Thermodynamic costs of Turing machines

Artemy Kolchinsky and David H. Wolpert*
Santa Fe Institute, Santa Fe, New Mexico 87501, USA

(Received 10 December 2019; accepted 5 August 2020; published 26 August 2020)

Turing machines (TMs) are the canonical model of computation in computer science and physics. We combine techniques from algorithmic information theory and stochastic thermodynamics to analyze the thermodynamic costs of TMs. We consider two different ways of realizing a given TM with a physical process. The first realization is designed to be thermodynamically reversible when fed with random input bits. The second realization is designed to generate less heat, up to an additive constant, than any realization that is computable (i.e., consistent with the physical Church-Turing thesis). We consider three different thermodynamic costs: The heat generated when the TM is run on each *input* (which we refer to as the "heat function"), the minimum heat generated when a TM is run with an input that results in some desired *output* (which we refer to as the "thermodynamic complexity" of the output, in analogy to the Kolmogorov complexity), and the expected heat on the input distribution that minimizes entropy production. For universal TMs, we show for both realizations that the thermodynamic complexity of any desired output is bounded by a constant (unlike the conventional Kolmogorov complexity), while the expected amount of generated heat is infinite. We also show that any computable realization faces a fundamental trade-off among heat generation, the Kolmogorov complexity of its heat function, and the Kolmogorov complexity of its input-output map. We demonstrate this trade-off by analyzing the thermodynamics of erasing a long string.

DOI: 10.1103/PhysRevResearch.2.033312

I. INTRODUCTION

The relationship between thermodynamics and information processing has been an important area of research since at least the 1960s, when Landauer proposed that any process which erases a bit of information must release at least $kT \ln 2$ of heat into its environment [1–16]. This research has greatly benefited from the dramatic progress in nonequilibrium statistical physics in the past few decades, in particular the development of trajectory-based and stochastic thermodynamics [17–19]. These developments now permit us to quantify and analyze heat, work, entropy production, and other thermodynamic properties of individual trajectories in far-from-equilibrium systems. They have also have led to a much deeper understanding of the relationship between thermodynamics and information processing, both for information erasure [20–25] and other more elaborate computations [10,26–39].

In this paper we extend this line of research by deriving new results on the thermodynamic costs of performing general computations, as formalized by the notion of *Turing machines* (TMs). A TM is an abstraction of a conventional modern computer, which run programs written in a conventional programming language (C, Python, etc.) [40–45]. A TM reads an input string of arbitrary length (a "program") and runs until it produces an output string. In the same way that any modern computer can simulate other computers (e.g., via an emulator), there exists an important class of TMs called *universal Turing Machines* (UTMs), each of which is able to simulate the operation of any other TM.

TMs are a keystone of the theory of computation [46] and touch on several foundational issues that lie at the intersection of mathematics and philosophy, such as whether $\mathsf{P} = \mathsf{NP}$ and Gödel's incompleteness theorems [47]. Their importance is partly due to the celebrated *Church-Turing thesis*, which postulates that any function that can be computed by a sequence of formal operations can also be computed by some TM [48–50]. For this reason, in computer science, a function is called *computable* if and only if it can be carried out by a TM [42]. TMs also play important roles in many facets of modern physics. For instance, TMs are used to formalize the difference between easy and hard computational problems in quantum computing [51–55]. There has also been some speculative, broader-ranging work on whether the foundations of physics may be restricted by some of the properties of TMs [56,57]. Finally, there has been extensive investigation of the *physical Church-Turing thesis*, which states that any function that can be implemented by a physical process can also be computed with a TM [51,53,58–70].

One of the most important concepts in the theory of TMs is *Kolmogorov complexity*. The Kolmogorov complexity of a string y, written as $K(y)$, is the length of the shortest input program which causes a UTM to produce y as the output (formal definitions are provided in Sec. II B). The Kolmogorov complexity of a string y captures the amount of randomness in

*Complexity Science Hub, Vienna; Arizona State University; http://davidwolpert.weebly.com

y, because a string with a nonrandom pattern can be produced by a short input program. For example, the string containing the first billion digits of π can be generated by running a very short program and so has small Kolmogorov complexity. In contrast, for a random string y without any patterns, the shortest program that produces y is a program of the type "`print 'y'`", which has about the same length as y. An important variant of Kolmogorov complexity is the *conditional Kolmogorov complexity* of y given x, written $K(y|x)$, which is the length of the shortest program which causes a UTM to produce y as output, when the UTM is provided with x as an additional input. Kolmogorov and conditional Kolmogorov complexity have many formal connections with entropy and conditional entropy from Shannon's information theory [43] and are studied in a field called *algorithmic information theory* (AIT) [42,71].

In this paper, we combine techniques from AIT and stochastic thermodynamics to analyze the thermodynamics of TMs. We imagine a discrete-state physical system that is coupled to a heat bath at temperature T and that evolves under the influence of a driving protocol. We identify the initial and final states of the physical system with the logical inputs and outputs of some TM, so that the dynamics over the states of the physical system corresponds to a computation performed by the TM. We refer to a physical process that is consistent with the laws of thermodynamics and whose dynamics correspond to the input-output map of a TM as a *realization* of that TM.

We derive numerous results that concern the thermodynamic properties of realizations of TMs. The core underlying idea behind these results is that the *logical properties* a given TM (such as the structure of the TM's input-output map or the Kolmogorov complexity of its inputs and outputs) provide constraints on the *thermodynamic costs* incurred by realizations of that TM (such as the amount of heat those realizations generate). Some of our results relate logical properties and thermodynamic costs at the ensemble level (i.e., relative to a probability distribution over computational trajectories of a TM), thereby building on the thermodynamic analysis initiated by Landauer and others. In addition to these, many of our results also relate logical properties and thermodynamic costs at the level of individual computational trajectories (i.e., individual runs of the TM), which goes beyond most existing research on thermodynamics of computation.

A. Summary of results

We investigate three different kinds of thermodynamic costs for a given realization of a TM:

(1) The amount of heat that is generated by running the realization of a given (universal or nonuniversal) TM on each individual input x. We refer to the map from inputs to their associated heat values as the *heat function* of the TM's realization and write it as $Q(x)$.

(2) The minimal amount of heat generated by running the realization of a given TM on some individual input that results in a desired output y. Here we assume that the TM is universal, so that it can in principle produce any output. This second cost is a function of the desired output y, rather than of the input x, and can be viewed as a thermodynamic analog of conventional Kolmogorov complexity. For this reason, we refer to this cost as the *thermodynamic complexity* of y.

(3) The ensemble-level expected heat $\langle Q \rangle$ generated by the realization of a TM, evaluated for the input distribution that minimizes entropy production (EP). For this cost, we again focus on the case of universal TMs.

In general, there are many physical processes that are realizations of the same TM, which can have different thermodynamic costs from one another. In this paper we consider the above three thermodynamic costs for two important types of realizations. The first realization we consider, which is called the *coin-flipping* realization, is constructed to be thermodynamically reversible when input programs are sampled from the "coin-flipping" distribution $p(x) \propto 2^{-\ell(x)}$, where $\ell(x)$ indicates the length of string x. This input distribution arises by feeding random bits into a TM (hence its name) and plays a fundamental role in AIT.

We show that the heat function of the coin-flipping realization of a given TM is proportional to $\ell(x)$ minus a "correction term" which reflects the logically irreversibility of the input-output map computed by the TM. Importantly, when the realized TM is a universal TM U, this correction term can be related to the Kolmogorov complexity of the output of U on input x. In this case, the heat function is given by

$$Q_{\text{coin}}(x) = kT \ln 2 \{\ell(x) - K[\phi_U(x)]\} + O(1), \qquad (1)$$

where $\phi_U(x)$ indicates the output of U on input x and $O(1)$ indicates equality up to an additive constant independent of x (see Sec. I C for a formal definition). Thus, up to an additive constant, the heat generated by running input x on the coin-flipping realization of some UTM U is proportional to the *excess* length of the input program x, over and above the length of the shortest program for U that produces the same output as x.

It follows from Eq. (1) that if x is the shortest program for U that produces output $\phi_U(x)$, then $Q_{\text{coin}}(x) = O(1)$. This means that by running the shortest program x that produces some desired y as output, one can produce that y for an amount of heat that is bounded by a constant. Thus, the thermodynamic complexity for the coin-flipping realization is a bounded function, unlike the Kolmogorov complexity, which grows arbitrarily large [42]. On the other hand, we also show that when inputs are sampled from the coin-flipping distribution, the expected heat $\langle Q \rangle$ generated by the coin-flipping realization of a UTM is infinite. This holds even though the heat necessary to run the UTM on any given input x is finite.

The second realization we analyze is inspired by the physical Church-Turing thesis. To begin, we refer to a realization of a TM with heat function Q as a *computable realization* if the function $x \mapsto Q(x)/kT$ is computable [i.e., there exist some TM that takes as input any desired x and outputs the corresponding heat value $Q(x)$ in units of kT]. Under common interpretations of the physical Church-Turing thesis [50,53,59–61,64], any realization that is *actually* constructable in the real world must be computable; in other words, a noncomputable realization is a hypothetical physical process which does not violate any laws of thermodynamics but which nonetheless cannot be constructed because of computational constraints. Motivated by these observations, we define the so-called

dominating realization of a TM M to be "optimal" in the following sense: The heat it generates on any input x is smaller than the heat generated by any computable realization of M on x, up to an additive constant which does not depend on x.[1] The heat function of the dominating realization is proportional to the conditional Kolmogorov complexity of the output given the input,

$$Q_{\rm dom}(x) = kT \ln 2\, K[x|\phi_M(x)], \tag{2}$$

where $\phi_M(x)$ indicates the output of TM M on input x. We show that this heat function is smaller than the heat function Q of any computable realization of M,

$$Q_{\rm dom}(x) \leqslant Q(x) + O(1). \tag{3}$$

Note that this result holds whether or not M is a UTM.

For the special case where M is a UTM, we show that for any desired output y, the thermodynamic complexity of y under the dominating realization is bounded by a constant that is independent of y, just like for the coin-flipping realization. Moreover, for the dominating realization there is a simple scheme for choosing the input x that will produce any desired output y with a bounded amount of heat. This differs from the coin-flipping realization, where one must know the shortest program that generates y in order to produce y with a bounded amount of heat (in general, finding the shortest program to produce a given output y is not computable).

Finally, we consider the expected heat that is generated by the dominating realization, given some probability distribution over input programs. A natural input distribution to consider is the one that minimizes the entropy production of the dominating realization. As for the coin-flipping realization, we show that the expected heat across inputs sampled from this distribution is infinite.

There are two important caveats concerning the dominating realization. First, while the dominating realization is better than any computable realization, in the sense of Eq. (3), it itself is not computable. This is because its heat function is defined in terms of the conditional Kolmogorov complexity, which is not a computable function. Nonetheless, as we discuss below, one can always define a sequence of computable realizations whose heat functions approach $Q_{\rm dom}$ from above. Thus, the dominating realization presents a fundamental bound on the heat generation of computable realizations, and this bound is achievable in the limit.

Second, for a given TM M, Eq. (3) states that the heat generated by the dominating realization on input x, $Q_{\rm dom}(x)$, is smaller than the heat generated by any computable realization, $Q(x)$, up to an additive constant that does not depend on x. This additive constant, however, can depend on the particular alternative realization of M that is being compared, i.e., on the choice of comparison heat function Q. In fact, depending on the alternative realization, that additive constant can be arbitrarily large and negative. This means that for a given TM M and some particular choice of input program x, there may exist alternative realizations of M that generate arbitrarily less heat than the dominating realization. It turns out, however, that the difference between $Q_{\rm dom}(x)$ and $Q(x)$ is upper bounded by the sum of the Kolmogorov complexity of the input-output function ϕ_M and the Kolmogorov complexity of the comparison heat function Q. Using this result, we show that any computable realization that produces output y from input x faces a fundamental cost of $K(x|y)$, which can be paid either by producing a large amount of heat, by computing an input-output map with high complexity or by having a heat function with high complexity.

The paper is laid out as follows. In the following subsections, we review relevant prior work and introduce notation. In Sec. II, we define TMs and review some relevant results from AIT. In Sec. III, we review the basics of statistical physics and discuss how a TM can be implemented as a physical system. We present our main results on the coin-flipping and dominating realizations in Sec. IV and Sec. V. In Sec. VI, we demonstrate the trade-off between heat and complexity by analyzing the thermodynamics of erasing a long string. In the last section we discuss potential directions for future research.

B. Prior work on thermodynamics of TMs

Some of the earliest work on the thermodynamics of TMs focused on TMs with deterministic and logically reversible dynamics [72,73]. Logically reversible TMs can perform computations without generating any heat or entropy production, at the cost of having to store additional information in their output, which logically irreversible TMs do not need to store. Due to the thermodynamic costs that would arise in reinitializing that extra stored information, there are some subtleties in calculating the thermodynamic cost of running a "complete cycle" of any logically reversible TM [34]. (See also Refs. [74,75] for a discussion of the relationship between thermodynamic and logical reversibility.) Logically reversible TMs form a special subclass of TMs and require special definitions of universality [76]. In this work, we focus on the thermodynamics of general-purpose TMs, whose computations will generally be logically irreversible. However, we will sometimes also discuss how our results apply in the logically reversible case.

More recently, Ref. [36] analyzed the thermodynamics of logically reversible TMs with stochastic forward-backward dynamics along a computational trajectory, which causes the state of the TM to become more uncertain with time.[2] This model incurs nonzero entropy production, even though each computational trajectory encodes a logically reversible computation. Note that this entropy production could in principle be made arbitrarily small by driving the TM forward with momentum (e.g., by coupling it to a large flywheel). In this

[1]Note that generating minimal heat is different from generating minimal EP. For example, the coin-flipping realization of a TM is thermodynamically reversible for the coin-flipping distribution over inputs x and thus generates zero EP when run on inputs sampled from that distribution. However, that does not mean that it generates less heat on any particular input x relative to the heat generated by another realization of the same TM on x.

[2]This kind of "stochastic TM" should not be confused with what are called "nondeterministic TMs" or "probabilistic TMs" in the computer science literature [40,44].

work, we will ignore possible stochasticity in the progression of a TM along its computational trajectory.

Finally, there has been recent work which interprets the coin-flipping distribution over strings x, as defined in Sec. IV, as a "Boltzmann distribution" induced by the "energy function" $\ell(x)$ [77]. Doing this allows one to formulate a set of equations concerning TMs that are formal analogs of Maxwell's relations for equilibrium thermodynamic systems.

In our own earlier work, we began to analyze the thermodynamic complexity of computing desired outputs, focusing on the coin-flipping realization and a three-tape UTM [78]. We first showed explicitly how to construct a system that is thermodynamically reversible for the coin-flipping distribution and then derived the associated heat function. We showed that for this realization, the minimal amount of heat needed to compute any given output y equals the Kolmogorov complexity of y, plus what we characterized as a "correction term." In other, more recent work, we rederived these results using stochastic thermodynamics and single-tape machines [79].

In this paper, we extend this earlier work on the coin-flipping realization. For simplicity, we consider the thermodynamics of systems that implement the entire computation of a given UTM in some fixed time interval. (In contrast, our earlier work considered systems that implement a given UTM's update function iteratively, taking varying amounts of time to halt, depending on the input to the UTM.) We then go further and use Levin's Coding theorem to show that the thermodynamic complexity of the coin-flipping realization is bounded, even though the conventional Kolmogorov complexity function is not. We also extend this earlier work by showing that the coin-flipping realization generates infinite expected heat when inputs are sampled from the coin-flipping distribution.

The other main contributions of this paper concern the thermodynamic costs of the dominating realization. These results are related to a series of groundbreaking papers begun by Zurek [5,6,80–86]. Those papers were generally written before the widespread adoption of trajectory-based analyses of thermodynamics [18] and contained a semiformal argument that computing an output string y from an input x has a minimal "thermodynamic cost" of at least $K(x|y)$. Even though that semiformal argument is quite different from our analysis, the same "thermodynamic cost" function also appears in our analysis of the dominating realization. We discuss connections between our results and this earlier work in more detail in Sec. VI.

C. Notation

We use uppercase letters, such as X and Y, to indicate random variables. We use lowercase letters, like x and y, to indicate their outcomes. We use p_X to indicate a probability distribution over random variable X and $p_{X|Y}$ to indicate a conditional probability distribution of random variable X given random variable Y. We also use $p_{X|Y=y}$ to indicate the probability distribution of X conditioned on one particular outcome $Y = y$. Finally, we use $\operatorname{supp} p_X$ to indicate the support of distribution p_X and notation like $\langle f(X)\rangle_{p_X} = \sum_x p_X(x) f(x)$ to indicate expectations.

A *partial function* $f : A \to B$ is a map from some subset of A, which is called the domain of definition of f, into B. We write $\operatorname{dom} f \subseteq A$ to indicate the domain of definition of f and $\operatorname{img} f := \{f(a) : a \in \operatorname{dom} f\}$ to indicate the image of f. The value of $f(a)$ is undefined for any $a \notin \operatorname{dom} f$.

For any set A, we use A^* to indicate the set of finite strings of elements from A. We use A^∞ to indicate the set of infinite strings of elements from A. In particular, $\{0, 1\}^*$ indicates the set of all finite binary strings. Note that for any finite A, A^* is a countably infinite set.

The Kronecker delta is indicated by $\delta(\cdot, \cdot)$. We sometimes write δ_x to indicate a δ-function probability distribution over outcome x of random variable X, $\delta_x(x') = \delta(x, x')$.

We use standard asymptotic notation, such as $f(x) = g(x) + O(1)$, which indicates that $|f(x) - g(x)| \leqslant \kappa$ for some $\kappa \in \mathbb{R}$ and all x. Similarly, notation like $f(x) \leqslant g(x) + O(1)$ indicates that $f(x) - g(x) \leqslant \kappa$ for some $\kappa \in \mathbb{R}$ and all x.

II. BACKGROUND ON TURING MACHINES AND AIT

A. Turing machines

In its canonical definition, a TM comprises three variables and a rule for their joint dynamics. First, there is a *tape* variable whose state is a semi-infinite string $s \in A^\infty$, where A is a finite set of tape symbols which includes a special *blank* symbol. Second, there is a *pointer* variable $v \in \{1, 2, 3, \dots\}$, which is interpreted as specifying a "position" on the tape (i.e., an index into the infinite-dimensional vector s). Finally, there is a *head* variable h whose state belongs to a finite set, which includes a specially designated *start state* and a specially designated *halt state*.

The TM starts with its head in the start state, the pointer set to position 1, and its tape containing some finite string of nonblank symbols, followed by blank symbols. The joint state of the tape, pointer, and head evolves over time according to a discrete-time *update function*. If during that evolution the head ever enters its halt state, then that is interpreted as the computation being completed. If and when the computation completes, we say that the TM has then *computed* its output, which is specified by the state of its tape at that time. Importantly, for some inputs, a TM might never complete its computation, i.e., it may go into an infinite loop and never enter the halt state. The operation of a TM is illustrated in a schematic way in Fig. 1. A more formal definition of a TM and the update function is provided in Appendix A.

There many other variants of TMs that have been considered in the literature, including ones with multiple tapes and multiple heads. However, all of these variants are computationally equivalent: Any computation that can be carried out with a particular TM variant can also be carried out with some TM that possesses a single tape and a single head [34,40,87].

For simplicity of analysis, we make two assumptions about the TMs analyzed in this paper, none of which affect the computational capabilities of the TMs. First, we assume that the tape alphabet A contains the binary symbols 0 and 1 and that these are the only nonblank symbols present on the tape at the beginning of the computation. Second, we assume that any TM we consider is designed so that if and when it reaches a halt state, its tape will contain a string from $\{0, 1\}^*$ followed

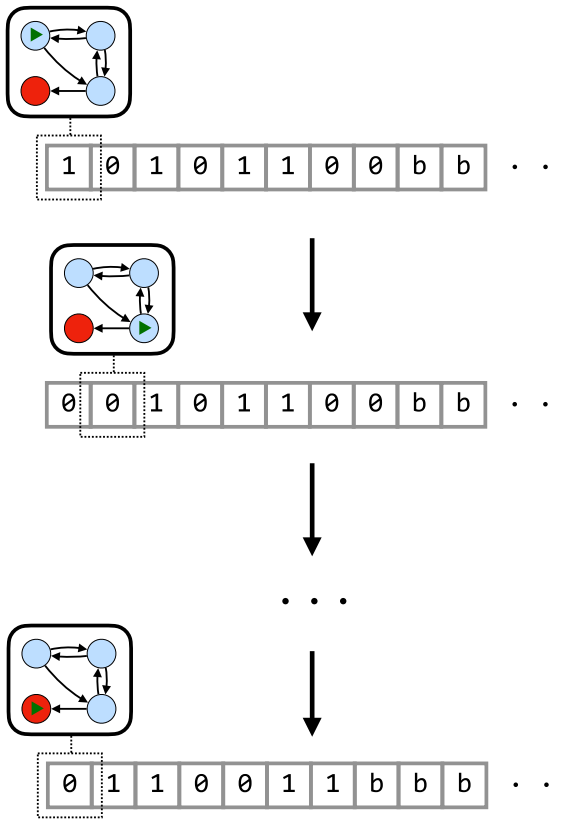

FIG. 1. A TM performing a computation. The update function is applied over a sequence of steps, causing the finite-state head (rounded box, states are colored circles) to move along an infinite tape of symbols (b indicates a special "blank" symbol). During each step, the head can read or write the tape symbol in the current position, move left or right along the tape, and change its current state (green triangle). The computation completes if and when the head reaches its halt state (red circle).

by all blank symbols, and the pointer will be set to 1 (i.e., returned to the start of the tape). This assumption of a "standardized" halt state properly accounts for the thermodynamic costs of running a complete cycle of the TM. For instance, after this standardized halt state is reached, the output of the TM can be moved from the tape onto an off-board storage device and a new input can be moved from another off-board storage device onto the tape, thus preparing the TM to run another program. Importantly, both of these operations can in principle be performed without incurring thermodynamic costs [34].

Given the above assumptions, one can represent the computation performed by any TM M as a partial function over the set of finite-length bit strings $\{0,1\}^*$ (see Appendix A), which we write as $\phi_M : \{0,1\}^* \to \{0,1\}^*$. In this notation, $\phi_M(x) = y$ indicates that when TM M is started with input program x, it eventually halts and produces the output string y. Note that ϕ_M is a partial function because it is undefined for any input x for which M does not eventually halt [40,42,43]. Thus, dom ϕ_M (the domain of definition of ϕ_M) is the set of all input strings on which M eventually halts, which is sometimes called the "halting set of M" in the literature.

As mentioned in the Introduction, a UTM is a TM that can simulate any other TM. More precisely, given some UTM U and any other TM M, there exists an "interpreter program" $\sigma_{U,M}$ such that for any input x of M, $\phi_U(\sigma_{U,M}, x) = \phi_M(x)$. Intuitively, this means that there exists programming languages which are "universal," meaning they can run programs written in any programming language, after appropriate translation from that other language. Note that, since M can itself be a UTM, any UTM can simulate any other UTM.

Given some partial function $f : \{0,1\}^* \to \{0,1\}^*$ and a TM M, we sometimes say that M *computes* f if $\phi_M = f$ [i.e., dom ϕ_M = dom f and $\phi_M(x) = f(x)$ for all $x \in$ dom f]. We say that "f is computable" if there exists some TM M that computes f. Importantly, there exist functions $\{0,1\}^* \to \{0,1\}^*$ which are *uncomputable*, meaning they cannot be computed by any TM. The existence of noncomputable functions follows immediately from the fact that there are an uncountable number of functions $\{0,1\}^* \to \{0,1\}^*$ but only a countable number of TMs. As an example of an uncomputable function, there is no TM which can take any input string x and output a 0 or 1, corresponding to whether or not x is in the halting set of some given UTM U [40,42,43].

We say that the halting set dom ϕ_M is a *prefix-free set* if for any input $x \in$ dom ϕ_M there is no other input $x' \in$ dom ϕ_M that is a proper prefix of x. In this paper we only consider TMs M such that dom ϕ_M is prefix-free, which are sometimes called "prefix TMs" in the literature. Importantly, the set of all prefix TMs is computationally equivalent to the set of all TMs: Any prefix TM can be simulated by some nonprefix TM and vice versa. However, prefix TMs have many useful mathematical properties and so have become conventional in the AIT literature [42]. See Appendix A for a discussion of how prefix TMs can be constructed.

Above we discussed computable functions from binary strings to binary strings, $\{0,1\}^* \to \{0,1\}^*$. It is also possible treat a finite binary string as an encoding of a pair of binary finite strings. More precisely, assume that along with any TM M, there is a one-to-one *pairing function* $\langle a, b\rangle$, which maps pairs of binary strings to single binary strings and whose image is a prefix-free set. By inverting the pairing function, one can uniquely interpret a single binary string as a pair of strings. This allows to interpret the domain and/or image of a partial function computed by a TM as a subset of $\{0,1\}^* \times \{0,1\}^*$ rather than a subset of $\{0,1\}^*$. We will write $\phi_M(a, b)$ as shorthand for $\phi_M(\langle a, b\rangle)$.

It is also possible to interpret a binary string as encoding an integer [42] or (by inverting the pairing function) as encoding two integers that specify a rational number. This allows us to formalize the computability of a function from binary strings to integers, $f : \{0,1\}^* \to \mathbb{Z}$, or from binary strings to rationals, $f : \{0,1\}^* \to \mathbb{Q}$. For a real-valued function $f : \{0,1\}^* \to \mathbb{R}$, we say that f is *computable* if there is a TM that can produce an approximation of $f(x)$ accurate to within any desired precision. Formally, f is computable if there exists some TM M such that $|\phi_M(x, n) - f(x)| \leqslant 2^{-n}$ for all $x \in$ dom f and $n \in \mathbb{N}$.

B. Algorithmic information theory

As mentioned in the Introduction, the *Kolmogorov complexity* of any bit string $x \in \{0,1\}^*$ is the length of the shortest

program which leads a given UTM U to produce x as output. We write this formally as

$$K_U(x) := \min_{z:\phi_U(z)=x} \ell(z). \tag{4}$$

The Kolmogorov complexity is unbounded: For any UTM U and any finite κ, there exists a string x such that $K_U(x) > \kappa$ (this follows from the fact that $\{0, 1\}^*$ is an infinite set, while only a finite number of different outputs can be produced by programs of length κ or less). Moreover, K_U is an uncomputable function. This implies that if the physical Church-Turing thesis is true, then no real-world physical system can take any desired string x as input and produce the value of $K_U(x)$ as output. On the other hand, Kolmogorov complexity can be bounded from above,[3] and it is possible to derive many formal results about its properties [42].

One can define the Kolmogorov complexity not just for strings but also for computable partial functions. Recall from the previous section that given any UTM U and TM M, there is a corresponding "interpreter program" $\sigma_{U,M}$, which can be used by U to simulate M on any input x. The Kolmogorov complexity of a computable function f is defined as the minimal Kolmogorov complexity of any interpreter program for U that simulates a TM that computes f:

$$K_U(f) := \min_{M:\phi_M=f} \ell(\sigma_{U,M}). \tag{5}$$

Similarly, the Kolmogorov complexity of some computable function $f : \{0, 1\}^* \to \mathbb{R}$ is given by the length of the shortest interpreter program that approximates f to arbitrary precision. $K_U(f)$ is undefined if f is not computable.

Above we defined Kolmogorov complexity relative to some particular choice of UTM U. In fact, the choice of U is only relevant up to an additive constant. To be precise, for any two UTMs U and U', the "invariance theorem" [42] states that

$$K_{U'}(x) = K_U(x) + O(1). \tag{6}$$

Given this result along with the unboundedness of K_U, for any two UTMs U and U' and any desired $\epsilon > 0$,

$$1 - \epsilon < K_U(x)/K_{U'}(x) < 1 + \epsilon \tag{7}$$

for all but a finite number of strings x (of the infinite set of all possible such strings). For many purposes, this allows us to dispense with specifying the precise UTM U when referring to the Kolmogorov complexity of a string x and simply write $K(x)$ instead of $K_U(x)$.

Finally, the *conditional Kolmogorov complexity* of $x \in \{0, 1\}^*$ given $y \in \{0, 1\}^*$ is the length of the shortest program that, when paired with y and then fed into a UTM U, produces x as output:

$$K_U(x|y) := \min_{z:\phi_U(z,y)=x} \ell(z). \tag{8}$$

[3]For any given x, one can compute an improving upper bound on $K_U(x)$ by running multiple copies of U in parallel with different input programs, while keeping track of the length of the shortest program found so far that has halted and produced output x [42]. In the limit, this procedure will converge on $K_U(x)$.

Like regular Kolmogorov complexity, the conditional Kolmogorov complexity is unbounded and uncomputable, though one can derive increasingly tight upper bounds on it. In addition, like regular Kolmogorov complexity, the conditional Kolmogorov complexity defined relative to two UTMs U and U' differs only up to an additive constant which does not depend on x or y [42],

$$K_{U'}(x|y) = K_U(x|y) + O(1). \tag{9}$$

Accordingly, for many purposes we can simply write $K(x|y)$, without specifying the precise UTM U.

III. BACKGROUND ON STATISTICAL PHYSICS

A. Physical setup

We consider a physical system with a countable state space $\mathcal{X}$. In practice, $\mathcal{X}$ will often be a "mesoscopic" coarse-graining of some underlying phase space, in which case $\mathcal{X}$ would represent the states of the system's "information bearing degrees of freedom" [88]. For simplicity, in this paper we ignore issues raised by coarse-graining and treat $\mathcal{X}$ as the microstates of our system.

We assume that the system is connected to a work reservoir and a heat bath at temperature T. The system evolves dynamically under the influence of a driving protocol, and we are interested in its dynamics over some fixed interval $t \in [0, t_f]$.

As mentioned in the Introduction, research in nonequilibrium statistical physics has defined thermodynamic quantities such as heat, work, and entropy production at the level of individual trajectories of a stochastically evolving process, so that ensemble averages of those measures over all trajectories obey the usual properties required by conventional statistical physics [18,19]. Adopting this approach, we define the *heat function* $Q(x)$ as the expected amount of heat transferred from our system to the heat bath during the interval $t \in [0, t_f]$, assuming that the system begins in initial state x. Following a standard setup in the literature [89–92], we assume that the joint Hamiltonian of the system and bath can be written as

$$H^t_X(x) + H_B(b) + H_{\text{int}}(x, b), \tag{10}$$

where H^t_X is the time-dependent Hamiltonian of the system, H_B is the bare Hamiltonian of the bath, and H_{int} is the interaction Hamiltonian (which is typically very small, reflecting weak-coupling). Regardless of the initial state of the system x, the bath is initially taken to be in a Boltzmann distribution $p_B(b) \propto e^{-H_B(b)/kT}$. Let $p'_{B|x}$ indicate the final distribution of the bath at $t = t_f$, given that the system began in initial state x. The heat function is then given by the increase of the expected energy of the bath [89,90],

$$Q(x) = \langle H_B \rangle_{p'_{B|x}} - \langle H_B \rangle_{p_B}. \tag{11}$$

The expectation of $Q(x)$ under any initial distribution p_X then gives the overall expected amount of generated heat averaged across all trajectories, assuming that initial system-bath states are sampled from $p_X(x)p_B(b)$. This setup can be used to model infinite-sized idealized heat baths (infinite heat capacity, fast equilibration, etc.) by taking appropriate limits [89–92].

A central quantity of interest in statistical physics is the (irreversible) *entropy production* (EP), which reflects the

overall increase of entropy in the system and the coupled environment. For a given physical process, let p_X be an initial-state distribution at time $t = 0$ and let p_Y be the corresponding final-state distribution at $t = t_f$. Then, the expected EP is

$$\Sigma(p_X) = S(p_Y) - S(p_X) + \langle Q \rangle_{p_X}/kT, \tag{12}$$

where $S(\cdot)$ indicates the Shannon entropy.[4] By the second law of thermodynamics, $\Sigma(p_X)$ is nonnegative for any physically allowed heat function Q and every initial distribution p_X [90]. A physical process is said to be *thermodynamically reversible* if it achieves zero EP.

We say that a physical process is a *realization* of some partial function $f : \mathcal{X} \to \mathcal{X}$ if the conditional probability of the system's ending state given the starting state obeys

$$p_{Y|X}(y|x) = \delta(f(x), y) \quad \forall x \in \text{dom}\, f. \tag{13}$$

The behavior of a realization of f on initial states $x \notin \text{dom}\, f$ can be arbitrary, as it is not constrained by Eq. (13).

The following technical result links the logical properties of a partial function f with the heat function of any realization of that f. This result will be central to our analysis, as it will allows us to establish thermodynamic constraints on processes that realize TMs.

Proposition 1. Given a countable set $\mathcal{X}$, let $f : \mathcal{X} \to \mathcal{X}$ and $G : \mathcal{X} \to \mathbb{R}$ be two partial functions with the same domain of definition. The following are equivalent:

(1) For all p_X with $\text{supp}\, p_X \subseteq \text{dom}\, f$,

$$\langle G \rangle_{p_X} + S[p_{f(X)}] - S(p_X) \geqslant 0. \tag{14}$$

(2) For all $y \in \text{img}\, f$,

$$\sum_{x: f(x)=y} e^{-G(x)} \leqslant 1. \tag{15}$$

(3) There exists a realization of f coupled to a heat bath at temperature T, whose heat function Q obeys

$$Q(x)/kT = G(x) \quad \forall\, x \in \text{dom}\, f. \tag{16}$$

This proposition is proved in Appendix C. The proof exploits a useful decomposition of EP into a sum of a conditional Kullback-Leibler divergence term and a nonnegative expectation term, which is derived in Appendix B.

We note two things about Proposition 1.

First, the remainder of the inequality in Eq. (15) determines the EP incurred by a realization of f. In particular, as we show in Appendix C, if that inequality is tight for all $y \in \text{img}\, f$, then the inequality in Eq. (14) is also tight for some initial distributions p_X. In this case, the realization of f referenced in Eq. (16) is thermodynamically reversible for those initial p_X.

Second, it is straightforward to generalize the setup described in this section to consider a system connected to multiple thermodynamic reservoirs instead of a single heat bath [17]. In the general case, Proposition 1 still holds if the left-hand side of Eq. (16) is interpreted as the amount of entropy increase in all coupled thermodynamic reservoirs, given that the process begins in initial state x. Equation (16) is a special case of this general formulation, since releasing $Q(x)$ of heat to a bath at temperature T increases the bath's entropy by $Q(x)/kT$.

B. Realizations of a TM

We briefly describe how a physical process can realize a TM M. Without loss of generality, we assume that the countable state space of the physical system $\mathcal{X}$ can be represented by a set of binary strings, so $\mathcal{X} \subseteq \{0, 1\}^*$.

As described in Sec. II A and Appendix A, the computation performed by a TM can be formalized as a partial function $\phi_M : \{0, 1\}^* \to \{0, 1\}^*$. We say that a physical process is a realization of a TM M if it realizes the partial function ϕ_M in the sense of Eq. (13) and Proposition 1. Note that this is only possible when $\text{dom}\, \phi_M \cup \text{img}\, \phi_M \subseteq \mathcal{X}$. Note also that there may be physical states $x \in \mathcal{X}$ that do not belong to $\text{dom}\, \phi_M$. When the system is initialized with such states at $t = 0$, it will undergo some well-defined dynamical evolution. However, its behavior for such initial states is not constrained by the fact that the system is a realization of the TM and can be arbitrary (in general, the dynamic and thermodynamic properties for such initial x are not our focus). The mapping between a TM and a physical system is illustrated in Fig. 2.

Many TMs, including all UTMs, can have arbitrarily long programs (i.e., unbounded input length) and can take an arbitrary number of steps before halting on any particular input (i.e., unbounded runtime). For such TMs, our formulation appears to assume a physical system that can store a tape of unbounded size, and which can complete an unbounded number of computational steps in a finite time interval $[0, t_f]$, which is not realistic from a physical point of view. In such cases, one can imagine a sequence of realizations, each of which involves manipulating a finite (but growing) tape over a finite (but growing) number of computational steps. Our analysis and results then apply to limit of this sequence, in which the tape size and runtime can be arbitrarily large.

In the following sections, we apply Proposition 1 with $f = \phi_M$ to establish constraints on the heat function Q of any realization of M. We emphasize that in general these constraints do not fully determine the heat function of any realization of M: There can be many different realizations of any given TM M, each with different heat functions and therefore with different thermodynamic properties (see also Ref. [34]). In the next sections, we analyze the thermodynamics of two particular realizations of a given TM, which we call the *coin-flipping realization* and the *dominating realization*. We work "backwards" for each one, first specifying its heat function and then using Proposition 1 to establish that there is in fact a realization with that heat function, and then analyzing the properties of that heat function.

Before proceeding, we discuss an important issue concerning the computability properties of realizations of TMs. We say that a realization of a TM M with heat function Q is a *computable realization* if the function $Q(x)/kT$ is

[4] For countably infinite state spaces (e.g., the state spaces of UTMs), the Shannon entropy of both the initial and final distribution can be infinite, making the expression in Eq. (12) ill defined. In such cases, a finite EP can often be defined by writing Eq. (12) as a limit $\Sigma(p_X) = \lim_{i\to\infty} \Sigma(p_i)$, where each p_i has finite support and $\lim_{i\to\infty} p_i = p_X$.

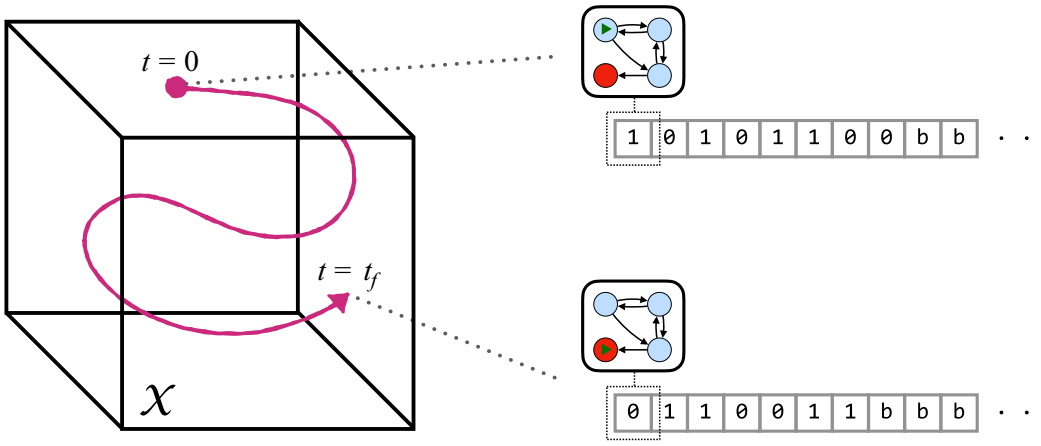

FIG. 2. A realization of a TM is a physical process over a countable state space $\mathcal{X} \subseteq \{0,1\}^*$, which maps initial states to final states according to the input-output function of the TM. As a hypothetical example, consider a process that evolves to the final state `0110011` at $t = t_f$ when started on initial state `10101100` at $t = 0$, as might correspond to a computation performed by the TM (see also Fig. 1).

computable [i.e., if there exists a TM that can take as input any $x \in \text{dom}\,\phi_M$ and output the value of $Q(x)/kT$ to arbitrary precision]. Some of our results below will rely on particular properties of computable realizations. At the same time, some of the realizations we construct and analyze below will not be computable. Whether such noncomputable realizations can *actually* be constructed in the real world depends on the status of the physical Church-Turing thesis. To see why, imagine that one could construct a noncomputable realization of a TM; for example, it might have $Q(x)/kT = K(x)$, where $K(x)$ is the (noncomputable) Kolmogorov complexity function. In that case, one could run the realization on various inputs x, use a calorimeter to measure the generated heat in units of kT [i.e., measure $Q(x)/kT$], and then arrive at the value of $K(x)$. The above procedure would use a physical process to evaluate a noncomputable function, thereby violating the physical Church-Turing thesis.

In this paper, we do not take a position on the validity of the physical Church-Turing thesis. Rather, we will explicitly discuss relevant (non)computability properties of our realizations, as well as how our noncomputable realization can be interpreted in light of the physical Church-Turing thesis. It is important to emphasize, however, that even our noncomputable realizations are consistent with the laws of thermodynamics, and are well-defined in terms of a sequence of time-varying Hamiltonians and stochastic dynamics (see the construction in the proof of Proposition 1, Appendix C). Their noncomputability arises from the fact that our construction uses various idealizations, such as the ability to apply arbitrary Hamiltonians to the system, which are standard in theoretical statistical physics but which disregard possible *computational constraints* on the set of achievable processes. For example, our construction disregards the fact that, if the physical Church-Turing thesis holds, then it should be impossible to apply noncomputable Hamiltonians to the system, such as $H(x) = K(x)$.

IV. COIN-FLIPPING REALIZATION

We first consider a realization of a TM M that achieves zero EP (i.e., is thermodynamically reversible) when run on input programs randomly sampled from a particular input distribution.

To begin, consider the following *coin-flipping* distribution over programs, which plays an important role in AIT:

$$m_X(x) := \begin{cases} 2^{-\ell(x)} & \text{if } x \in \text{dom}\,\phi_M \\ 0 & \text{otherwise} \end{cases}. \tag{17}$$

Note that m_X sums to a value less than 1 [42], and therefore m_X is a nonnormalized probability distribution. Nonetheless, we refer to it as a "distribution," following the convention in the AIT literature.

To understand m_X more concretely, imagine that the initial state of the TM's tape is set to a sample of an infinitely long sequence of independent and uniformly distributed bits. Then $m_X(x)$ is proportional to the probability that M eventually halts after reading the bit string x from the tape.[5] Under this hypothetical initialization procedure, the TM will halt on output y with probability

$$m_Y(y) = \sum_{x:\phi_M(x)=y} 2^{-\ell(x)}. \tag{18}$$

This output distribution is biased toward strings that can be generated by short input programs. Note that, like m_X, this output distribution is not normalized.

We now consider the thermodynamic cost of running a TM on the coin-flipping distribution. We first define a normalized version of the coin-flipping distribution,

$$p_X^{\text{coin}}(x) := m_X(x)/\Omega_M, \tag{19}$$

where $\Omega_M := \sum_{x \in \text{dom}\,\phi_M} 2^{-\ell(x)} \leqslant 1$ is a normalization constant (which in AIT is called the "halting probability").

[5]For clarity, we omit various technicalities regarding the random process that motivates the coin-flipping distribution. To be precise, this process should be defined in terms of a multitape machine, in which one of the tapes is a one-way read-only "input tape" (see Appendix A). Then $m_X(x)$ is the probability that the multitape machine halts after reading the string x from the input tape, assuming the input tape is initialized with an infinitely long random bit string.

$p_X^{\text{coin}}(x)$ is the probability that a TM halts after running input program x, conditioned on the TM halting on *some* input program, given the random initial tape described above. We also define a normalized version of the output distribution,

$$p_Y^{\text{coin}}[\phi_M(x)] := m_Y[\phi_M(x)]/\Omega_M. \tag{20}$$

Now consider the associated function

$$G(x) = -\ln p_X^{\text{coin}}(x) + \ln p_Y^{\text{coin}}[\phi_M(x)]. \tag{21}$$

It can be verified that this function satisfies condition 2 of Proposition 1. Thus, there is at least one realization of M, which we call the *coin-flipping realization*, whose heat function obeys

$$Q_{\text{coin}}(x) = kT\left\{-\ln p_X^{\text{coin}}(x) + \ln p_Y^{\text{coin}}[\phi_M(x)]\right\}. \tag{22}$$

By plugging Q_{coin} into Eq. (12), we can verify that this realization achieves $\Sigma(p_X^{\text{coin}}) = 0$, meaning that it is thermodynamically reversible when run on input distribution p_X^{coin}.

Equation (22) can be further simplified by using the definitions of p_X^{coin} and p_Y^{coin}:

$$Q_{\text{coin}}(x) = kT \ln 2\,\{\ell(x) + \log_2 m_Y[\phi_M(x)]\}. \tag{23}$$

This establishes the claim in the Introduction that the heat generated under the coin-flipping realization on input x is proportional to the length of x, minus a "correction term" $-\log_2 m_Y[\phi_M(x)]$. This correction term is always positive, since $m_Y(y) \leqslant 1$ for all y. Moreover, it reflects the logical irreversibility of the partial function ϕ_M on input x: It achieves its minimal value of $-\log_2 \Omega$ when ϕ_M maps all inputs to a single output, and its maximal value of $\ell(x)$ when ϕ_M is logically reversible on input x [i.e., when x is the only input that produces output $\phi_M(x)$]. In the latter (logically reversible) case, $Q_{\text{coin}}(x) = 0$ for all x.

Equation (23) implies that if one wishes to produce some desired output $y \in \text{img}\,\phi_M$ while minimizing heat generation, then one should choose the shortest input x such that $\phi_M(x) = y$. Loosely speaking, the "less efficient" one is in choosing what program to use to compute y, the greater the heat that is expended in that computation. Note that this relationship between shorter programs and less heat generation is not a universal feature of all realizations of TMs. It holds for the coin-flipping realization because this realization is explicitly designed to be thermodynamically reversible for the coin-flipping input distribution, which has a "built-in bias" for shorter input strings.

An important special case is when the TM of interest is a universal TM. For any UTM U, the output distribution in Eq. (18) is called the *universal distribution* in AIT. The universal distribution possesses many important mathematical properties and is one of the cornerstones of AIT [42,71,97,104,105], and has attracted attention in artificial intelligence [93–98], foundations of physics [99,100], and statistical physics [77,101–103]. In particular, Levin's coding theorem [42] relates the universal distribution to Kolmogorov complexity,

$$-\log_2 m_Y(y) = K(y) + O(1). \tag{24}$$

This implies that for a UTM, the "correction term" mentioned above is equal to the Kolmogorov complexity of the output, up to an additive constant.

Plugging Eq. (24) into Eq. (23) lets us write the heat function of the coin-flipping realization of a UTM as

$$Q_{\text{coin}}(x) = kT \ln 2\,\{\ell(x) - K[\phi_U(x)]\} + O(1). \tag{25}$$

So for a coin-flipping realization of a UTM, the heat generated on input x reflects how much the length of x exceeds the shortest program which produces the same output as x.

These results allow us to calculate the thermodynamic complexity of any output string y using the coin-flipping realization of a UTM U, i.e., the minimal heat necessary to generate some desired output y:

$$\min_{x:\phi_U(x)=y} Q_{\text{coin}}(x) = O(1), \tag{26}$$

where we have used Eq. (25) and the fact that $\min_{x:\phi_U(x)=y} \ell(x) = K(y)$ by definition. Thus, for the coin-flipping realization, the minimal heat required by the UTM to compute y is bounded by a constant. As emphasized above, this is a fundamental difference between thermodynamic complexity of the coin-flipping realization and Kolmogorov complexity, which is unbounded as one varies over y.

However, in order to actually produce a desired output y on a UTM U while generating the minimal possible amount of heat, one needs to know the shortest program for that y. Unfortunately, the shortest program for a given output is not computable in general. In fact, we prove in Appendix D that there cannot exist a computable function that maps any desired output y to some corresponding input x such that both $\phi_U(x) = y$ and the heat is bounded by a constant, $Q_{\text{coin}}(x) = O(1)$.

We finish by considering the expected heat that would be generated by a realization of a UTM U if inputs were drawn from the distribution p_X^{coin}. To begin, rewrite Eq. (12) as

$$\langle Q\rangle_{p_X^{\text{coin}}} = kT\left[S\left(p_X^{\text{coin}}\right) - S\left(p_Y^{\text{coin}}\right) + \Sigma\left(p_X^{\text{coin}}\right)\right]. \tag{27}$$

In Appendix F, we show that the difference of entropies on the right-hand side of Eq. (27) is infinite. Since $\Sigma(p_X^{\text{coin}})$ is always nonnegative, any realization of U must, on average, expend an infinite amount of heat to run input programs sampled from p_X^{coin}. This applies to the coin-flipping distribution, for which $\Sigma(p_X^{\text{coin}}) = 0$, as well as any other realization. Note that $\ell(x) \geqslant Q_{\text{coin}}(x)/(kT \ln 2)$ [by Eq. (23) and the fact that $m_Y(y) \leqslant 1$ for all y] and that $\ell(x)$ is a lower bound on the number of steps that a prefix UTM needs to run program x (since it must take at least one step per read-in bit). Thus, the fact that programs sampled from the coin-flipping distribution have infinite expected heat generation also implies that they have an infinite expected length and take an infinite expected number of steps before halting.

We finish by emphasizing that EP and expected heat vary in different ways as one changes the initial distribution. For example, if we run the coin-flipping realization on input distribution p_X^{coin}, then EP is zero while expected heat is infinite. On the other hand, since expected heat is a linear function of the input distribution, minimal expected heat corresponds to a δ-function input distribution centered on the x that minimizes $Q_{\text{coin}}(x)$. However, some simple algebra shows that any such δ-function distribution incurs a strictly positive EP for any

UTM.[6] Thus, the distribution that minimizes expected heat cannot be the one that minimizes EP.

V. DOMINATING REALIZATION

A. Minimal possible heat function

We now consider a realization of a TM whose heat function is smaller, up to an additive constant, than the function of any computable realization.

To begin, given any (universal or nonuniversal) TM M, consider the associated function $G(x) = \ln 2\, K[x|\phi_M(x)]$. Note that this conditional Kolmogorov complexity can be defined in terms of any desired UTM, with no *a priori* relation to M. In Appendix E, we show that this function G satisfies condition 2 in Proposition 1. Therefore, there must be at least one realization of M, which we call the *dominating realization*, whose heat function obeys

$$Q_{\rm dom}(x) = kT \ln 2\, K[x|\phi_M(x)]. \tag{28}$$

Intuitively speaking, the inputs x that generate a large amount of heat under the dominating realization of a TM M are long and incompressible, even when given knowledge of their associated outputs $\phi_M(x)$. An example of such an input is a program x that instructs M to read through a long and incompressible bit string and then output nothing, so that $\phi_M(x)$ is an empty string (this example is analyzed in more depth below, in Sec. VI). In contrast, the inputs x that generate little heat under the dominating realization are those in which the output provides a large amount of information about the associated input program. For instance, if M is universal, then a program x that consists of the instruction "`print 'y'`" (represented in some appropriate binary encoding) generates little heat, since $K(\text{"}\texttt{print 'y'}\text{"}|y) = O(1)$ for any y. More generally, if ϕ_M is logically reversible over its domain, then $K[x|\phi_M(x)] = O(1)$ for *all* x in that domain, because one can always reconstruct the input x from the output $\phi_M(x)$ by applying ϕ_M^{-1}. Thus, in the logically reversible case, the heat generated by the dominating realization on any input x is bounded by a constant that does not depend on x.

Now consider any alternative computable realization of M that is coupled to a heat bath at temperature T, whose heat function we indicate by Q. The assumption of computability means that the function $Q(x)/kT$ is computable [i.e., there is some TM that, for any desired x, can approximate the value of $Q(x)$ in units of kT to arbitrary precision].

As we prove in Appendix E, the heat function of this alternative realization must obey the following inequality:

$$Q(x) \geqslant Q_{\rm dom}(x) - kT \ln 2\, K(Q/kT) + K(\phi_M) + O(1), \tag{29}$$

where $K(Q/kT)$ is the Kolmogorov complexity of the heat function Q in units of kT, $K(\phi_M)$ is the Kolmogorov complexity of the partial function computed by M, and $O(1)$ represents equality up to an additive constant (that does not depend on x, Q, or M).

Since neither $K(Q/kT)$ nor $K(\phi_M)$ depends on the input x, Eq. (29) implies $Q(x) \geqslant Q_{\rm dom}(x) + \kappa$ for some constant κ that is independent of x. Note though that κ can depend on ϕ_M (the partial function being computed) and the alternative realization Q, and note also that in principle this constant may be arbitrarily large and negative. This means that for any fixed input x, there may be computable realizations that result in far less heat when run on x than does the dominating realization. However, this can only occur if ϕ_M has high complexity [large value of $K(\phi_M)$], or if the heat function has high complexity, as reflected by a large value of $K(Q/kT)$. This shows that any computable realization must face a fundamental trade-off between three different factors: the "lost" algorithmic information about the input in the output, the complexity of the input-output map being realized, and the complexity of the heat function. We explore this trade-off using an example of erasing a long string in Sec. VI.

When the TM under question is universal, then it is guaranteed that there exists some program that can generate any desired output y. This permits us to analyze the thermodynamic complexity of the dominating realization. It turns out that, as for the coin-flipping realization, this amount is bounded by a constant:

$$\min_{x:\phi_U(x)=y} Q_{\rm dom}(x) = O(1). \tag{30}$$

This minimum is achieved by programs of the form $x =$ "`print 'y'`", since these programs achieve $K[x|\phi_U(x)] = O(1)$. Equation (30) also holds if the TM is not a UTM, as long as for each each output y, there is some x that obeys $\phi_M(x) = y$ and $K[x|\phi_M(x)] = O(1)$ (e.g., if ϕ_M is logically reversible).

Finally, we consider the expected heat that would be generated by running the dominating realization of a UTM U, assuming that inputs are sampled randomly from some input distribution. To parallel the analysis of the coin-flipping realization, we consider the input distribution which results in minimal EP for the dominating realization, which we call p_X^*. In Appendix F, we prove that the expected heat generated by the dominating realization on the input distribution p_X^* is infinite. It is interesting to note that $\ell(x) \geqslant Q_{\rm dom}(x)/(kT \ln 2) + O(1)$ and, as we mentioned above, $\ell(x)$ is a lower bound on the number of steps that a UTM needs to run program x.[7] Thus, the fact that programs sampled from p_X^* have infinite expected heat generation also implies that they have an infinite expected length and an infinite expected runtime. Note that the dominating realization of a UTM will in general incur a strictly positive amount of EP, even when run on the optimal input distribution p_X^* (see Appendix G for details).

B. Practical implications of the dominating realization

Our analysis of the dominating realization uses several abstract computer science concepts, such as the computability of

[6] Given a UTM and any string y, there are many inputs x that result in $\phi_U(x) = y$. This means that $p_Y^{\rm coin}[\phi_U(x)] > p_X^{\rm coin}(x)$ for any x, so $Q_{\rm coin}(x) > 0$ by Eq. (22). Thus, for any δ-function distribution δ_x, $\Sigma(\delta_x) = S[\delta_{\phi_U(x)}] - S(\delta_x) + Q(x) = Q(x) > 0$, where we have used $S[\delta_{\phi_U(x)}] = S(\delta_x) = 0$.

[7] We have the inequalities $K(x|y) \leqslant K(x) + O(1) \leqslant \ell(x) + O(1)$. The first comes from subadditivity of Kolmogorov complexity [42], while the second comes from Lemma 5 in Appendix H.

the heat function and its Kolmogorov complexity. It is worth making some comments about the real-world significance of such concepts for the thermodynamics of physical systems.

First, the computability properties of the heat function are entirely separate from the computability properties of the logical map ϕ_M realized by a physical process. In particular, the heat function can be uncomputable even though ϕ_M is computable by definition (since ϕ_M is the partial function implemented by a TM). On the other hand, common interpretations of the physical Church-Turing thesis imply that the heat function of any *actually* constructable real-world physical process must be computable. This implies that if the physical Church-Turing holds, then the dominating realization generates less heat, up to an additive constant, than any realization that can actually be constructed in the real world.

At the same time, while the dominating realization is better than any computable realization, it is important to note that it itself is not computable. This is because the conditional Kolmogorov complexity is not a computable function, i.e., there is no TM that can take as input two strings x and y and output the value of $K(x|y)$. However, this does not necessarily imply that the dominating realization is irrelevant from a practical point of view. This is because $K(x|y)$ is an *upper-semicomputable* function, meaning that it is possible to compute an improving sequence of upper bounds that converges on $K(x|y)$. Formally, there is a computable function f such that $f(x, y, n) \geqslant f(x, y, n+1)$ and $\lim_{n\to\infty} f(x, y, n) = K(x|y)$.[8]

The upper-semicomputability of Q_{dom} allows one to approach the performance of Q_{dom} by constructing a sequence $i = 1, 2, \ldots$, of realizations of ϕ_M, each with a computable heat function Q_i, such that Q_i converge from above on Q_{dom}. Each subsequent realization in this sequence is guaranteed to be better (generate less heat) on every input than the previous. Moreover, because the heat functions converge on Q_{dom}, by advancing far enough in this sequence one can run any input x with only $Q_{\text{dom}} + \epsilon$ heat for any $\epsilon > 0$. An important subtlety, however, is that one cannot compute how far into the sequence to advance so as to be within ϵ of Q_{dom} (if one could compute this, then Q_{dom} would be computable, and not just upper-semicomputable).

Finally, while we showed that Q_{dom} is better than any computable realization in terms of heat generation, we also mentioned that it itself is only upper-semicomputable, not computable. One might ask whether there is some other upper-semicomputable realization (i.e., one whose heat function can be approached by above) which is even better than Q_{dom}. It is known that this is not the case: The optimality result of Eq. (29) holds not only for any computable Q but more generally for any upper-semicomputable Q.

C. Comparison of coin-flipping and dominating realizations

We finish our discussion of the dominating realization by briefly comparing it to the coin-flipping realization.

First, for both dominating and coin-flipping realizations, the minimal heat necessary to generate a given output y on a UTM U, which we call the thermodynamic complexity of the realization, is bounded by a constant that does not depend on y. There is no *a priori* relationship between those two constants, and in principle it is possible that, for all y, the thermodynamic complexity is larger under the dominating realization than the coin-flipping realization or vice versa. In general, the constants will depend on the realized UTM U, as well as the UTM used to define the conditional Kolmogorov complexity in Eq. (28) (which does not have to be the same as U).

Second, to achieve bounded heat production for output y under the coin-flipping realization, one must know the shortest program for producing y, which is uncomputable. In contrast, to achieve bounded heat production for output y under the dominating realization, it is sufficient to choose an input of the form "`print 'y'`".

Third, for both realizations, there is an infinite amount of expected heat generated, assuming that inputs are sampled from the EP-minimizing distribution.

Fourth, the coin-flipping realization is (by design) thermodynamically reversible for input distribution p_X^{coin}. The dominating realization, on the other hand, is not thermodynamically reversible for any input distribution (see Appendix G).

Finally, neither the coin-flipping nor the dominating realization of a UTM has a computable heat function. In fact, the heat function of the coin-flipping realization is not even upper-semicomputable.[9] This means that our results concerning the superiority of the dominating realization do not apply when comparing to the coin-flipping realization, and in particular it is not necessarily the case that $Q_{\text{coin}}(x) \geqslant Q_{\text{dom}}(x) + O(1)$. Nonetheless, it turns out that for any UTM U, the additional heat incurred by the dominating realization on input x, beyond that incurred by the coin-flipping realization, is bound by a logarithmic term in the complexity of the output,

$$Q_{\text{coin}}(x) \geqslant Q_{\text{dom}}(x) - O\{\log K[\phi_U(x)]\}. \tag{31}$$

(See Appendix H for proof.) Such logarithmic correction terms are considered inconsequential in some previous analyses of the thermodynamics of TMs [5,82].

VI. HEAT VS. COMPLEXITY TRADE-OFF

Our analysis of the dominating realization uncovered a trade-off between heat and complexity faced by any computable physical process. In this section, we illustrate this trade-off by analyzing the thermodynamics of erasing a long bit string.

As before, consider a physical system with a countable state space, which undergoes driving while coupled to a heat bath at temperature T. For notational simplicity, in this section we choose units so that $kT = 1$. Assume that the process realizes some deterministic and computable map from initial

[8]This function can be computed by a TM that runs multiple programs in parallel, while keeping track of the shortest program which has halted on input y with output x.

[9]Recall that $Q_{\text{coin}}(x) = \ell(x) + \log m_Y[U(x)]$. $\ell(\cdot)$ is computable while $-\log m_Y(\cdot)$ is upper-semicomputable [[42], Thm. 4.3.3]. This implies that Q_{coin} is "lower-semicomputable," meaning it can be approximated by an improving sequence of computable lower bounds.

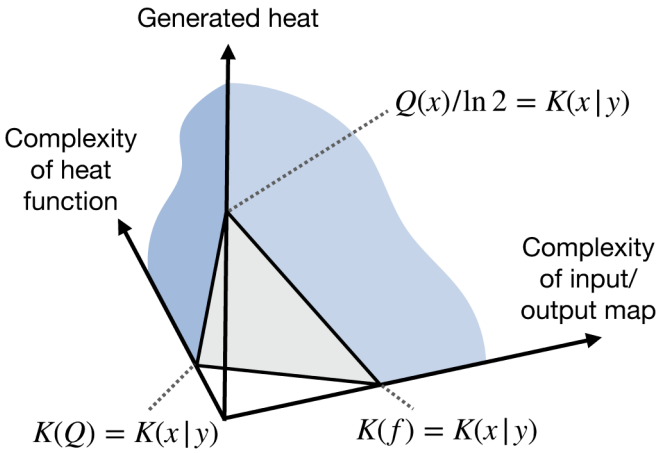

FIG. 3. Any computable process that realizes a deterministic input-output map f faces a fundamental cost of $K(x|y)$ for mapping input x to output $y = f(x)$. This cost can be paid through some combination of three different strategies: generating a large amount of heat, having a high complexity heat function, or having a high complexity input-output map f. This trade-off is illustrated on three axes, with blue indicating the feasible region.

to final states, which we indicate generically as $f : \{0, 1\}^* \to \{0, 1\}^*$. Now imagine that one observes a single realization of this physical process, in which initial state x is mapped to final state $y = f(x)$.

Since this is a computable realization of f, it must obey the dominating realization bound of Eq. (29). Plugging Eq. (28) into that inequality and rearranging gives

$$Q(x)/\ln 2 + K(Q) + K(f) \geqslant K(x|y) + O(1), \tag{32}$$

where we have used the assumption that $kT = 1$. This shows that there is a fundamental cost of $K(x|y)$ that is incurred by any computable realization that maps input x to output y. This fundamental cost can be paid either by generating a lot of heat [large $Q(x)/\ln 2$], by having a high complexity heat function [large $K(Q)$] or by realizing a high-complexity input-output function [large $K(f)$]. This trade-off is illustrated in Fig. 3.

We demonstrate this trade-off using an example of a process that erases a long binary string. In this example, x is a long string consisting of n binary digits, while the final state y is a string of n 0s, which we write '00...00'. Assuming x is incompressible (which is true for the vast majority of all strings), the fundamental cost of mapping $x \to y$ is given by $K(x|y) = K(x) \approx \ell(x)$ up to logarithmic factors [42]. Different processes can pay this fundamental cost in different ways, thereby satisfying Eq. (32):

(1) A process can generate a lot of heat. For example, in order to erase string x, the process can run an erasure map:

$$f(x') := \text{'00...00'} \quad \forall x', \tag{33}$$

while using the dominating implementation. In this case, $Q(x)/\ln 2 = K(x|y)$ by Eq. (28).

(2) A process can have a high-complexity heat function, so that $K(Q) \geqslant \ell(x)$. For example, one can tweak the dominating realization of the erasure map, so that the heat values for input x and the input consisting of all 0s are swapped:

$$Q(x') := \begin{cases} Q_{\text{dom}}(x') & \text{if } x' \notin \{x, \text{'00...00'}\} \\ Q_{\text{dom}}(\text{'00...00'}) & \text{if } x' = x \\ Q_{\text{dom}}(x) & \text{if } x' = \text{'00...00'} \end{cases}.$$

One can verify that since Q_{dom} satisfies condition 2 in Proposition 1, so does this Q. Moreover, this realization generates a small amount of heat when erasing x,

$$\begin{aligned} Q(x) &= Q_{\text{dom}}(\text{'00...00'}) \\ &= K(\text{'00...00'}|\text{'00...00'}) \approx 0. \end{aligned}$$

Note, however, that the long input string x is now "hard-coded" into the definition of the heat function Q, leading to a large value of $K(Q)$.

(3) A process can realize a high-complexity input-output map f, so that $K(f) \geqslant K(x|y)$. This strategy could be used, for example, by a process which implements the following logically reversible map:

$$f(x') := \begin{cases} x' & \text{if } x' \notin \{x, \text{'00...00'}\} \\ \text{'00...00'} & \text{if } x' = x \\ x & \text{if } x' = \text{'00...00'} \end{cases}.$$

Since logically reversible function can be carried out without generating heat, it is possible to implement this f while achieving $Q(x') = 0$ for all x'. In this case, not only does erasing x not generate any heat, $Q(x) = 0$, but also the heat function has low complexity, $K(Q) \approx 0$. However, the long input string x is now "hard-coded" into the definition of the input-output map f, leading to a large value of $K(f)$.

We finish by noting that in a series of papers by Zurek and others [5,6,80–86], it was argued that the conditional Kolmogorov complexity $K(x|y)$ is "the minimal thermodynamic cost" of computing some output y from input x. However, most of these early papers were written before the development of modern nonequilibrium statistical physics. As a result, the arguments in those papers are rather informal, which in turn makes it difficult to translate them in a fully rigorous manner into modern nonequilibrium statistical physics. (See Sec. 14.4 in Ref. [34] for one possible translation.) To give one example of these difficulties, those earlier analyses quantified the "thermodynamic cost" in terms of the number of physical bits (binary degrees of freedom) that are erased during that computation, independent of the initial probability distributions over those binary degrees of freedom. However, we now know that minimal heat generation is given by changes in Shannon entropy, i.e., in terms of statistical bits rather than physical bits. Relatedly, these papers led to some proposals that the foundations of statistical physics be changed, so that thermodynamic entropy is identified not only with Shannon entropy but also a Kolmogorov complexity term [6,42].

In contrast, our analysis is grounded in modern nonequilibrium physics and does not involve any foundational modifications to the definition of thermodynamic entropy. Moreover, it covers some issues not considered in earlier analyses. In particular, we show that the lower bound of $K(x|y)$ is a cost that in general applies only to computable realizations (i.e., ones with a computable heat function), not for all possible realizations, as implied in the earlier papers. The significance

of this restriction depends on the legitimacy of the physical Church-Turing thesis. Finally, we also demonstrate different ways in which one can pay the fundamental cost $K(x|y)$: by generating heat, by having a large Kolmogorov complexity of the heat function $K(Q)$, or by having a large Kolmogorov complexity of the input-output map, $K(f)$.

VII. DISCUSSION

In this paper we combine AIT and nonequilibrium statistical physics to analyze the thermodynamics of TMs. We consider a physical process that realizes a deterministic input-output function, representing the computation performed by some TM. We derive numerous results concerning two different realizations of TM: a *coin-flipping realization*, which is designed to be thermodynamically reversible when fed with random input bits, and a *dominating realization*, which is designed to generate less heat than any computable realization.

Using our analysis of the dominating realization, we uncover a fundamental trade-off, faced by any computable realization of a deterministic input-output map, between heat generation, the Kolmogorov complexity of the heat function, and the Kolmogorov complexity of the input-output map. An interesting topic for future research is how the Kolmogorov complexity of the heat function and the input-output map relates to the "physical complexity" of the driving process, as commonly understood in physics (e.g., whether the Hamiltonians must have many-body interactions, etc.).

For simplicity, in this paper we represented a TM M as a physical system whose dynamics carries out the partial function $\phi_M : \{0,1\}^* \to \{0,1\}^*$ during some finite time interval $[0, t_f]$. This representation allowed us to abstract away many implementation details of the realization, such as the fact that a TM consists of a separate tape, head, and pointer variables and that a TM operates in a sequence of discrete steps. Essentially, this representation does not distinguish whether the physical process operates via the same sequence of steps as a TM or simply implements a "lookup table" that maps outputs to inputs.

While this representation simplifies our analysis, it provides no guidance on how to actually construct a physical process that realizes a TM in the laboratory, and it leaves implicit some important issues. Alternatively, one could represent a realization of a TM in a more conventional and "mechanistic" way, as a dynamical system over the state of the TM's tape, pointer, and head, which evolves iteratively according to the update function of the TM until the head reaches the halt state. In contrast to the representation we adopted, this kind of mechanistic representation could easily be physically constructed and would correspond more closely to the step-by-step operation of real-world physical computers. Moreover, this kind of mechanistic representation could be used to analyze the thermodynamic costs of TMs in a more realistic manner. For example, it could be used to analyze how the heat and EP incurred by the TM depends on the number of steps taken. As another example, it could be used to impose constraints on how the degrees of freedom of the head, tape, and pointer can be coupled together (e.g., via interaction terms of applied Hamiltonians). One might postulate, for instance, that the head of the TM can only interact with tape locations that are located near the pointer. These kinds of constraint will generally increase the heat and EP incurred by each step of the TM [34,106]. These complications concerning the thermodynamics of more mechanistic representations of TMs are absent from the analysis in this paper and are topics of future research.

ACKNOWLEDGMENTS

We thank Josh Grochow, Cris Moore, Daniel Polani, Simon DeDeo, Damian Sowinski, Eric Libby, Sankaran Ramakrishnan, Bernat Corominas-Murtra, and Brendan D. Tracey for many stimulating discussions and the Santa Fe Institute for helping to support this research. This paper was made possible through the support of Grant No. TWCF0079/AB47 from the Templeton World Charity Foundation, Grant No. CHE-1648973 from the U.S. National Science Foundation, Grant No. FQXi-RFP-1622 from the Foundational Questions Institute (FQXi), and Grant No. FQXi-RFP-IPW-1912 from the Foundational Questions Institute (FQXi) and Fetzer Franklin Fund, a donor advised fund of Silicon Valley Community Foundation. The opinions expressed in this paper are those of the author and do not necessarily reflect the view of Templeton World Charity Foundation.

APPENDIX A: MODELS OF SINGLE-TAPE TMs

In this Appendix we present a formal definition of a single-tape TM.

In Sec. II A, we define the state of a TM as being composed of a tape state $s \in A^\infty$, a pointer state $v \in \mathbb{N}$, and head state $h \in H$. Here A is a finite alphabet of tape symbols which includes a special "blank" symbol, while H is a finite set of head states which includes a special "start" head state and a special "halt" head state. Any particular value of the triple (s, v, h) is called an *instantaneous description* (ID) of the TM. The dynamics of a particular TM is given by iteratively applying an *update function* f to the ID,

$$f : (s, v, h) \mapsto (s', v', h'). \tag{A1}$$

Following standard definitions, we assume that $f(s, v, h)$ only depends on $(s(v), h)$, i.e., the next ID of the TM can only depend on the current state of the head and the current contents of the tape s at position v. We also assume that the new value of the pointer v' does not differ by more than 1 from v and that the tape state s' be identical to the tape state s at all positions, except possibly position v. By iteratively applying f, the head moves back and forth along the tape, while both changing its state as well as reading and writing symbols onto the tape at its current position.

At the beginning of a computation, the state of the TM must be a *valid initial ID*, meaning that the head h is in the start state, the pointer is set to $v = 1$, and the tape s consists of finite string of nonblank symbols, followed by an infinite sequence of blank symbols. The TM then visits a sequence of IDs by iteratively applying the update function f. The TM stops if the head ever reaches the halt state (i.e., any ID where the head in the halt state is a fixed point of f). In general, there can be valid initial IDs for which the TM never halts.

For simplicity, we assume that 0 and 1 are elements of the alphabet A and that the nonblank finite string at the beginning of the initial tape state is some $x \in \{0,1\}^*$. In addition, we assume that if the head of the TM reaches a halt state after starting from some valid initial ID, then at that time the pointer is set to 1 and the final tape state begins with some $y \in \{0,1\}^*$, followed by blank symbols. In that case, we refer to the string $x \in \{0,1\}^*$ as the *input* or *program* for the TM and the corresponding string $y \in \{0,1\}^*$ as the *output* of the TM for program x.

Given these assumption, we can represent the overall computation performed by a TM M as a partial function $\phi_M : \{0,1\}^* \to \{0,1\}^*$. Here $\phi_M(x) = y$ indicates that when the TM is initialized with its tape containing x followed by an infinite sequence of blank symbols, then it will halt with its tape containing y followed by an infinite sequence of blank symbols. If the TM does not halt for some particular initial tape state x, then the value of $\phi_M(x)$ is undefined (for this reason, in general ϕ_M is a partial function). When we talk about a realization of a TM M in the main text, we refer to a physical process over a countable state space, whose dynamics from initial states to final states can be mapped onto the partial function ϕ_M implemented by some TM M.

As we mention in the main text, we assume that any TM under consideration is a prefix TM, meaning that it has a prefix-free halting set. Prefix TMs are typically TMs with multiple tapes, where one of the tapes is a read-only input tape that is read left to right [42]. If this kind of multitape machine halts after reading some string x from the input tape, then it means that the machine did not halt after reading some string x' on the input tape which is a strict prefix of x (otherwise, it would never get to read-in all of x), thereby guaranteeing the prefix property. For simplicity, however, in this paper we assume that the prefix TM is single-tape. This can be done without loss of generality, as it is always possible to transform a prefix TM with multiple tapes into an equivalent single-tape prefix TM, using any of the conventional techniques for transforming between multitape and single-tape TMs (see Ref. [[87], Thm. 2.1] and Ref. [40] for details). Note that these techniques may involve adding additional symbols to the tape alphabet A, which may be used at intermediate steps of the computation.

APPENDIX B: DECOMPOSITION OF ENTROPY PRODUCTION

In this Appendix, we derive a useful decomposition of the EP incurred by a realization of a deterministic input-output function. We also relate this decomposition to our previous work, which analyzed the dependence of EP on the initial distribution of a process [34,106,107].

Consider some physical process that realizes the function $f : \mathcal{X} \to \mathcal{X}$ in the sense of Eq. (13). Then, the conditional distribution of an initial state $x \in \mathrm{dom}\, f$ given final state $f(x)$ can be written as

$$p_{X|f(X)}[x|f(x)] := \frac{p_X(x)}{\sum_{x':f(x')=f(x)} p_X(x')}. \tag{B1}$$

We use this expression to rewrite the EP from Eq. (12) as

$$\Sigma(p_X) = \sum_x p_X(x) \left\{ \ln \frac{p_{X|f(X)}[x|f(x)]}{e^{-Q(x)/kT - \ln Z[f(X)]}} - \ln Z[f(x)] \right\}, \tag{B2}$$

where we have defined

$$Z(y) := \sum_{x:f(x)=y} e^{-Q(x)/kT}. \tag{B3}$$

Now define the following conditional distribution:

$$w_{X|f(X)}[x|f(x)] := e^{-Q(x)/kT} - \ln Z[f(x)]. \tag{B4}$$

Using this definition, we can further rewrite Eq. (B2) as

$$\Sigma(p_X) = D[p_{X|f(X)} \| w_{X|f(X)}] - \langle \ln Z[f(x)] \rangle_{p_X}, \tag{B5}$$

where $D[p_{X|f(X)} \| w_{X|f(X)}]$ indicates the conditional KL divergence between the conditional distribution $p_{X|f(X)}$ and $w_{X|f(X)}$ [108].

As we show below in Eq. (B6), $-\ln Z[f(x)] \geqslant 0$ for all x. Thus, Eq. (B5) implies $\Sigma(p_X) \geqslant D[p_{X|f(X)} \| w_{X|f(X)}]$. Note that this lower bound is nonnegative and vanishes whenever $p_{X|f(X)} = w_{X|f(X)}$. This means that $w_{X|f(X)}$, as defined in Eq. (B4), encodes that conditional probability of inputs x given outputs $f(x)$ that achieves minimal EP for a realization of f with heat function Q.

In our previous work, we have sometimes referred to the conditional KL divergence in Eq. (B5) as *mismatch cost*. Using the chain rule for KL divergence, we write mismatch cost as

$$D[p_{X|f(X)} \| w_{X|f(X)}] = D(p_X \| w_X) - D[p_{f(X)} \| w_{f(X)}],$$

where $w_{f(X)}(y) = \sum_{x:f(x)=y} w_X(x)$, while $w_X(x)$ is any distribution that obeys

$$w_X(x)/w_X(x') = e^{[Q(x')-Q(x)]/kT} \quad \forall x, x' : f(x) = f(x').$$

In our previous work, we referred to the distribution $w_X(x)$ as a *prior*. (This term was originally motivated by a Bayesian interpretation of EP [107].) As long as $|\mathrm{img}\, f| > 1$, there are an infinite number of priors for any given $w_{X|f(X)}$, since the relative probabilities of any pair x, x' with $f(x) \neq f(x')$ are unconstrained.

In our previous work [34,106], we referred to the term $-\langle \ln Z[f(X)] \rangle_{p_X}$ in Eq. (B5) as the *residual EP*. Observe that for any $y \in \mathrm{img}\, f$,

$$\begin{aligned} \Sigma[w_{X|f(X)=y}] &= D[w_{X|f(X)=y} \| w_{X|f(X)}] - \ln Z(y) \\ &= -\ln Z(y). \end{aligned} \tag{B6}$$

Since $\Sigma[w_{X|f(X)=y}] \geqslant 0$ by the second law, $-\ln Z(y)$ is nonnegative for all $y \in \mathrm{img}\, f$ and therefore residual EP is always nonnegative. Note also that the residual EP is an expectation under p_X, and thus it is linear in p_X. In fact, it only depends on the probabilities assigned to each output $p_{f(X)}(y)$, not the conditional distribution of inputs corresponding to each output. In our other work [106], we have sometimes called the indexed set $\{-\ln Z(y)\}_y$ the *residual EP parameter*.

Finally, define an *island* of f as a preimage $f^{-1}(y)$ for some y, with $L(f)$ the set of all islands of U. We can rewrite

Eq. (B5) as

$$\Sigma(p_X) = \sum_{c \in L(f)} p(c)\{D(p_{X|X\in c} \| w_{X|X\in c}) - \ln Z[f(c)]\},$$

where $p(c) = \sum_{x\in c} p_X(x)$. Intuitively, this expression shows that any realization of the function f can be thought of a set of (island-indexed) "parallel" processes, operating independently of one another on nonoverlapping subsets of $\mathcal{X}$, each generating EP given by the associated mismatch cost and residual EP.

This form of mismatch cost, residual EP, and island decomposition was introduced in [34,106,107]. It holds even in the general case of nondeterministic dynamics, with an appropriate (more general) definition of the prior w_X and the island decomposition. However, that previous work on mismatch cost and residual EP assumed finite state spaces. The derivation presented above does not have that restriction.

APPENDIX C: PROOF OF PROPOSITION 1

The following proof will make use of the decomposition of EP derived in Appendix B.

Proposition 1. Given a countable set $\mathcal{X}$, let $f : \mathcal{X} \to \mathcal{X}$ and $G : \mathcal{X} \to \mathbb{R}$ be two partial functions with the same domain of definition. The following are equivalent:

(1) For all p_X with $\text{supp}\, p_X \subseteq \text{dom}\, f$,

$$\langle G \rangle_{p_X} + S[p_{f(X)}] - S(p_X) \geqslant 0. \tag{14}$$

(2) For all $y \in \text{img}\, f$,

$$\sum_{x:f(x)=y} e^{-G(x)} \leqslant 1. \tag{15}$$

(3) There exists a realization of f coupled to a heat bath at temperature T, whose heat function Q obeys

$$Q(x)/kT = G(x) \quad \forall\, x \in \text{dom}\, f. \tag{16}$$

Proof. Note that condition 1 follows from condition 3 by the second law of thermodynamics. To show equivalence of all three conditions, we proceed in the following way:

(1) We show that condition 2 is implied by condition 1.

(2) We show that condition 1 is implied by condition 2.

(3) We show by construction that condition 2 implies condition 3.

Given that $\mathcal{X}$ is countable, we assume that $\mathcal{X} \subseteq \mathbb{N}$. This is done without loss of generality: if elements of $\mathcal{X}$ are not natural numbers, one can put a total order on $\mathcal{X}$ using the natural numbers.

We now prove that condition 1 implies condition 2. First, define the function F to refer the expression in Eq. (14),

$$F(p_X) = \sum_x p_X(x)\{G(x) - \ln p_{f(X)}[f(x)] + \ln p_X(x)\}. \tag{C1}$$

Let $\mathcal{A}_n(y)$ indicate the first n elements of $f^{-1}(y)$, and define the initial distribution

$$p_X^{(n)}(x) = \begin{cases} e^{-G(x)}/Z_n(y) & \text{if } x \in \mathcal{A}_n(y) \\ 0 & \text{otherwise} \end{cases},$$

where $Z_n(y) = \sum_{x\in\mathcal{A}_n(y)} e^{-G(x)}$. Note that $\text{supp}\, p_X^{(n)} \subseteq \text{dom}\, f$. Plugging into Eq. (C1) and simplifying gives

$$F\left(p_X^{(n)}\right) = -\ln Z_n(y) \geqslant 0,$$

or equivalently $Z_n(y) \leqslant 1$. Since this holds for all n,

$$Z(y) = \sum_{x:f(x)=y} e^{-G(x)} = \lim_{n\to\infty} Z_n(y) \leqslant 1. \tag{C2}$$

We now prove that condition 1 is implied by condition 2. Define $w_{X|f(X)}(x|f(x))$ as in Eq. (B4), while taking $Q/kT = G$. Then, use the results in Appendix B to rewrite F as

$$\begin{aligned} F(p_X) &= D[p_{X|f(X)} \| w_{X|f(X)}] - \langle \ln Z[f(X)] \rangle_{p_X} \\ &\geqslant D[p_{X|f(X)} \| w_{X|f(X)}] \geqslant 0. \end{aligned}$$

The first inequality follows from the assumption that $Z(y) = \sum_{x:f(x)=y} e^{-G(x)} \leqslant 1$ for all $y \in \text{img}\, f$, and the second inequality follows from the nonnegativity of conditional KL divergence [108].

The rest of this proof shows by construction that condition 3 follows from condition 2. For simplicity, assume that the physical process has access to a set of "auxiliary" states, one for each $y \in \text{img}\, f$. We use x_y to indicate the auxiliary state corresponding to each y, and assume that $x_y \notin \text{dom}\, f$. For notational convenience, let $\mathcal{W} := \text{dom}\, f \cup \{x_y : y \in \text{img}\, f\}$. Then, define the following function $\hat{f} : \mathcal{W} \to \mathcal{X}$,

$$\text{For any } x \in \text{dom}\, f, \quad \hat{f}(x) := f(x),$$

$$\text{For any } y \in \text{img}\, f, \quad \hat{f}(x_y) := y.$$

In words, any x in the domain of f is mapped by $\hat{f}$ to $f(x)$, while any auxiliary state x_y is mapped by $\hat{f}$ to y.

Now define the following Hamiltonian $H : \mathcal{W} \to \mathbb{R} \cup \{\infty\}$,

$$\forall x \in \text{dom}\, f : H(x) := f(x) + kT\, G(x) \tag{C3}$$

$$\forall y \in \text{img}\, f : H(x_y) := y - kT\, \ln\left[1 - \sum_{x:f(x)=y} e^{-G(x)}\right]. \tag{C4}$$

We use $\pi(w) = e^{-H(w)/kT}/Z$ to indicate the Boltzmann distribution for Hamiltonian H, where $Z = \sum_{x\in\mathcal{W}} e^{-H(x)/kT}$ is the partition function. Note that the partition function converges when $\beta > 0$,

$$\begin{aligned} Z &= \sum_{y\in\text{img}\, f}\left[e^{-H(x_y)/kT} + \sum_{x:f(x)=y} e^{-H(x)/kT}\right] \\ &= \sum_{y\in\text{img}\, f} e^{-y/kT} \leqslant \sum_{i\in\mathbb{N}} e^{-i/kT} = 1/(e^{1/kT} - 1). \end{aligned} \tag{C5}$$

To derive the second line, we plugged Eqs. (C3) and (C4) into Eq. (C5) and simplified.

We now consider the following physical process over $t \in [0, t_f]$, applied to a system coupled to a work reservoir and a heat bath at temperature T:

(1) At $t = 0$, the Hamiltonian H is applied to the system.

(2) Over $t \in (0, \tau]$, the system is allowed to freely relax toward equilibrium. However, the only allowed transitions are those between pairs of states w, w' that have $\hat{f}(w) = \hat{f}(w')$.

We assume that by $t = \tau$, the system has reached a stationary distribution.

(3) Over $t \in (\tau, t_f]$, the system undergoes a quasistatic physical process that implements the map $\hat{f}$ from initial to final states, and does so in a thermodynamically reversible way for initial distribution π. There are numerous known ways of constructing such a process [15,16,109].

Note that the above procedure assumes a separation of timescales (i.e., the relaxation time of the system is infinitely faster than τ and $t_f - \tau$). Step (3) also assumes an idealized heat bath (infinite heat capacity, weak coupling, infinitely fast relaxation time [110]).

The above procedure will map any $x \in \text{dom}\, f$ to final state $f(x)$. Let Q indicate the heat function of this process. We will show that $Q(x)/kT = G(x)$ for any $x \in \text{dom}\, f$. First, let δ_x indicate an initial distribution which is a δ function over some state x. Note that

$$\Sigma(\delta_x) = S[\delta_{f(x)}] - S(\delta_x) + \langle Q \rangle_{\delta_x}/kT = Q(x)/kT, \quad \text{(C6)}$$

where we have used the fact that $S(\delta_x) = S[\delta_{f(x)}] = 0$. We then analyze $\Sigma(\delta_x)$. Step (1) and step (3) in the above construction incur no EP. For step (2), EP incurred during free relaxation from $t = 0$ to $t = \tau$ is given by

$$\Sigma(\delta_x) = D(\delta_x \| \pi) - D(p_x^\tau \| \pi), \quad \text{(C7)}$$

where p_x^τ is the state distribution at time τ, given that the system started in distribution δ_x at $t = 0$. By construction, p_x^τ will be equal to the equilibrium distribution restricted to a subset of states,

$$p_x^\tau(w) = \frac{\delta(\hat{f}(w), \hat{f}(x))\pi(w)}{\sum_{w'} \delta(\hat{f}(w'), \hat{f}(x))\pi(w')}.$$

It can be verified, using the definition of δ_x and π, that

$$D(\delta_x \| \pi) = f(x)/kT + G(x) + \ln Z.$$

Similarly, it can be verified using the definition of p_x^τ that

$$D(p_x^\tau \| \pi) = f(x)/kT + \ln Z.$$

Plugging these two KL divergences into Eq. (C7) gives

$$\Sigma(\delta_x) = G(x). \quad \text{(C8)}$$

Combining with Eq. (C6) gives $Q(x)/kT = \Sigma(\delta_x) = G(x)$. ■

It can be verified that the physical process constructed in the proof of Proposition 1 is thermodynamically reversible if it is started with the initial equilibrium distribution π_X, so that the free relaxation in step (2) incurs no EP. Generally, this equilibrium distribution will have support on the auxiliary states, which are outside of dom f. However, consider the case when Eq. (15) is an equality for all $y \in \text{img}\, f$. Then the definition in Eq. (C4) gives $H(x_y) = \infty$ and $\pi_X(x_y) = 0$ for all $y \in \text{img}\, f$. In this case, the input distribution $p_X = \pi_X$ obeys supp $p_X \subseteq$ dom f and achieves zero EP. Moreover, using the decomposition in Appendix B, it can be verified that if Eq. (15) is an equality for all $y \in \text{img}\, f$, then any input distribution that obeys $p_{X|f(X)} = \pi_{X|f(X)}$, as defined in Eq. (B1), also achieves zero EP.

APPENDIX D: $O(1)$ HEAT FOR COIN-FLIPPING REALIZATION IS UNCOMPUTABLE

Let ϕ_U indicate the partial function computed by some UTM U. Imagine there is some computable function f such that for any y, $f(y)$ returns an input for ϕ_U that outputs y and generates bounded heat under the coin-flipping realization [i.e., $\phi_U[f(y)] = y$ and $Q_{\text{coin}}[f(y)] = O(1)$]. Then, by Eq. (25), it must be that $\ell[f(y)] = K(y) + O(1)$. Since $\ell(\cdot)$ is a computable function, this would in turn imply that there is a computable function $g(y) = K(y) + O(1)$. However, such a function cannot exist, as shown in the following proposition.

Proposition 1. There is no computable partial function $g : \{0, 1\}^* \to \mathbb{N}$ such that for all y,

$$g(y) = K(y) + O(1). \quad \text{(D1)}$$

Proof. We say that $p_Y(y)$ is a *semimeasure* if $p_Y(y) \geqslant 0$ for all y and $\sum_y p_Y(y) \leqslant 1$ (i.e., it is a nonnormalized probability distribution). We say that a semimeasure $p_Y(y)$ *(multiplicatively) dominates* another semimeasure $q_Y(y)$ if there is some constant $c > 0$ such that $p_Y(y) \geqslant c q_Y(y)$ for all y.

Assume that a computable $g(y) = K(y) + O(1)$ exists. Then $q_Y(y) := 2^{-g(y)}$ would be a computable semimeasure that dominates $p_Y(y) := 2^{-K(y)}$. It is known that $p_Y(y)$ dominates every computable semimeasure [[42], Thm. 4.3.3 and Cor. 4.3.1]. Since domination is transitive, if $g(y)$ were computable, then $q_Y(y)$ would be a computable semimeasure that dominates every computable semimeasure. However, such a semimeasure cannot exist by Lemma 4.3.1 in Ref. [42]. ■

APPENDIX E: PROOF OF EQ. (29)

Let f indicate any computable partial function. In this Appendix, we show that the dominating realization of f, with heat function

$$Q_{\text{dom}}(x) = kT \ln 2\, K[x|f(x)], \quad \text{(E1)}$$

is better than any other realization of f with an upper-semicomputable heat function Q, up to an additive constant.

We first prove the following two useful results.

Lemma 2. For any partial function $f : \{0, 1\}^* \to \{0, 1\}^*$,

$$\sum_{x:f(x)=y} e^{-\ln 2 \cdot K(x|y)} \leqslant 1 \quad \forall y \in \text{img}\, f.$$

Proof. For all $y \in \text{img}\, f$, we have the following:

$$\sum_{x:f(x)=y} e^{-\ln 2 \cdot K(x|y)} = \sum_{x:f(x)=y} 2^{-K(x|y)} \leqslant \sum_{x \in \{0,1\}^*} 2^{-K(x|y)}.$$

In addition, we have the bound

$$\sum_{x \in \{0,1\}^*} 2^{-K(x|y)} \leqslant 1, \quad \text{(E2)}$$

which comes from Kraft's inequality and the fact that, for any given y, the set $\{K(x|y) : x \in \{0, 1\}^*\}$ specifies the lengths of a prefix-free code [[42], p. 252 and p. 287]. Combining gives the desired result. ■

Proposition 3. Let $f : \{0, 1\}^* \to \{0, 1\}^*$ be a computable partial function, $Q : \{0, 1\}^* \to \mathbb{R}$ a upper-semicomputable

partial function with dom $Q \supseteq$ dom f. If for all $y \in$ img f,

$$\sum_{x:f(x)=y} e^{-Q(x)} \leqslant 1, \tag{E3}$$

then for all $x \in$ dom f,

$$Q(x) \geqslant \ln 2\{K[x|f(x)] - K(Q, f)\} + O(1), \tag{E4}$$

where $O(1)$ is a constant independent of x and Q.

Proof. Let M indicate the TM that computes f, and let $a(x, n)$ be a computable partial function which upper-semicomputes $Q(x)/\ln 2$. Then, define the following TM B: Given inputs $x \in \{0, 1\}^*$, $y \in \{0, 1\}^*$, and $n \in \mathbb{N}$, the TM B runs M for n steps on input x. If M halts within that time on output y, then B outputs $2^{-a(x,n)}$. Otherwise, B outputs 0 and halts.

Then, for any $x \in$ dom f, define

$$\begin{aligned} s(x|y) &:= \lim_{n\to\infty} \phi_B(\langle x, y\rangle, n) \\ &= \delta(f(x), y)2^{-Q(x)/\ln 2} \\ &= \delta(f(x), y)e^{-Q(x)}. \end{aligned} \tag{E5}$$

It is easy to check that $\phi_B(\langle x, y\rangle, n)$ is nondecreasing in n, so $s(x|y)$ is lower-semicomputable [i.e., $\phi_B(\langle x, y\rangle, n) \leqslant \phi_B(\langle x, y\rangle, n + 1)$ and $\lim_{n\to\infty} \phi_B(\langle x, y\rangle, n) = s(x|y)$]. Moreover, if one had a program that computed both f and Q, then one could lower-semicompute s. This means that

$$K(s) \leqslant K(Q, f) + O(1), \tag{E6}$$

where $K(Q, f)$ is the Kolmogorov complexity of jointly computing the functions f and Q.

By assumption in Eq. (E3), for any $y \in$ img f,

$$\sum_{x\in\text{dom } f} s(x|y) = \sum_{x:f(x)=y} 2^{-Q(x)/\ln 2} = \sum_{x:f(x)=y} e^{-Q(x)} \leqslant 1. \tag{E7}$$

This means that $s(x|y)$ is a so-called *conditional semimeasure* of x given y (i.e., a nonnormalized conditional probability measure). For any lower-semicomputable conditional semimeasure s, an existing result in AIT [[111], Cor. 2] states

$$K(x|y) \leqslant -\log_2 s(x|y) + K(s) + O(1).$$

Taking $y = f(x)$ and plugging in Eqs. (E5) and (E6) gives

$$K[x|f(x)] \leqslant Q(x)/\ln 2 + K(Q, f) + O(1). \tag{E8}$$

Equation (E4) follows by rearranging. ∎

Given that Eq. (E3) holds, by Proposition 1 there must be a realization of f with heat function $Q(x) = kTG(x)$. By Lemma 2, we can take $G(x) = \ln 2K[x|f(x)]$. Thus, there must exists a realization of f with heat function Q_{dom}, as defined in Eq. (E1).

Combining $G(x) = Q(x)/kT$ with Eq. (E4), and multiplying both sides by kT, gives the following inequality:

$$Q(x) \geqslant Q_{\text{dom}}(x) - kT \ln 2\, K(Q/kT, f) + O(1).$$

We can derive a slightly weaker, but more interpretable, lower bound by using $K(Q/kT, f) \leqslant K(Q/kT) + K(f) + O(1)$, which follows from the subadditivity of Kolmogorov complexity [[42], p. 202]. This allows to rewrite the above as

$$Q(x) \geqslant Q_{\text{dom}}(x) - kT \ln 2\,[K(Q/kT) + K(f)] + O(1),$$

which appears in the main text as Eq. (29), with $f = \phi_M$.

APPENDIX F: INFINITE EXPECTED HEAT

Let ϕ_U be the partial function computed by some UTM U. In the following results, we will make use of the following decomposition of the drop of entropy, which holds for any initial distribution p_X:

$$S(p_X) - S(p_Y) = \sum_{y\in\text{img } \phi_U} p_Y(y) S[p_{X|\phi_U(x)=y}]. \tag{F1}$$

Note that (discrete) Shannon entropy is nonnegative, so $S[p_{X|\phi_U(x)=y}] \geqslant 0$ for all y. For simplicity, and without loss of generality, in this section we will write Shannon entropies in units of bits.

We will make use of the following lemmas.

Lemma 4. For any $y \in \{0, 1\}^*$,

$$\sum_{x:\phi_U(x)=y} 2^{-\ell(x)}\ell(x) = \infty.$$

Proof. To derive this result, we make use of a simple prefix-free code for natural numbers $i \in \mathbb{N}$:

$$g(i) = \underbrace{111...111}_{\lceil\log_2 i\rceil \text{ 1s}} 0 \underbrace{1110...0110}_{\substack{\text{Encoding of } i \\ \text{with } \lceil\log_2 i\rceil \text{ bits}}}. \tag{F2}$$

(See also Ref. [[42], Section 1.11].) It is straightforward to check that this prefix-free code achieves a code length

$$\ell[g(i)] = 2\lceil\log_2 i\rceil + 1. \tag{F3}$$

In addition, we will also use programs of the form $z_y + g(i) + x$ such that $\phi_U[z_y + g(i) + x] = y$, where z_y is some appropriate prefix string, $g(i)$ is defined in Eq. (F2), x is any binary string with $\ell(x) = i$, and "+" indicates concatenation. In words, the program $z_y + g(i) + x$ causes U to read in a code for y (corresponding to z_y), then a prefix-free code for any $i \in \mathbb{N}$ [corresponding to $g(i)$], then "swallow" i bits of input (corresponding to x), and halt after outputting y. Using Eq. (F3), it can be checked that

$$\begin{aligned} i = \ell(x) &< \ell[z_y + g(i) + x] \\ &= \ell(z_y) + \ell[g(i)] + i \leqslant \ell(z_y) + 2\log_2 i + 3 + i. \end{aligned} \tag{F4}$$

We now bound the sum $\sum_{x:\phi_U(x)=y} 2^{-\ell(x)}\ell(x)$. Since all terms in this sum are positive, we can lower bound it by focusing only on the subset of programs of the form $z_y + g(i) + x$:

$$\begin{aligned} \sum_{x:\phi_U(x)=y} 2^{-\ell(x)}\ell(x) &\geqslant \sum_{i\in\mathbb{N},x:\ell(x)=i} 2^{-\ell(z_y+g(i)+x)}\ell[z_y + g(i) + x] \\ &\overset{(a)}{\geqslant} \sum_{i\in\mathbb{N},x:\ell(x)=i} 2^{-\ell(z_y)-2\log_2 i-3-i} i \\ &= 2^{-\ell(z_y)-3} \sum_{i\in\mathbb{N},x:\ell(x)=i} 2^{-2\log_2 i-i} i \end{aligned}$$

$$\stackrel{(b)}{=} 2^{-\ell(z_y)-3} \sum_{i\in\mathbb{N}} 2^i 2^{-i-2\log_2 i} i$$
$$= 2^{-\ell(z_y)-3} \sum_{i\in\mathbb{N}} i/i^2$$
$$= 2^{-\ell(z_y)-3} \sum_{i\in\mathbb{N}} 1/i = \infty.$$

In (a), we use the lower and upper bounds on $\ell[z_y + g(i) + x]$ from Eq. (F4), and in (b) we use that there are 2^i different bit strings x that obey $\ell(x) = i$. The rest of the steps follow from rearranging and simplifying. ■

Lemma 5. For any computable partial function $f : \{0,1\}^* \to \{0,1\}^*$ and $x \in \text{dom}\, f$,

$$K(x) \leqslant \ell(x) + O(1).$$

Proof. Let M be a TM which computes f, and note that $\text{dom}\, \phi_M$ is a prefix-free set. Consider the Kolmogorov complexity $K_U(x)$, which is defined in terms of a UTM U which operates in the following way: U takes inputs of the form $b + x$, where $b \in \{0,1\}$, $x \in \{0,1\}^*$ and "+" indicates string concatenation. If $b = 0$, then U emulates some prefix UTM on input x and outputs the result. If $b = 1$, then U emulates M on input x while swallowing the output; if and when M halts on input x, U outputs a copy of the input x and halts. It is clear that U is universal, due to its behavior when $b = 0$, and that it is prefix-free. It is also clear that U has a program of length $\ell(x) + 1$ that can be used to output any $x \in \text{dom}\, \phi_M$, due to its behavior when $b = 1$. Thus, $K_U(x) \leqslant \ell(x) + 1$. The result follows by recalling the invariance theorem, $K(x) = K_U(x) + O(1)$, where $K(\cdot)$ is the Kolmogorov complexity defined for some arbitrary reference UTM. ■

1. Coin-flipping distribution

In this section, we consider the coin-flipping input distribution, p_X^{coin}, as defined in Eq. (17). We show that the drop in entropy for this input distribution is infinite,

$$S\big(p_X^{\text{coin}}\big) - S\big(p_Y^{\text{coin}}\big) = \infty. \tag{F5}$$

Thus, by the second law of thermodynamics, Eq. (12), any realization which carries out U on p_X^{coin} must generate an infinite amount of heat.

To derive Eq. (F5), first use Eq. (F1), to write

$$S\big(p_X^{\text{coin}}\big) - S\big(p_Y^{\text{coin}}\big) = \sum_{y\in\text{img}\, U} p_Y^{\text{coin}}(y) S\big[p^{\text{coin}}_{X|\phi_U(x)=y}\big]. \tag{F6}$$

We now show that $S[p^{\text{coin}}_{X|\phi_U(x)=y}] = \infty$ for any $y \in \text{img}\, \phi_U$. First, write

$$S[p^{\text{coin}}_{X|\phi_U(x)=y}] = -\sum_{x:\phi_U(x)=y} p^{\text{coin}}_{X|Y}(x|y) \log_2 p^{\text{coin}}_{X|Y}(x|y)$$
$$= \log_2 m_Y(y) - \frac{1}{m_Y(y)} \sum_{x:\phi_U(x)=y} 2^{-\ell(x)} \log_2 2^{-\ell(x)}$$
$$= \log_2 m_Y(y) + \frac{1}{m_Y(y)} \sum_{x:\phi_U(x)=y} 2^{-\ell(x)} \ell(x), \tag{F7}$$

where we use that $p^{\text{coin}}_{X|Y}(x|y) = 2^{-\ell(x)}/m_Y(y)$ when $\phi_U(x) = y$ (similarly to the derivation in Sec. IV). Note that the multiplicative constant $1/m_Y(y)$ is strictly positive and the additive constant $\log_2 m_Y(y)$ is finite. Then Eq. (F7) is infinite by Lemma 4.

2. EP optimal distribution for the dominating realization

Consider any initial distribution of the form

$$p_X(x) = \frac{w_Y[\phi_U(x)]}{C[\phi_U(x)]} 2^{-K[x|\phi_U(x)]}, \tag{F8}$$

where $C(y) := \sum_{x:\phi_U(x)=y} 2^{-K(x|y)}$ is a normalization constant and w_Y is any probability distribution over $\text{img}\, \phi_U$. It can be verified, using results discussed in Appendix B, that any input distribution of the form Eq. (F8) achieves 0 mismatch cost for the dominating realization. Thus, this distribution achieves minimal EP for the dominating realization.

In this section, we show that any input distribution of the form Eq. (F8) also incurs an infinite drop in entropy,

$$S(p_X) - S(p_Y) = \infty. \tag{F9}$$

Thus, by the second law of thermodynamics, Eq. (12), any realization which carries out U on such an input distribution p_X must generate an infinite amount of heat.

Our derivation proceeds in a similar manner as that used above to show that the drop in entropy for p_X^{coin} was infinite. First, use Eq. (F1) to write

$$S(p_X) - S(p_Y) = \sum_{y\in\text{img}\, \phi_U} p_Y(y) S[p_{X|\phi_U(x)=y}]. \tag{F10}$$

We derive Eq. (F9) by showing that $S[p_{X|\phi_U(x)=y}] = \infty$ for any $y \in \text{supp}\, w_Y$. First, write

$$S[p_{X|\phi_U(x)=y}] = -\sum_{x:\phi_U(x)=y} p_{X|Y}(x|y) \log_2 p_{X|Y}(x|y)$$
$$= \log_2 C(y) + \frac{1}{C(y)} \sum_{x:\phi_U(x)=y} 2^{-K(x|y)} K(x|y), \tag{F11}$$

where we use that $p_{X|Y}(x|y) = 2^{-K(x|y)}/C(y)$ when $\phi_U(x) = y$ and $w_Y(y) > 0$. To show that Eq. (F11) is infinite, we note that $C(y) > 0$ and then focus on the inner sum

$$\sum_{x:\phi_U(x)=y} 2^{-K(x|y)} K(x|y). \tag{F12}$$

Note that any x such that $\phi_U(x) = y$ must obey $x \in \text{dom}\, \phi_U$. This means that

$$K(x|y) \leqslant K(x) + O(1) \leqslant \ell(x) + O(1),$$

where the first inequality comes from subadditivity of Kolmogorov complexity [42], while the second comes from Lemma 5. We will use $\kappa \geqslant 0$ to indicate some finite constant that makes the rightmost inequality hold.

Now note that $2^{-a}a$ is nonincreasing in $a \in \mathbb{N}$ for all $a \geqslant 1$. Assume for the moment that there is no x such that $\phi_U(x) = y$ and $K(x|y) = 0$. Then,

$$2^{-K(x|y)} K(x|y) \geqslant 2^{-\ell(x)-\kappa}[\ell(x) + \kappa] \geqslant 2^{-\kappa} 2^{-\ell(x)} \ell(x)$$

for all x such that $\phi_U(x) = y$. This gives the following lower bound for Eq. (F12):

$$\sum_{x:\phi_U(x)=y} 2^{-K(x|y)} K(x|y) \geqslant 2^{-\kappa} \sum_{x:\phi_U(x)=y} 2^{-\ell(x)} \ell(x) = \infty,$$

where the last equality uses Lemma 4. Now imagine that there is an x such that $\phi_U(x) = y$ and $K(x|y) = 0$ (for any given y, there can be at most one such x). In that case, the above lower bound should be decreased by $2^{-\kappa} 2^{-\ell(x)} \ell(x)$, which is a finite constant, so Eq. (F12) is still infinite.

APPENDIX G: STRICTLY POSITIVE EP FOR THE DOMINATING DISTRIBUTION

Consider any computable partial function f, and recall the decomposition of EP developed in Appendix B, into a nonnegative "mismatch cost" (conditional KL) term and a nonnegative "residual EP" term, Eq. (B5). The residual EP term is an expected over nonnegative values $-\ln Z(y)$ for $y \in \text{img}\, f$.

Using Eq. (B3), we write this residual term for the dominating realization as

$$-\ln Z(y) = -\ln \sum_{x:f(x)=y} e^{-Q_{\text{dom}}(x)/kT} = -\ln \sum_{x:f(x)=y} 2^{-K(x|y)},$$

where we substituted in the definition of Q_{dom} from Eq. (28). Assume that the conditional Kolmogorov complexity is defined relative to some reference UTM U, $K(x|y) = K_U(x|y)$. Then, consider the inner sum,

$$\sum_{x:\phi_U(x)=y} 2^{-K_U(x|y)} \leqslant \sum_{x\in\{0,1\}^*} 2^{-K_U(x|y)} < \sum_{(z,y)\in \text{dom}\, \phi_U} 2^{-\ell(z)} \leqslant 1.$$

The strict inequality comes from the fact that not all programs $(z, y) \in \text{dom}\, \phi_U$ are the shortest program for some output string $x \in \{0, 1\}^*$. The last inequality comes from the Kraft inequality.

This shows that for the dominating realization of a computable function f, $-\ln Z(y) > 0$ for all $y \in \text{img}\, f$. Thus, the residual EP term in Eq. (B5) is strictly positive for any input distribution.

APPENDIX H: DERIVATION OF EQ. (31)

For a coin-flipping realization of some UTM U, Eq. (25) states that the heat generated on input x is given by

$$\begin{aligned} Q_{\text{coin}}(x) &= kT \ln 2\, [\ell(x) - K[\phi_U(x)]] + O(1) \\ &\geqslant kT \ln 2\, [K(x) - K[\phi_U(x)]] + O(1), \end{aligned}$$

where the second line uses Lemma 5. We now use the following inequality [[42], Sec. 3.9.2]:

$$\begin{aligned} &K[\phi_U(x)] \\ &\quad \leqslant K[x, \phi_U(x)] - K[x|\phi_U(x)] + O\{\log K[\phi_U(x)]\} \\ &\quad = K(x) - K[x|\phi_U(x)] + O\{\log K[\phi_U(x)]\}, \end{aligned}$$

where in the last line we have used that $K(x, \phi_U(x)) = K(x) + O(1)$ (since the value of $\phi_U(x)$ is by definition computable from x). Combining the above results with the definition of Q_{dom} gives the desired result,

$$Q_{\text{coin}}(x) \geqslant Q_{\text{dom}}(x) - O\{\log K[\phi_U(x)]\}. \tag{H1}$$

[1] L. Brillouin, Negentropy principle of information, J. Appl. Phys. **24**, 1152 (1953).

[2] L. Brillouin, *Science and Information Theory* (Academic Press, New York, 1962).

[3] R. Landauer, Irreversibility and heat generation in the computing process, IBM J. Res. Dev. **5**, 183 (1961).

[4] L. Szilard, On the decrease of entropy in a thermodynamic system by the intervention of intelligent beings, Behav. Sci. **9**, 301 (1964).

[5] W. H. Zurek, Thermodynamic cost of computation, algorithmic complexity and the information metric, Nature **341**, 119 (1989).

[6] W. H. Zurek, Algorithmic randomness and physical entropy, Phys. Rev. A **40**, 4731 (1989).

[7] C. H. Bennett, The thermodynamics of computation—A review, Int. J. Theor. Phys. **21**, 905 (1982).

[8] S. Lloyd, Use of mutual information to decrease entropy: Implications for the second law of thermodynamics, Phys. Rev. A **39**, 5378 (1989).

[9] J. Dunkel, Thermodynamics: Engines and demons, Nat. Phys. **10**, 409 (2014).

[10] É. Roldán, I. A. Martínez, J. M. R. Parrondo, and D. Petrov, Universal features in the energetics of symmetry breaking, Nat. Phys. **10**, 457 (2014).

[11] S. Lloyd, Ultimate physical limits to computation, Nature **406**, 1047 (2000).

[12] E. Fredkin, An informational process based on reversible universal cellular automata, Physica D **45**, 254 (1990).

[13] T. Toffoli and N. H. Margolus, Invertible cellular automata: A review, Physica D **45**, 229 (1990).

[14] H. S. Leff and A. F. Rex, *Maxwell's Demon: Entropy, Information, Computing* (Princeton University Press, Princeton, NJ, 2014).

[15] O. J. E. Maroney, Generalizing landauer's principle, Phys. Rev. E **79**, 031105 (2009).

[16] S. Turgut, Relations between entropies produced in nondeterministic thermodynamic processes, Phys. Rev. E **79**, 041102 (2009).

[17] C. Van den Broeck, Stochastic thermodynamics: A brief introduction, *Physics of Complex Colloids*, Vol. 184, Proceedings of the International School of Physics "Enrico Fermi" (IOS Press, 2013), pp. 155–193.

[18] C. Van den Broeck and M. Esposito, Ensemble and trajectory thermodynamics: A brief introduction, Physica A **418**, 6 (2015).

[19] U. Seifert, Stochastic thermodynamics, fluctuation theorems and molecular machines, Rep. Prog. Phys. **75**, 126001 (2012).

[20] A. Bérut, A. Arakelyan, A. Petrosyan, S. Ciliberto, R. Dillenschneider, and E. Lutz, Experimental verification of landauer's principle linking information and thermodynamics, Nature **483**, 187 (2012).

[21] G. Diana, G. B. Bagci, and M. Esposito, Finite-time erasing of information stored in fermionic bits, Phys. Rev. E **87**, 012111 (2013).

[22] P. R. Zulkowski and M. R. DeWeese, Optimal finite-time erasure of a classical bit, Phys. Rev. E **89**, 052140 (2014).

[23] Y. Jun, M. Gavrilov, and J. Bechhoefer, High-Precision Test of Landauer's Principle in a Feedback Trap, Phys. Rev. Lett. **113**, 190601 (2014).

[24] S. Ciliberto, Experiments in Stochastic Thermodynamics: Short History and Perspectives, Phys. Rev. X **7**, 021051 (2017).

[25] A. C. Barato and U. Seifert, Unifying Three Perspectives on Information Processing in Stochastic Thermodynamics, Phys. Rev. Lett. **112**, 090601 (2014).

[26] K. Wiesner, M. Gu, E. Rieper, and V. Vedral, Information-theoretic lower bound on energy cost of stochastic computation, Proc. R. Soc. Lond. A **468**, 4058 (2012).

[27] T. Sagawa and M. Ueda, Fluctuation Theorem with Information Exchange: Role of Correlations in Stochastic Thermodynamics, Phys. Rev. Lett. **109**, 180602 (2012).

[28] S. Still, D. A. Sivak, A. J. Bell, and G. E. Crooks, Thermodynamics of Prediction, Phys. Rev. Lett. **109**, 120604 (2012).

[29] M. Prokopenko, J. T. Lizier, and D. C. Price, On thermodynamic interpretation of transfer entropy, Entropy **15**, 524 (2013).

[30] M. Prokopenko and J. T. Lizier, Transfer entropy and transient limits of computation, Sci. Rep. **4**, 5394 (2014).

[31] J. V. Koski, V. F. Maisi, T. Sagawa, and 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).

[32] J. M. Parrondo, J. M. Horowitz, and T. Sagawa, Thermodynamics of information, Nat. Phys. **11**, 131 (2015).

[33] D. H. Wolpert, C. P. Kempes, P. Stadler, and J. Grochow (eds.), *Energetics of Computing in Life and Machines* (SFI Press, Santa Fe, NM, 2019).

[34] D. H. Wolpert, The stochastic thermodynamics of computation, J. Phys. A: Math. Theor. **52**, 193001 (2019).

[35] A. Boyd, Thermodynamics of correlations and structure in information engines, Ph.D. thesis, University of California Davis, 2018.

[36] P. Strasberg, J. Cerrillo, G. Schaller, and T. Brandes, Thermodynamics of stochastic turing machines, Phys. Rev. E **92**, 042104 (2015).

[37] J. A. Grochow and D. H. Wolpert, Beyond number of bit erasures: New complexity questions raisedby recently discovered thermodynamic costs of computation, ACM SIGACT News **49**, 33 (2018).

[38] P. Riechers, Transforming metastable memories: The nonequilibrium thermodynamics of computation, in *Energetics of Computing in Life and Machines*, edited by D. H. Wolpert, C. P. Kempes, P. Stadler, and J. Grochow (SFI Press, Santa Fe, NM, 2019).

[39] T. Ouldridge, R. Brittain, and P. Rein Ten Wolde, The power of being explicit: demystifying work, heat, and free energy in the physics of computation, in *Energetics of Computing in Life and Machines*, edited by D. H. Wolpert, C. P. Kempes, P. Stadler, and J. Grochow (SFI Press, Santa Fe, NM, 2019).

[40] M. Sipser, *Introduction to the Theory of Computation*, 2nd ed. (Thomson Course Technology, Boston, MA, 2006).

[41] J. E. Hopcroft, R. Motwani, and J. Ullman, *Introduction to Automata Theory, Languages and Computability* (Addison-Wesley Longman, Boston, MA, 2000).

[42] M. Li and 1, *An Introduction to Kolmogorov Complexity and Its Applications* (Springer, Berlin, 2008).

[43] P. Grunwald and P. Vitányi, Shannon information and Kolmogorov complexity, arXiv:cs/0410002 (2004).

[44] S. Arora and B. Barak, *Computational Complexity: A Modern Approach* (Cambridge University Press, Cambridge, UK, 2009).

[45] J. E. Savage, *Models of Computation* (Addison-Wesley, Reading, MA, 1998), Vol. 136.

[46] C. Moore and S. Mertens, *The Nature of Computation* (Oxford University Press, Oxford, 2011).

[47] S. Aaronson, Why philosophers should care about computational complexity, in *Computability: Turing, Gödel, Church, and Beyond* (MIT Press, Cambridge, MA, 2013), pp. 261–327.

[48] A. M. Turing, Intelligent machinery, Report for National Physical Laboratory, 1948.

[49] A. Church, Review of Turing (1936), J. Symb. Log. **2**, 42 (1937).

[50] G. Piccinini, Computation in physical systems, in *The Stanford Encyclopedia of Philosophy*, edited by E. N. Zalta, Metaphysics Research Lab (Stanford University, Berkeley, CA, 2017).

[51] T. S. Cubitt, D. Perez-Garcia, and M. M. Wolf, Undecidability of the spectral gap, Nature **528**, 207 (2015).

[52] T. S. Cubitt, J. Eisert, and M. M. Wolf, Extracting Dynamical Equations from Experimental Data is np Hard, Phys. Rev. Lett. **108**, 120503 (2012).

[53] D. Deutsch, Quantum theory, the church–turing principle and the universal quantum computer, Proc. R. Soc. Lond. A **400**, 97 (1985).

[54] P. Benioff, Quantum mechanical hamiltonian models of turing machines, J. Stat. Phys. **29**, 515 (1982).

[55] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, UK, 2010).

[56] J. D. Barrow, Godel and physics, in *Kurt Gödel and the Foundations of Mathematics: Horizons of Truth* (Cambridge University Press, New York, 2011), p. 255.

[57] S. Aaronson, NP-complete problems and physical reality, arXiv:quant-ph/0502072.

[58] R. Gandy, Church's thesis and principles for mechanisms, in *Studies in Logic and the Foundations of Mathematics* (Elsevier, Amsterdam, 1980), Vol. 101, pp. 123–148.

[59] S. Wolfram, Undecidability and Intractability in Theoretical Physics, Phys. Rev. Lett. **54**, 735 (1985).

[60] R. Geroch and J. B. Hartle, Computability and physical theories, Found. Phys. **16**, 533 (1986).

[61] M. A. Nielsen, Computable Functions, Quantum Measurements, and Quantum Dynamics, Phys. Rev. Lett. **79**, 2915 (1997).

[62] P. Arrighi and G. Dowek, The physical church-turing thesis and the principles of quantum theory, Int. J. Found. Comput. Sci. **23**, 1131 (2012).

[63] G. Piccinini, The physical church–turing thesis: Modest or bold?, Br. J. Philos. Sci. **62**, 733 (2011).

[64] I. Pitowsky, The physical church thesis and physical computational complexity, Iyyun: The Jerusalem Philosophical Quarterly **39**, 81 (1990).

[65] M. Ziegler, Physically-relativized church–turing hypotheses: Physical foundations of computing and complexity theory of computational physics, Appl. Math. Comput. **215**, 1431 (2009).

[66] C. Moore, Unpredictability and Undecidability in Dynamical Systems, Phys. Rev. Lett. **64**, 2354 (1990).

[67] N. C. da Costa and F. A. Doria, Undecidability and incompleteness in classical mechanics, Int. J. Theor. Phys. **30**, 1041 (1991).

[68] I. Kanter, Undecidability Principle and the Uncertainty Principle Even for Classical Systems, Phys. Rev. Lett. **64**, 332 (1990).

[69] T. D. Kieu, Computing the non-computable, Contemp. Phys. **44**, 51 (2003).

[70] B. J. Copeland, Hypercomputation, Minds and Machines **12**, 461 (2002).

[71] G. J. Chaitin, *Algorithmic Information Theory* (Cambridge University Press, Cambridge, UK, 2004), Vol. 1.

[72] C. Bennett, Logical reversibility of computation, IBM J. Res. Dev. **17**, 525 (1973).

[73] C. H. Bennett, Time/space trade-offs for reversible computation, SIAM J. Comput. **18**, 766 (1989).

[74] T. Sagawa, Thermodynamic and logical reversibilities revisited, J. Stat. Mech. (2014) P03025.

[75] T. Sagawa, Second law, entropy production, and reversibility in thermodynamics of information, in *Energy Limits in Computation* (Springer, Berlin, 2019), pp. 101–139.

[76] K. Morita, *Theory of Reversible Computing*, Monographs in Theoretical Computer Science. An EATCS Series (Springer Japan, Tokyo, 2017).

[77] J. Baez and M. Stay, Algorithmic thermodynamics, Math. Struct. Comput. Sci. **22**, 771 (2012).

[78] D. H. Wolpert, Extending Landauer's bound from bit erasure to arbitrary computation, arXiv:1508.05319 [cond-mat.stat-mech].

[79] D. H. Wolpert, Overview of information theory, computer science theory, and stochastic thermodynamics for thermodynamics of computation, in *Energetics of Computing in Life and Machines*, edited by D. H. Wolpert, C. P. Kempes, P. Stadler, and J. Grochow (SFI Press, Santa Fe, NM, 2019).

[80] W. H. Zurek, Algorithmic information content, church-turing thesis, physical entropy, and maxwell's demon, Technical Report, Los Alamos National Laboratory, New Mexico (1990).

[81] M. Li and P. Vitányi, Mathematical theory of thermodynamics of computation (unpublished).

[82] C. H. Bennett, P. Gács, M. Li, P. Vitányi, and W. H. Zurek, Thermodynamics of computation and information distance, in *Proceedings of the 25th Annual ACM Symposium on Theory of Computing* (ACM, New York, 1993), pp. 21–30.

[83] C. H. Bennett, P. Gács, M. Li, P. M. Vitányi, and W. H. Zurek, Information distance, IEEE Trans. Inf. Theory **44**, 1407 (1998).

[84] C. M. Caves, Entropy and information: How much information is needed to assign a probability? in *Complexity, Entropy, the Physics of Information*, Santa Fe Institute Studies in the Sciences of Complexity, Vol. VIII, edited by W. H. Zurek (Addison-Wesley, Redwood City, CA, 1990), pp. 91–115.

[85] C. M. Caves, Information and entropy, Phys. Rev. E **47**, 4010 (1993).

[86] Ä. Baumeler and S. Wolf, Free energy of a general computation, Phys. Rev. E **100**, 052115 (2019).

[87] C. H. Papadimitriou, *Computational Complexity* (John Wiley and Sons Ltd., New York, 2003).

[88] C. H. Bennett, Notes on landauer's principle, reversible computation, and maxwell's demon, Stud. Hist. Philos. Sci. B **34**, 501 (2003).

[89] C. Jarzynski, Hamiltonian derivation of a detailed fluctuation theorem, J. Stat. Phys. **98**, 77 (2000).

[90] M. Esposito, K. Lindenberg, and C. Van den Broeck, Entropy production as correlation between system and reservoir, New J. Phys. **12**, 013013 (2010).

[91] T. Sagawa, Thermodynamics of information processing in small systems, Prog. Theor. Phys. **127**, 1 (2012).

[92] J. Gemmer, M. Michel, and G. Mahler, *Quantum Thermodynamics: Emergence of Thermodynamic Behavior Within Composite Quantum Systems*, Vol. 657 of Lecture Notes in Physics, (Springer, Berlin, 2004).

[93] M. Hutter, *Universal Artificial Intelligence: Sequential Decisions Based on Algorithmic Probability* (Springer Science & Business Media, New York, 2004).

[94] S. Rathmanner and M. Hutter, A philosophical treatise of universal induction, Entropy **13**, 1076 (2011).

[95] R. J. Solomonoff, A formal theory of inductive inference. Part I, Inf. Cont. **7**, 1 (1964).

[96] J. Rissanen, A universal prior for integers and estimation by minimum description length, Ann. Stat. **11**, 416 (1983).

[97] M. Hutter, On the existence and convergence of computable universal priors, in *Proceedings of the 14th International Conference on Algorithmic Learning Theory (ALT'03)*, Vol. 2842 (Springer, Berlin, 2003), p. 298.

[98] J. Schmidhuber, The new ai: General & sound & relevant for physics, in *Artificial General Intelligence* (Springer, Berlin, 2007), pp. 175–198.

[99] J. Schmidhuber, Algorithmic theories of everything, arXiv:quant-ph/0011122.

[100] M. P. Mueller, Law without law: From observer states to physics via algorithmic information theory, Quantum **4**, 301 (2020).

[101] K. Tadaki, A generalization of Chaitin's halting probability ω and halting self-similar sets, Hokkaido Math. J. **31**, 219 (2002).

[102] C. S. Calude and M. A. Stay, Natural halting probabilities, partial randomness, and zeta functions, Inf. Comput. **204**, 1718 (2006).

[103] K. Tadaki, A statistical mechanical interpretation of algorithmic information theory: Total statistical mechanical interpretation based on physical argument, J. Phys.: Conf. Ser. **201**, 012006 (2010).

[104] G. Chaitin, On the length of programs for computing finite binary sequences, J. Assoc. Comput. Mach. **13**, 547 (1966).

[105] M. Hutter, Algorithmic complexity, Scholarpedia **3**, 2573 (2008).

[106] D. Wolpert and A. Kolchinsky, The thermodynamics of computing with circuits, New J. Phys. (2020), doi: 10.1088/1367-2630/ab82b8.

[107] A. Kolchinsky and D. H. Wolpert, Dependence of dissipation on the initial distribution over states, J. Stat. Mech. (2017) 083202.

[108] T. M. Cover and J. A. Thomas, *Elements of Information Theory* (John Wiley & Sons, New York, 2012).

[109] D. H. Wolpert, A. Kolchinsky, and J. A. Owen, A space-time tradeoff for implementing a function with master equation dynamics, Nat. Commun. **10**, 1727 (2019).

[110] S. Deffner and C. Jarzynski, Information Processing and the Second Law of Thermodynamics: An Inclusive, Hamiltonian Approach, Phys. Rev. X **3**, 041003 (2013).

[111] P. M. Vitányi, Conditional Kolmogorov complexity and universal probability, Theor. Comput. Sci. **501**, 93 (2013).