficiency of resource utilization in large, distributed computational systems: modularity and hierarchy.

Modularity refers to the property that a network can be divided into a set of discontinuous modules, such that each module's intraconnectivity (number of edges among its own nodes) is much higher than would be expected in a random graph with the same number of edges. Conversely, its interconnectivity (number of edges between its nodes and the nodes of other modules) is much lower (122).

Hierarchy refers to the property that subsystems can be grouped into partially ordered "levels," as in a tree-like network. Several definitions of the hierarchy have been proposed in the literature, e.g., refs. 120 and 128. Both modularity and hierarchy are posited to reduce the cost of establishing long-range correlations in systems of increasing size and complexity. It is known that modularity in a system's architecture increases EP (95, 129), even though it is often useful for robustness (125, 126). However, very little is known beyond this about how the detailed architectural constraints—how each component's state is allowed to affect the dynamics of other components— give rise to EP. The very structure of hierarchical, modular systems leads to additional energy dissipation, e.g., due to mismatch cost (86, 87). Currently, nothing is known about how the benefits of such networks connecting computer components balance with their thermodynamic costs.

Another important open issue arises from the fact that in both real-world artificial computers and real-world biological computers, communication costs—*transferring* information, from point to point, rather than *transforming* information. Formally speaking, communication is a very specific computational process, typically involving encoders and decoders on both sides of a (noisy) communication channel. Focusing on just the channel. that is a system that consists of an "input" subsystem and an "output" subsystem, and the computation is copying the state of the input subsystem to that of the output subsystem. Despite its major importance in real-world computers, applying stochastic thermodynamics to analyze the EP of communication systems—and potentially help design such systems—is still in its infancy (96, 130).

Wireless sensor networks (WSNs) are distributed computing systems that often have lightweight nodes made of single-chip microcomputers. WSNs typically manage many diverse sensors and communicate with one another via a wireless link to collectively detect, analyze, summarize, or react to phenomena that their sensors encounter. As they are usually battery-powered or run off of limited scavenged energy and share common RF channels, much of the effort in WSN systems involves reducing consumed energy. So far, these efforts have involved minimizing communications and processor load, dynamic information routing, and deciding what information to transmit from a given node (131). Stochastic thermodynamics can help reduce energy costs by modifying macroscopic parameters beyond the energy flowing through single gates.

Stochastic thermodynamics also has the potential to analyze certain nonconventional forms of computation. For example, one potentially promising application of stochastic thermodynamics to computation is in the growing sphere of molecular computation, including DNA (17, 132) and small molecule computation (133–135). In these forms of computation, molecular concentrations are used to encode information, and reactions among molecules are used to compute. Given that molecules are distributed throughout a solution, molecular computation is an inherently distributed form of computation that has the potential to involve many separate, parallel operations. Stochastic thermodynamics has been previously applied to model reaction networks (105), but much remains to be learned about the thermodynamic limits of molecular computation and how close they can be approached by selection of molecular species and tuning of their reactions.

As another example, stochastic thermodynamics might be a fruitful way to analyze alternative computational systems, such as stochastic computing. Finally, a major motivation of neuromorphic computing is reducing the energetic costs of communication among the components of a computer (136). However, to date, stochastic thermodynamics has not been used to help design such computers.

4. Conclusion

The energetic cost of computation is a long-standing, deep theoretical concern in fields ranging from statistical physics to computer science and biology. It has also recently become a major topic in the fight to reduce society's energy costs.

Although CS theory has mostly focused on computational resource costs regarding accuracy, time, and memory consumption, energetic costs are another important cost that has barely been considered in the CS theory community. Until quite recently, most of the research regarding the thermodynamics of computation has focused either on systems in equilibrium or archetypal examples of small systems, including only basic operations such as bit erasure.

In this paper, we argue that the recent results of stochastic thermodynamics can provide a mathematical framework for quantifying the energetic costs of realistic (both artificial and biological) computational devices. This may provide major benefits for the design of future artificial computers. It may also provide important new insights into the biological computers. Finally, by combining CS theory with the theoretical tools of stochastic thermodynamics, we

may uncover important new insights into the mathematical nature of all physical systems that perform computation in our universe.

Data, Materials, and Software Availability. There are no data underlying this work.

1. L. Cailloce, New technologies' wasted energies. *CNRS News*, 2018. https://news.cnrs.fr/articles/new-technologies-wasted-energies. Accessed 1 October 2023.
2. K. Bettayeb, Making applications more energy-efficient. *CNRS News*, 2022. https://news.cnrs.fr/articles/making-applications-more-energy-efficient. Accessed 1 October 2023.
3. D. Zhao *et al.*, "A green(er) world for A.I." in *2022 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* (IEEE, 2022), pp. 742–750.
4. L. Brillouin, Maxwell's demon cannot operate: Information and entropy. I. *J. Appl. Phys.* **22**, 334–337 (1951).
5. L. Szilárd, Ü ber die Entropieverminderung in einem thermodynamischen System bei Eingriffen intelligenter Wesen. *Z. Phys.* **53**, 840–856 (1929).
6. R. Landauer, Irreversibility and heat generation in the computing process. *IBM J. Res. Dev.* **5**, 183–191 (1961).
7. U. Seifert, Stochastic thermodynamics, fluctuation theorems and molecular machines. *Rep. Prog. Phys.* **75**, 126001 (2012).
8. C. Van den Broeck, M. Esposito, Ensemble and trajectory thermodynamics: A brief introduction. *Phys. A* **418**, 6–16 (2015).
9. L. Peliti, S. Pigolotti, *Stochastic Thermodynamics: An Introduction* (Princeton University Press, 2021).
10. N. Shiraishi, *An Introduction to Stochastic Thermodynamics: From Basic to Advanced* (Springer Nature, 2023), vol. 212.
11. N. Freitas, J. C. Delvenne, M. Esposito, Stochastic thermodynamics of nonlinear electronic circuits: A realistic framework for computing around kT. *Phys. Rev. X* **11**, 031064 (2021).
12. C. Y. Gao, D. T. Limmer, Principles of low dissipation computing from a stochastic circuit model. *Phys. Rev. Res.* **3**, 033169 (2021).
13. V. Balasubramanian, Brain power. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2107022118 (2021).
14. C. P. Kempes, D. Wolpert, Z. Cohen, J. Perez-Mercader, The thermodynamic efficiency of computations made in cells across the range of life. *Philos. Trans. R. Soc. A* **375**, 20160343 (2017).
15. B. Lambson, D. Carlton, J. Bokor, Exploring the thermodynamic limits of computation in integrated systems: Magnetic memory, nanomagnetic logic, and the Landauer limit. *Phys. Rev. Lett.* **107**, 010604 (2011).
16. W. B. Levy, V. G. Calvert, Communication consumes 35 times more energy than computation in the human cortex, but both costs are needed to predict synapse number. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2008173118 (2021).
17. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. *Science* **332**, 1196–1201 (2011).
18. H. L. Chen, D. Doty, D. Soloveichik, Deterministic function computation with chemical reaction networks. *Nat. Comput.* **13**, 517–534 (2014).
19. D. Woods *et al.*, Diverse and robust molecular algorithms using reprogrammable DNA self-assembly. *Nature* **567**, 366–372 (2019).
20. C. H. Bennett, The thermodynamics of computation - a review. *Int. J. Theor. Phys.* **21**, 905–940 (1982).
21. G. Piccinini, C. Maley, "Computation in physical systems https://plato.stanford.edu/archives/sum2021/entries/computation-physicalsystems/" in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed. (Metaphysics Research Lab, Stanford University, 2010).
22. Journal of Physics: Complexity, Focus Issue on Computation in Dynamical Systems, D. Wolpert, J. Korbel, M. Hofer, Eds. (IOP Science, 2024). https://iopscience.iop.org/collections/jpcomplex-240115-463. Accessed 1 May 2024.
23. M. Sipser, Introduction to the theory of computation. *ACM Sigact News* **27**, 27–29 (1996).
24. S. Arora, B. Barak, *Computational Complexity: A Modern Approach* (Cambridge University Press, 2009).
25. C. Moore, S. Mertens, *The Nature of Computation* (OUP Oxford, 2011).
26. A. M. Turing, "Computing machinery and intelligence" in *Parsing the Turing Test*, R. Epstein, G. Roberts, G. Beber, Eds. (Springer, 2009), pp. 23–65.
27. E. D. Demaine, J. Lynch, G. J. Mirano, N. Tyagi, "Energy-efficient algorithms" in *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, S. Devitt, I. Lanese, Eds. (Springer Cham, 2016), pp. 321–332.
28. T. M. Cover, *Elements of Information Theory* (John Wiley and Sons, 1999).
29. T. Sagawa, Thermodynamic and logical reversibilities revisited. *J. Stat. Mech.* **2014**, P03025 (2014).
30. J. M. Parrondo, J. M. Horowitz, T. Sagawa, Thermodynamics of information. *Nat. Phys.* **11**, 131–139 (2015).
31. T. E. Ouldridge, R. Brittain, "PR ten wolde" in *Energetics of Computing in Life and Machines*, P. S. D. H. Wolpert, C. P. Kempes, J. Grochow, Eds. (SFI Press, Santa Fe, NM, 2019).
32. A. Berut *et al.*, Experimental verification of Landauer's principle linking information and thermodynamics. *Nature* **483**, 187–189 (2012).
33. J. Song, S. Still, R. D. H. Rojas, I. P. Castillo, M. Marsili, Optimal work extraction and mutual information in a generalized Szilárd engine. *Phys. Rev. E* **103**, 052121 (2021).
34. G. Falasco, M. Esposito, Dissipation-time uncertainty relation. *Phys. Rev. Lett.* **125**, 120604 (2020).
35. T. E. Ouldridge, C. C. Govern, P. R. ten Wolde, The thermodynamics of computational copying in biochemical systems. *Phys. Rev. X* **7**, 021004 (2017).
36. A. Kolchinsky, D. H. Wolpert, Entropy production given constraints on the energy functions. *Phys. Rev. E* **104**, 034129 (2021).
37. A. Kolchinsky, D. H. Wolpert, Work, entropy production, and thermodynamics of information under protocol constraints. *Phys. Rev. X* **11**, 041024 (2021).
38. A. Kolchinsky, D. H. Wolpert, Dependence of dissipation on the initial distribution over states. *J. Stat. Mech. Theory Exp.* **2017**, 083202 (2017).
39. P. M. Riechers, M. Gu, Impossibility of achieving Landauer's bound for almost every quantum state. *Phys. Rev. A* **104**, 012214 (2021).
40. S. Ito, Geometric thermodynamics for the fokker-planck equation: Stochastic thermodynamic links between information geometry and optimal transport. *Inf. Geom.* **7**, 441–483 (2024).
41. N. Shiraishi, K. Funo, K. Saito, Speed limit for classical stochastic processes. *Phys. Rev. Lett.* **121**, 070601 (2018).
42. A. C. Barato, U. Seifert, Thermodynamic uncertainty relation for biomolecular processes. *Phys. Rev. Lett.* **114**, 158101 (2015).
43. J. M. Horowitz, T. R. Gingrich, Proof of the finite-time thermodynamic uncertainty relation for steady-state currents. *Phys. Rev. E* **96**, 020103 (2017).
44. K. Liu, Z. Gong, M. Ueda, Thermodynamic uncertainty relation for arbitrary initial states. *Phys. Rev. Lett.* **125**, 140602 (2020).
45. D. Collin *et al.*, Verification of the Crooks fluctuation theorem and recovery of RNA folding free energies. *Nature* **437**, 231–234 (2005).
46. F. Douarche, S. Ciliberto, A. Petrosyan, I. Rabbiosi, An experimental test of the Jarzynski equality in a mechanical experiment. *Europhys. Lett.* **70**, 593 (2005).
47. S. An *et al.*, Experimental test of the quantum Jarzynski equality with a trapped-ion system. *Nat. Phys.* **11**, 193–199 (2015).
48. J. M. Horowitz, T. R. Gingrich, Thermodynamic uncertainty relations constrain non-equilibrium fluctuations. *Nat. Phys.* **16**, 15–20 (2020).
49. T. Koyuk, U. Seifert, Thermodynamic uncertainty relation for time-dependent driving. *Phys. Rev. Lett.* **125**, 260604 (2020).
50. T. Van Vu *et al.*, Unified thermodynamic-kinetic uncertainty relation. *J. Phys. A Math. Theor.* **55**, 405004 (2022).
51. K. Funo, N. Shiraishi, K. Saito, Speed limit for open quantum systems. *New J. Phys.* **21**, 013006 (2019).
52. J. S. Lee, S. Lee, H. Kwon, H. Park, Speed limit for a highly irreversible process and tight finite-time Landauer's bound. *Phys. Rev. Lett.* **129**, 120603 (2022).
53. T. Van Vu, K. Saito, Topological speed limit. *Phys. Rev. Lett.* **130**, 010402 (2023).
54. T. Van Vu, K. Saito, Thermodynamic unification of optimal transport: Thermodynamic uncertainty relation, minimum dissipation, and thermodynamic speed limits. *Phys. Rev. X* **13**, 011013 (2023).
55. U. Seifert, Entropy production along a stochastic trajectory and an integral fluctuation theorem. *Phys. Rev. Lett.* **95**, 040602 (2005).
56. C. Jarzynski, Nonequilibrium equality for free energy differences. *Phys. Rev. Lett.* **78**, 2690–2693 (1997).
57. G. E. Crooks, Entropy production fluctuation theorem and the nonequilibrium work relation for free energy differences. *Phys. Rev. E* **60**, 2721 (1999).
58. C. Jarzynski, Hamiltonian derivation of a detailed fluctuation theorem. *J. Stat. Phys.* **98**, 77–102 (2000).
59. I. D. Terlizzi, M. Baiesi, Kinetic uncertainty relation. *J. Phys. A Math. Theor.* **52**, 02LT03 (2018).
60. Y. Hasegawa, Thermodynamic correlation inequality. *Phys. Rev. Lett.* **132**, 087102 (2024).
61. D. H. Wolpert, The stochastic thermodynamics of computation. *J. Phys. A Math. Theor.* **52**, 193001 (2019).
62. W. H. Zurek, Thermodynamic cost of computation, algorithmic complexity and the information metric. *Nature* **341**, 119–124 (1989).
63. E. DeGiuli, Random language model. *Phys. Rev. Lett.* **122**, 128301 (2019).
64. L. Zdeborová, F. Krzakala, Statistical physics of inference: Thresholds and algorithms. *Adv. Phys.* **65**, 453–552 (2016).
65. Y. Bahri *et al.*, Statistical mechanics of deep learning. *Annu. Rev. Condens. Matter Phys.* **11**, 501–528 (2020).
66. D. Randall, "Mixing [markov chain]" in *44th Annual IEEE Symposium on Foundations of Computer Science, Proceedings*, M. Agarwal *et al.*, Eds. (IEEE Computer Society, 2003), pp. 4–15.
67. S. Li *et al.*, Programming active cohesive granular matter with mechanically induced phase changes. *Sci. Adv.* **7**, eabe8494 (2021).
68. D. Gamarnik, C. Moore, L. Zdeborová, Disordered systems insights on computational hardness. *J. Stat. Mech. Theory Exp.* **2022**, 114015 (2022).

https://doi.org/10.1073/pnas.2321112121

69. L. A. Levin, Universal sequential search problems. *Probl. Peredachi Inf.* **9**, 115–116 (1973).
70. N. Garnier, S. Ciliberto, Nonequilibrium fluctuations in a resistor. *Phys. Rev. E* **71**, 060101 (2005).
71. R. van Zon, S. Ciliberto, E. G. D. Cohen, Power and heat fluctuation theorems for electric circuits. *Phys. Rev. Lett.* **92**, 130601 (2004).
72. S. Ciliberto, A. Imparato, A. Naert, M. Tanase, Heat flux and entropy produced by thermal fluctuations. *Phys. Rev. Lett.* **110**, 180601 (2013).
73. N. Freitas, J. C. Delvenne, M. Esposito, Stochastic and quantum thermodynamics of driven RLC networks. *Phys. Rev. X* **10**, 031005 (2020).
74. G. Paneru, D. Y. Lee, T. Tlusty, H. K. Pak, Lossless brownian information engine. *Phys. Rev. Lett.* **120**, 020601 (2018).
75. J. P. Pekola, Towards quantum thermodynamics in electronic circuits. *Nat. Phys.* **11**, 118–123 (2015).
76. J. P. Pekola, I. M. Khaymovich, Thermodynamics in single-electron circuits and superconducting qubits. *Annu. Rev. Condens. Matter Phys.* **10**, 193–212 (2019).
77. J. V. Koski, V. F. Maisi, T. Sagawa, J. P. Pekola, Experimental observation of the role of mutual information in the nonequilibrium dynamics of a Maxwell demon. *Phys. Rev. Lett.* **113**, 030601 (2014).
78. J. V. Koski, V. F. Maisi, J. P. Pekola, D. V. Averin, Experimental realization of a Szilard engine with a single electron. *Proc. Natl. Acad. Sci. U.S.A.* **111**, 13786–13789 (2014).
79. P. Helms, D. T. Limmer, Stochastic thermodynamic bounds on logical circuit operation. arXiv [Preprint] (2022). https://arxiv.org/abs/2211.00670 (Accessed 1 October 2023).
80. J. Kuang, X. Ge, Y. Yang, L. Tian, Modeling and optimization of low-power and gates based on stochastic thermodynamics. *IEEE Trans. Circuits Syst. II* **69**, 3729–3733 (2022).
81. K. Y. Camsari, R. Faria, B. M. Sutton, S. Datta, Stochastic *p*-bits for invertible logic. *Phys. Rev. X* **7**, 031014 (2017).
82. K. Y. Camsari, B. M. Sutton, S. Datta, p-bits for probabilistic spin logic. *Appl. Phys. Rev.* **6**, 011305 (2019).
83. W. A. Borders *et al.*, Integer factorization using stochastic magnetic tunnel junctions. *Nature* **573**, 390–393 (2019).
84. S. Misra *et al.*, Probabilistic neural computing with stochastic devices. *Adv. Mater.* **35**, 2204569 (2023).
85. N. S. Singh *et al.*, Cmos plus stochastic nanomagnets enabling heterogeneous computers for probabilistic inference and learning. *Nat. Commun.* **15**, 2685 (2024).
86. T. E. Ouldridge, D. H. Wolpert, Thermodynamics of deterministic finite automata operating locally and periodically. *New J. Phys.* **25**, 123013 (2023).
87. G. Manzano, G. Kardeş, E. Roldán, D. H. Wolpert, Thermodynamics of computations with absolute irreversibility, unidirectional transitions, and stochastic computation times. *Phys. Rev. X* **14**, 021026 (2024).
88. G. Kardeş, D. Wolpert, Inclusive thermodynamics of computational machines. arXiv [Preprint] (2022). http://arxiv.org/abs/2206.01165 (Accessed 1 October 2023).
89. I. Neri, Second law of thermodynamics at stopping times. *Phys. Rev. Lett.* **124**, 040601 (2020).
90. I. Neri, E. Roldan, F. Julicher, Statistics of infima and stopping times of entropy production and applications to active molecular processes. *Phys. Rev. X* **7**, 011019 (2017).
91. G. Falasco, M. Esposito, Dissipation-time uncertainty relation. *Phys. Rev. Lett.* **125**, 120604 (2020).
92. G. Manzano *et al.*, Thermodynamics of gambling demons. *Phys. Rev. Lett.* **126**, 080603 (2021).
93. P. Strasberg, J. Cerrillo, G. Schaller, T. Brandes, Thermodynamics of stochastic turing machines. *Phys. Rev. E* **92**, 042104 (2015).
94. A. Kolchinsky, D. H. Wolpert, Thermodynamic costs of Turing machines. *Phys. Rev. Res.* **2**, 033312 (2020).
95. D. H. Wolpert, A. Kolchinsky, Thermodynamics of computing with circuits. *New J. Phys.* **22**, 063047 (2020).
96. F. Tasnim, N. Freitas, D. H. Wolpert, Entropy production in communication channels. arXiv [Preprint] (2023). https://arxiv.org/abs/2302.04320 (Accessed 1 March 2024).
97. F. S. Gnesotto, F. Mura, J. Gladrow, C. P. Broedersz, Broken detailed balance and non-equilibrium dynamics in living systems: A review. *Rep. Prog. Phys.* **81**, 066601 (2018).
98. J. J. Hopfield, Kinetic proofreading: a new mechanism for reducing errors in biosynthetic processes requiring high specificity. *Proc. Natl. Acad. Sci. U.S.A.* **71**, 4135–4139 (1974).
99. J. Ninio, Kinetic amplification of enzyme discrimination. *Biochimie* **57**, 587–595 (1975).
100. J. Hopfield, T. Yamane, V. Yue, S. Coutts, Direct experimental evidence for kinetic proofreading in amino acylation of tRNAIle. *Proc. Natl. Acad. Sci. U.S.A.* **73**, 1164–1168 (1976).
101. R. Bar-Ziv, T. Tlusty, A. Libchaber, Protein-DNA computation by stochastic assembly cascade. *Proc. Natl. Acad. Sci. U.S.A.* **99**, 11589–11592 (2002).
102. J. T. Reardon, A. Sancar, Thermodynamic cooperativity and kinetic proofreading in DNA damage recognition and repair. *Cell Cycle* **3**, 139–142 (2004).
103. T. W. McKeithan, Kinetic proofreading in T-cell receptor signal transduction. *Proc. Natl. Acad. Sci. U.S.A.* **92**, 5042–5046 (1995).
104. T. Schmiedl, U. Seifert, Stochastic thermodynamics of chemical reaction networks. *J. Chem. Phys.* **126**, 044101 (2007).
105. R. Rao, M. Esposito, Nonequilibrium thermodynamics of chemical reaction networks: Wisdom from stochastic thermodynamics. *Phys. Rev. X* **6**, 041064 (2016).
106. H. C. Berg, E. M. Purcell, Physics of chemoreception. *Biophys. J.* **20**, 193–219 (1977).
107. C. C. Govern, P. R. Ten Wolde, Optimal resource allocation in cellular sensing systems. *Proc. Natl. Acad. Sci. U.S.A.* **111**, 17486–17491 (2014).
108. S. J. Bryant, B. B. Machta, Physical constraints in intracellular signaling: The cost of sending a bit. *Phys. Rev. Lett.* **131**, 068401 (2023).
109. G. Lan, P. Sartori, S. Neumann, V. Sourjik, Y. Tu, The energy-speed-accuracy trade-off in sensory adaptation. *Nat. Phys.* **8**, 422–428 (2012).
110. V. Ngampruetikorn, D. J. Schwab, G. J. Stephens, Energy consumption and cooperation for optimal sensing. *Nat. Commun.* **11**, 975 (2020).
111. S. Still, Thermodynamic cost and benefit of memory. *Phys. Rev. Lett.* **124**, 050601 (2020).
112. C. W. Lynn, D. S. Bassett, The physics of brain network structure, function and control. *Nat. Rev. Phys.* **1**, 318–332 (2019).
113. J. J. Harris, R. Jolivet, D. Attwell, Synaptic energy use and supply. *Neuron* **75**, 762–777 (2012).
114. W. B. Levy, R. A. Baxter, Energy efficient neural codes. *Neural Comput.* **8**, 531–543 (1996).
115. H. B. Barlow *et al.*, Possible principles underlying the transformation of sensory messages. *Sens. Commun.* **1**, 217–233 (1961).
116. C. W. Lynn, C. M. Holmes, W. Bialek, D. J. Schwab, Decomposing the local arrow of time in interacting systems. *Phys. Rev. Lett.* **129**, 118101 (2022).
117. C. W. Lynn, E. J. Cornblath, L. Papadopoulos, M. A. Bertolero, D. S. Bassett, Broken detailed balance and entropy production in the human brain. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2109889118 (2021).
118. Y. S. Perl *et al.*, Nonequilibrium brain dynamics as a signature of consciousness. *Phys. Rev. E* **104**, 014411 (2021).
119. P. Zhang, C. Moore, Scalable detection of statistically significant communities and hierarchies, using message passing for modularity. *Proc. Natl. Acad. Sci. U.S.A.* **111**, 18144–18149 (2014).
120. A. Clauset, C. Moore, M. E. Newman, Hierarchical structure and the prediction of missing links in networks. *Nature* **453**, 98–101 (2008).
121. D. S. Bassett, M. S. Gazzaniga, Understanding complexity in the human brain. *Trend. Cogn. Sci.* **15**, 200–209 (2011).
122. M. E. Newman, Modularity and community structure in networks. *Proc. Natl. Acad. Sci. U.S.A.* **103**, 8577–8582 (2006).
123. E. Ravasz, A. L. Barabási, Hierarchical organization in complex networks. *Phys. Rev. E* **67**, 026112 (2003).
124. H. A. Simon, "The architecture of complexity" in *Facets of Systems Science*, G. J. Klir, Ed. (Springer, 1991), pp. 457–476.
125. M. Yim *et al.*, Modular self-reconfigurable robot systems [grand challenges of robotics]. *IEEE Robot. Autom. Mag.* **14**, 43–52 (2007).
126. G. Schlosser, G. P. Wagner, *Modularity in Development and Evolution* (University of Chicago Press, 2004).
127. G. P. Wagner, M. Pavlicev, J. M. Cheverud, The road to modularity. *Nat. Rev. Genet.* **8**, 921–931 (2007).
128. E. Mones, L. Vicsek, T. Vicsek, Hierarchy measure for complex networks. *PloS One* **7**, e33799 (2012).
129. A. B. Boyd, D. Mandal, J. P. Crutchfield, Thermodynamics of modularity: Structural costs beyond the Landauer bound. *Phys. Rev. X* **8**, 031036 (2018).
130. X. Ge, L. Yen, Information thermodynamics communications. *IEEE Wirel. Commun.* **30**, 130–137 (2022).
131. I. F. Akyildiz, M. C. Vuran, *Wireless Sensor Networks* (John Wiley and Sons, 2010).
132. L. Qian, E. Winfree, J. Bruck, Neural network computation with DNA strand displacement cascades. *Nature* **475**, 368–372 (2011).
133. C. E. Arcadia *et al.*, Multicomponent molecular memory. *Nat. Commun.* **11**, 691 (2020).
134. C. E. Arcadia *et al.*, Leveraging autocatalytic reactions for chemical domain image classification. *Chem. Sci.* **12**, 5464–5472 (2021).
135. A. A. Agiza *et al.*, Digital circuits and neural networks based on acid-base chemistry implemented by robotic fluid handling. *Nat. Commun.* **14**, 496 (2023).
136. D. Markovic, A. Mizrahi, D. Querlioz, J. Grollier, Physics for neuromorphic computing. *Nat. Rev. Phys.* **2**, 499–510 (2020).