Mathematical Theory of Thermodynamics of Computation

Ming Li*
University of Waterloo

Paul M.B. Vitányi†
CWI and University of Amsterdam

Abstract

We investigate a new research area: we are interested in the ultimate thermodynamic cost of computing from x to y. Other than its fundamental importance, such research has potential implications in miniaturization of VLSI chips and applications in pattern recognition.

It turns out that the theory of thermodynamic cost of computation can be axiomatically developed. Our fundamental theorem connects physics to mathematics, providing the key that makes such a theory possible. It establishes optimal upper and lower bounds on the ultimate thermodynamic cost of computation.

By computing longer and longer, the amount of dissipated energy gets closer to these limits. In fact, one can trade time for energy: there is a provable time-energy trade-off hierarchy. The fundamental theorem also induces a thermodynamic distance metric. The topological properties of this metric show that neighborhoods are sparse, and get even sparser if they are centered on random elements. The proofs use Symmetry of Information in a basic way.

These notions also find an application in pattern recognition. People have been looking without success for an objective notion of cognitive distance to account for the intuitive notion of 'similarity' of pictures. Thermodynamic considerations lead to a recursively invariant notion of cognitive distances. It turns out that the thermodynamic distance is a universal cognitive distance which discovers all effective features used by any cognitive distance whatsoever.

---

*Supported in part by NSERC operating grant OGP-046506. Address: Computer Science Department, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. Email: mli@math.waterloo.edu

†Partially supported by NWO through NFI Project ALADDIN and NSERC International Scientific Exchange Award ISE0125663. Address: Centrum voor Wiskunde en Informatica, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands. Email: paulv@cwi.nl

1 Introduction

Computers can be regarded as engines that dissipate energy in order to process information. The ultimate limits of miniaturization of computing devices, and therefore the speed of computation, are governed by unavoidable heat increase through energy dissipation. Such limits have already been reached by current high density electronic chips [18]. Therefore, the question of how to reduce the energy dissipation of computation determines future advances in computing power.

In the early fifties, J. von Neumann [22] thought that a computer operating at temperature T must dissipate at least $kT \ln 2$ Joule per elementary bit operation (about 3×10^{-21} J at room temperature), where k is Boltzmann's constant. Around 1960, R. Landauer [14] more thoroughly analyzed this question and concluded that it is only 'logically irreversible' operations that must dissipate energy. An operation is *logically reversible* if its inputs can always be deduced from the outputs. Erasure of information is not reversible. Erasing each bit costs $kT \ln 2$ energy, when computer operates at temperature T. Solidly based on principles of physics, we develop a mathematical theory of the thermodynamic cost of computation.

• Firstly, the minimum thermodynamic cost of a computation is the sum of the energy involved in the providing (inverse erasure) the extra bits required in the course of a computation plus the destroying (erasure, [5]) of the generated garbage bits. This corresponds to the nonreversible part of the computation, and according to Landauer's principle only this nonreversible part of computation dissipates heat. We axiomatize this in terms of effective computations. Our "Fundamental Theorem" gives tight upper and lower bounds on the ultimate limits of the thermodynamic cost of effective computations, and makes a full theory of thermodynamics of computation possible.

• It has been stated before on the evidence of physical analogies that slow computations may dissipate less energy—like slower moving billiard balls in water generate less friction, [5]. We mathematically prove this statement to be true: there is a proper time-energy trade-off hierarchy of diminishing energy costs using increasing time of computation. Essentially, like in real life, garbage (like disposable information) needs to be compressed before it is destroyed, and this costs time.

• An *effective* distance is a distance which can be computed by a Turing machine. To compute an effective distance we have to spend some minimal thermodynamic cost, the effective *thermodynamic distance.* Thermodynamic distance is symmetric and induces a distance metric. We analyze the topological properties of this metric. This topology is sparse: each d-ball contains at most 2^d elements. (Compare this with a 1-ball around each $x \in \{0,1\}^n$ contains n elements in $\{0,1\}^n$ for Hamming distance.) The more random an object is, the less elements of the same size there are in a d-ball around it; if it is completely random then this number of elements is about $2^{d/2}$. Finally, in each set of size d almost all pairs of elements have distance $2d$ (which is also the maximum if the set is recursively enumerable).

• Given two pictures, are they similar? Answering such question is all what Pattern Recognition is about. Whatever we mean by picture similarity or picture distance is the first fundamental question that must be dealt with in pattern recognition. For example, Hamming distance is way off between positive and negative prints of the same image: in this case, Hamming distance is the largest, while the pictures are *cognitively* close to human eyes.

This is a question about cognition. Up till now no objective measure for cognitive distance has been found [23]. Intuitively, the minimal cognitive distance between two objects corresponds to the minimal amount of work involved in transforming one object into the other—by brain or computer.

The thermodynamic cost measure readily induces a mathematical theory for a recursively invariant notion of *cognitive distance* that was seemingly undefinable for the long history of pattern recognition. We show that the class of cognitive distances contains a universal cognitive distance, which turns out to be the thermodynamic distance. This universal cognitive distance minorizes *all* cognitive distances: if two pictures are d-close under some cognitive distance, then they are $O(d)$-close under this universal cognitive distance. That is, it discovers all effective feature similarities between two objects.

1.1 Related Work

There is a large body of proposals for effective physical realization of (almost) energy free reversible computing. Among others, this has been analyzed with respect to bistable magnetic devises for reversible copying/canceling of records in [14] and Brownian computers [12], for Turing machines and Brownian enzymatic computers [3, 4, 6], with respect to reversible Boolean circuits by [9], for molecular (billiard ball) computers by [21], Brownian computing using Josephson devices in [16], quantum mechanic computers in [1, 2, 17] and notably by R. Feynman [7, 8]. All these models seem mutually simulatable. For background information, see [5].

In the last three decades there have been many partial precursors and isolated results to the complete mathematical theory developed in this paper. However, it is the formulation of our *Fundamental Theorem* $E(x,y) \approx K(x|y) + K(y|x)$ that provides the key to the theory of thermodynamics of computation. Technically, this theorem is also nontrivially stronger than, and implies all, previous results on this issue which are comparable. Informally, $E(x,y)$ is the optimal thermodynamic cost of computing y from x, and $K(x|y)$ is the length of the shortest effective description of x given y.

At least in [5] Kolmogorov complexity was used in the analysis of reversible computing. In [19] a Kolmogorov complexity based metric for picture similarity was proposed (which is too complex), without clear justification or further results. One of us, stimulated by that paper, proposed (but did not publish) the proper definition $K(x|y) + K(y|x)$ of universal cognitive measure (presented in this paper) in 1988, but did not obtain any further results on it. With respect to cognitive distance, we are not aware of further comparable work: all previous work involves ad hoc approaches, and no objective measure has been proposed [23].

The closest in spirit, and most important stimulation, is the work of W. Zurek [25]. Since he does not provide a formal model, and charges costs in different ways in different places, we need to interpret his work in our model to obtain a proper comparison with our results. He established that the ultimate thermodynamic cost of erasure of a record x, provided the shortest program of length $K(x)$ for x is given, has an upper bound of $K(x)$ (units of $kT \ln 2$). Since we charge both for the provided bits and for the erased bits, this says $E(x, \epsilon) \leq 2K(x)$. Moreover, he gives a lower bound on the thermodynamic cost of computing y from x of $K(x|y)$. In our terminology this is $E(x,y) \geq K(x|y)$. He gives a thermodynamic distance assuming that $K(x|y) + K(y|x)$ bits are provided, which also have to be erased. This shows $E(x,y) \leq 2K(x|y) + 2K(y|x)$ (his information metric).

2 Thermodynamics of Computation

Many physical realizations of reversible computing devices, dissipating almost no energy in theory, have been proposed (see references in Section 1.1). For example, one basic issue is that almost energy free copying of records, and cancelling of one record with respect to an identical record provided it is known that they are identical, is physically realizable. Such operations are logically reversible, and their energy dissipation free execution gives substance to the idea that logically reversible computations can be performed with zero energy dissipation.

According to currently accepted physical viewpoints that the unavoidable thermodynamic cost, incurred by a computation which replaces input x by output y, is at least the number of irreversibly erased bits, each unit counted as $kT \ln 2$.

Complementary to this idea, if such a computation uses initially irreversibly provided bits apart from input x, then they must be accounted the same cost as that for irreversible erasure. Namely, providing extra records from external media can be viewed as irreversible inverse erasure: to get rid of them we have to irreversibly erase them. Because of the reversibility of the computation, we can also argue by symmetry. Namely, suppose we run a reversible computation starting when memory contains input x and additional record p, and ending with memory containing output y and additional garbage bits q. Then p is irreversibly provided, and q is irreversibly deleted. But if we run the computation backward, then the roles of x, p and y, q are simply interchanged.

Thus, we consider the following axioms as the codification of abundant concrete evidence in the form of a formal basis on which to develop a theory of computation thermodynamics in the manner of statistical mechanics or thermodynamics.

Axiom 1 Reversible computations do not incur any thermodynamic cost.

Axiom 2 Irreversibly provided and irreversibly deleted bits in a computation incur unit cost each.

Axiom 3 In a reversible computation which replaces input x by output y, the input x is not irreversibly provided and the output y is not irreversibly deleted.

Axiom 4 All physical computations are effective.

We emphasize that the first three axioms are solidly based on principles of physics. Axiom 4 is simply a form of *Church's Thesis*: the notion of physical computation coincides with effective computation which coincides with the formal notion of Turing machines computation.

We will be talking about the ultimate limits of energy dissipation by computation. Since these limits will be expressed in the number of bits in the irreversibly provided records and the irreversibly erased records, by Axioms 1–3, we consider compactification of records. Rather as in analogy of garbage collection by a garbage truck: the cost is less if we compact the garbage before we throw it away.

The ultimate compactification which can be effectively exploited is expressed in terms of Kolmogorov complexity. This is a recursively invariant concept, and to express the ultimate limits *no other notion will do*. Consequently, this mundane matter of energy dissipation of physical computation is *unavoidably* linked to, and expressed in, the pristine theoretical notion of Kolmogorov complexity.

2.1 The Invariant Notion of Thermodynamic Cost

We need the following notion. The Kolmogorov complexity, [13, 26, 15], of x is the length of the shortest effective description of x. Formally, this can be defined as follows. Let $x, y, z \in \mathcal{N}$, where $\mathcal{N}$ denotes the natural numbers and we identify $\mathcal{N}$ and $\{0,1\}^*$ according to the correspondence $(0, \epsilon), (1, 0), (2, 1), (3, 00), (4, 01), \ldots$. Hence, the length $|x|$ of x is the number of bits in the binary string x. Let $T_1, T_2, \ldots$ be a standard enumeration of all Turing machines, let $\phi_1, \phi_2, \ldots$ be the enumeration of corresponding partial recursive functions. So T_i computes ϕ_i. Let $|T_i|$ be the length of self-delimiting code for T_i. Let $< \cdot >$ be a standard invertible effective bijection from $\mathcal{N} \times \mathcal{N}$ to $\mathcal{N}$.

Definition 1 *The* Kolmogorov complexity *of x given y (for free) is*

$$K(x|y) = \min\{|p| + |T_i| : \phi_i(< p, y >) = x, p \in \{0,1\}^*, i \in \mathcal{N}\}.$$

Axioms 1-4 leads to the definition of the thermodynamic cost of a computation as the number of bits we added plus the number of bits we erased in computing one string from another. Let $R_1, R_2, \ldots$ be a standard enumeration of reversible Turing machines, [3], and let $\Psi = \psi_1, \psi_2, \ldots$ be the corresponding standard enumeration of partial recursive functions.

Definition 2 *Let ψ be a function computed by a reversible Turing machine. The* thermodynamic cost $E_\psi(x,y)$ *of computing y from x is*

$$E_\psi(x,y) = \min\{|p| + |q| : \psi(< x, p >) =< y, q >\}.$$

We denote the class of all such cost functions by $\mathcal{E}$.

We call an element of $\mathcal{E}$ a *universal thermodynamic cost function*, if for all $\psi \in \Psi$,

$$E_{\psi_0}(x,y) \leq E_\psi(x,y) + c_\psi,$$

for all x and y, where c_ψ is a constant which depends on ψ but not on x or y. Standard arguments from the theory of Turing machines show the following.

Lemma 1 *There is a universal thermodynamic cost function in $\mathcal{E}$. Denote it by ψ_0.*

Proof. In [3] a universal reversible Turing machine U is constructed. The function ψ_0 is the function computed by U. □

Two such universal (or optimal) functions ψ_0 and ψ_0' will assign the same thermodynamic cost to a computation apart from an additive constant term c which is *independent* of x and y. We select a reference universal function ψ_0 and define the *thermodynamic cost* $E(x,y)$ of computing y from x as

$$E(x,y) \equiv E_{\psi_0}(x,y).$$

In physical terms this cost is in units of $kT \ln 2$, where k is Boltzmann's constant, T is the absolute temperature in degrees Kelvin, and ln is the natural logarithm.

Because the computation is reversible, this definition is *symmetric*: we have $E(x,y) = E(y,x)$.

2.2 Ultimate Thermodynamic Cost of Computing y from x

Now let us consider a general computation which outputs string y from input string x. We want to know the minimum thermodynamic cost for such computation. This leads to the following theorem, which forms the basis of our theory.

Theorem 1 (Fundamental theorem) *Up to an additive logarithmic term*[1]*,*

$$E(x,y) = K(x|y) + K(y|x).$$

[1]Which is $O(\min\{K(K(y|x)|y), K(K(x|y)|x)\}) = O(\log \min\{K(y|x), K(x|y)\})$. It has been shown, [10], that for some x of each length n we have

$$\log n - \log\log n \leq K(K(x)|x),$$

and for all x of length n we have

$$K(K(x)|x) \leq \log n + 2\log\log n.$$

Proof. We prove first an upper bound and then a lower bound.

Claim 1 $E(x,y) \le K(y|x) + K(x|y) + 2[K(K(y|x)|y) + K(K(x|y)|x)]$.

Proof. We start out the computation with programs p, q, r. Program p computes y from x and $|p| = K(y|x)$. Program q computes the value $K(x|y)$ from x and $|q| = K(K(x|y)|x)$. Program r computes the value $K(y|x)$ from y and $|r| = K(K(y|x)|y)$. The computation is as follows.

1. Use p to compute y from x producing garbage bits $g(x,y)$.

2. Copy y, and use one copy of y and $g(x,y)$ to reverse the computation to x and p. Now we have p, q, r, x, y.

3. Copy x, and use one copy of x and q to compute $K(x|y)$ plus garbage bits.

4. Use $x, y, K(x|y)$ to dovetail the running of all programs of length $K(x|y)$ to find s, a shortest program to compute x from y. Doing this, we produce more garbage bits.

5. Copy s, and reverse the computations in Steps 4, 3, canceling the extra copies and all garbage bits. Now we have p, q, r, s, x, y.

6. Copy y, and use this copy to compute the value $K(y|x)$ from r and y producing garbage bits.

7. Use $x, y, K(y|x)$, to dovetail the running of all programs of length $K(y|x)$ to obtain a copy of p, the shortest program to compute y from x, producing more garbage bits.

8. Delete a copy of p and reverse the computation of Steps 7, 6 cancelling the superfluous copy of y and all garbage bits. Now we are left with x, y, r, s, q.

9. Compute from y and s a copy of x and cancel a copy of x. Reverse the computation. Now we have y, r, s, q.

10. Erase s, r, q.

We started out with additional shortest programs p, q, r apart from x. We have thermodynamically erased the shortest programs s, q, r, where $|s| = K(x|y)$, leaving only y. This proves the claim. $\square$

Note that all bits supplied in the beginning to the computation, apart from input x, as well as all bits thermodynamically erased at the end of the computation, are *random* bits. This is because we supply and delete only shortest programs, and a shortest program p satisfies $K(p) \ge |p|$, that is, it is maximally random.

Claim 2 $E(x,y) \ge K(y|x) + K(x|y)$.

Proof. To compute y from x we must be given a program to do so to start out with. By definition the shortest such program has length $K(y|x)$.

Assume the computation from x to y produces $g(x, y)$ garbage bits. Since the computation is reversible we can compute x from y and $g(x, y)$. Consequently, $|g(x, y)| \geq K(x|y)$ by definition [25]. To end the computation with y alone we therefore must thermodynamically erase $g(x, y)$ which is at least $K(x|y)$ bits. $\square$

Together Claims 1, 2 prove the theorem.
$\square$

Erasing a record x is actually a computation from x to the empty string ϵ. Hence its thermodynamic cost is $E(x, \epsilon)$, and given by a corollary to Theorem 1.

Corollary 1 *Up to a logarithmic additive term, the thermodynamic cost of erasure is* $E(x, \epsilon) = K(x)$.

2.3 Trading Time for Energy

In order to erase a record x, Corollary 1 actually requires us to have, apart from x, a program p of length $K(K(x)|x)$ for computing $K(x)$, given x. The precise bounds are $K(x) \leq E(x, \epsilon) \leq K(x) + 2K(K(x)|x)$. This optimum is not effective, it requires that p be given in some way. But we can use the same method as in the proof of Theorem 1, by compressing x using some time t. Let $K^t(x)$ be the length of a shortest program x_t^* which can be constructed from x in t steps.

Theorem 2 (Thermodynamic cost of effective erasure) *If t is a time bound such that $K(t|x) = O(1)$, then erasing an n bit record x can be done at thermodynamic cost $K^t(x)$ bits,*

Proof. Initially we have input x in memory.

1. Use x to compute t. Copy t and reverse the computation. Now we have x and t.

2. Use t to dovetail the running of all programs of length less than x to find the shortest one halting in time t with output x. This is x_t^*. The computation has produced garbage bits $g(x, x_t^*)$. Copy x_t^*, and reverse the computation to obtain x erasing all garbage bits $g(x, x_t^*)$. Now we have x, x_t^*, t in memory.

3. Compute t from x, cancel one copy of t, and reverse the computation. Now we have x, x_t^* in memory.

4. Erase x using x_t^* by the standard method, and then erase x_t^* irreversibly.

□

Effective compression methods are given in [20, 24, 11]. For example [11], if L is a context-free language, then each x in L can be compressed logarithmically in polynomial time. Note that thermodynamic cost in this way introduces another interesting measure on languages. Languages that dissipate less energy can be accepted faster, theoretically, than those languages that dissipate a lot of energy (which slows down the computation).

Corollary 2 *A lower bound on the thermodynamic cost of erasure is given by:*

$$E(x, \epsilon) \geq \lim_{t \to \infty} K^t(x) = K(x).$$

Essentially, by spending more time we can reduce the thermodynamic cost of erasure of x_t^* to its absolute minimum. In the limit we spend the optimal value $K(x)$ by erasing x^*, since $\lim_{t \to \infty} x_t^* = x^*$. This suggests the existence of a tradeoff hierarchy between time and energy. The longer one computes, the less energy one spends. This can be formally proved.

Let $E^t(x, y)$ be the minimum energy computing from x to y in time t. Formally,

$$E^t(x, y) = \min_{p,q \in \mathcal{N}} \{|p| + |q| : \psi_0(< x, p >) =< y, q > \text{ in} \leq t(|x|) \text{ steps}\}.$$

Theorem 3 (Energy-time tradeoff hierarchy) *For each y and each large n and $b > 3n/b + O(\log n)$, there is a string x of length n, and $t_1(n) < t_2(n) < \ldots < t_m(n)$, such that, for $m = n/b$,*

$$E^{t_1}(x, y) > E^{t_2}(x, y) > \ldots > E^{t_m}(x, y).$$

Proof. For simplicity, here we will only give the proof for $y = \epsilon$. Given n, we will construct a string x satisfying the requirements of the theorem. String x will be constructed in m steps, and x will contain m blocks $x_1, x_2, \ldots, x_m$ each of length $b = n/m$. The idea is to make these blocks harder and harder to compress. Define, for $1 \leq k \leq m$,

$$t_k(n) = 2^{kn}.$$

In our construction, we will enforce the following things:

- All m blocks can be compressed iff given enough time. Precisely, x_k can be compressed to constant size given $t_{k+1}(n)$ time, but x_k cannot be compressed at all given $t_k(n)$ time, where $n = |x|$.
- No "collective compression". *I.e.*, if x_k cannot be compressed in time t then $x_k \ldots x_m$, as a single string, cannot be compressed in time t either. In the construction, we will use only prefixes from strings in set S_k which contains strings that are not compressible at step k.

The Construction.

Step 0.

Let S_0 be the set of all strings of length n, and $t_0(n) = 0$.

Step $k+1$ (for $0 \leq k < m$).

Assume that the first k blocks $x_1, \ldots, x_k$ of x are already constructed and we have $|S_k| \geq 2^{n-kb-2k}$, where S_k contains strings of length $n - kb$ which cannot be computed from programs of size less than $n - kb - 2k$ in time $t_k(n)$. We now construct x_{k+1} using strings in S_k. Let s be the string of length b such that

$$|\{s' : ss' \in S_k\}| \geq 2^{n-(k+1)b-2k}. \tag{1}$$

Such s exists (Claim 3). Set $x_{k+1} := s$.

We now construct S_{k+1} from S_k. Let $S'_k = \{s' : x_{k+1}s' \in S_k\}$. We have $|S'_k| \geq 2^{n-(k+1)b-2k}$ by Equation 1. Simulate each of the programs of length less than $n-(k+1)b-2k-1$ for $t_{k+2}(n)/2$ steps. Let S_{k+1} contain all strings s of length $n - (k+1)b$ such that $s \in S'_k$ and s is not an output of above simulations. We have $|S_{k+1}| \geq 2^{n-(k+1)b-2(k+1)}$.

End of Construction.

Claim 3 *There is a string s of length b such that*

$$|\{s' : ss' \in S_k\}| \geq 2^{n-(k+1)b-2k}.$$

Proof. If this claim is not true, the number of elements in S_k must be less than

$$2^b 2^{n-(k+1)b-2k} = 2^{n-kb-2k},$$

contradiction. □

Claim 4 *For each $k = 1, \ldots, m$, block x_k can be generated by a constant sized program in time $\frac{1}{n}t_{k+1}(n)$,*

Proof. To prove the claim, we notice that we can construct S_k in $\sum_{i=1}^{k} t_i(n)$ steps. This is less than $\frac{1}{2n}t_{k+1}(n)$ steps. Then we can find x_k in less than $\frac{1}{2n}t_{k+1}(n)$ steps. In total, this is less than $\frac{1}{n}t_{k+1}(n)$ steps. Thus in time $\frac{1}{n}t_{k+1}(n)$, given k, we can compute x_k by a constant size program. □

Claim 5 *Recall $b = n/m$.*

1. $K^{\frac{1}{n}t_{k+1}}(x) \leq n - kb \pm O(\log n)$.

2. $n - kb - 2k \pm O(\log n) \le K^{t_{k+1}}(x)$.

Proof. We first prove Item 1, $K^{\frac{1}{n}t_{k+1}}(x) \le n - kb \pm O(\log n)$. Since $\frac{1}{n}t_{k+1}(n)$ is sufficient to simulate above construction to the end of step k by Claim 4, this produces the first k blocks of x. Thus $K^{\frac{1}{n}t_{k+1}}(x) \le n - kb \pm O(\log n)$.

We next show that Item 2, $K^{t_{k+1}}(x) \ge n - kb - 2k - O(\log n)$, by contradiction. Suppose $K^{t_{k+1}}(x) < n - kb - 2k - p\log n$, p is a constant large enough. Then in $t_{k+1}(n)$ time, with a program of size $n-kb-2k-p\log n$ can print x. Thus a program of size $n - kb - 2k - (p-1)\log n$ can print $x_{k+1}\dots x_m \in S_{k+1}$ in time $t_{k+1}(n)$. This contradicts to the construction, since the strings in S_{k+1} are not compressible in time t_{k+1} by more than $2(k+1) + O(1)$ bits. □

With time $\frac{1}{n}t_{k+1}(n)$, we can compress x to size $n - kb \pm O(\log n)$. Thus by a reversible computation, using methods developed earlier in this paper, we need only to erase $n - kb \pm O(\log n)$ bits in order to erase x in $t_{k+1}(n)$ time. Precisely,

Claim 6 *If in t steps we can compute from s to s' and from s' to s, then $E^{O(t)}(s,\epsilon) \le |s'|$.*

Proof. We show how to erase s by erasing $|s'|$ bits in $O(t)$ time.

1. Reversibly compute s' from s, with garbage $g(s,s')$, using t steps.
2. Copy s^*, then reverse the computation of Item 1, absorbing the garbage bits $g(s,s^*)$, using at most $O(t)$ steps.
3. Reversibly compute from s' to s, with garbage $g(s',s)$; then cancel a copy of s, using at most $O(t)$ time.
4. Reverse the computation of Item 3, absorbing the garbage bits $g(s',s)$, then remove s' irreversibly, using at most time $O(t)$.

In total, above erasing procedure uses less than $O(t)$ steps and erases s' bits. Thus $E^{O(t)}(s) \le |s'|$. □

Thus, by Claim 5(1) and Claim 6, we have

$$E^{t_{k+1}}(x,\epsilon) \le n - kb \pm O(\log n). \tag{2}$$

On the other hand, with time $t_k(n)$, by Claim 5(2), we have

$$K^{t_k}(x) \ge n - (k-1)b - 2(k-1) \pm O(\log n).$$

Thus, by a time-bounded version of Corollary 1, we must erase at least $n - (k-1)b - 2(k-1) \pm O(\log n)$ bits. That is,

$$E^{t_k}(x,\epsilon) \ge n - (k-1)b - 2(k-1) \pm O(\log n). \tag{3}$$

By Equations 2 and 3, and the assumption that $b > 3n/b + O(\log n)$, we conclude, for $1 \leq k < m$,

$$E^{t_k}(x, \epsilon) > E^{t_{k+1}}(x, \epsilon).$$

□

When $y = \epsilon$, Theorem 3 gives a time-energy tradeoff hierarchy for erasure.

2.4 Tough Guys Have Less Neighbors, Thermodynamically

A major property of $E(x, y)$ is its *symmetry*, $E(x, y) = E(y, x)$.

It is easily verified that $E : \mathcal{N} \times \mathcal{N} \rightarrow \mathcal{N}$ is a distance function. Up to a logarithmic additive term we have $E(x, y) \geq 0$ with equality only for $x = y$; $E(x, y) = E(y, x)$; and $E(x, z) \leq E(x, y) + E(y, z)$.

Definition 3 *We call $E(x, y)$ the* thermodynamic distance *between x and y.*

Random objects x have sparsely populated neighborhoods. Indeed, the following Theorem 4 says that if x of length n has complexity $K(x) \geq n$, then there are at most $2^{d/2}$ elements y of length n within thermodynamic distance d.

The more random a string is, the less number of strings of the same length are nearby. In fact, an object x must be compressible by d bits so that $K(x) \leq n - d$ before there can be about 2^d elements y of length n within thermodynamic distance d.

Theorem 4 (Topology of Randomness) *Let $x, y \in \mathcal{N}$ have length n. For each x the number of y's such that $E(x, y) \leq d$ is 2^α with*

$$\alpha = \frac{n + d - K(x)}{2} \pm O(\log n),$$

while $n - K(x) \leq d$. For $n - K(x) \geq d$ we have $\alpha = d \pm O(\log n)$.

Proof. Let $K(x) = n - \delta(n)$. In the remainder of the proof all (in)equalities involving complexities hold up to an $O(\log n)$ additional term.

($\geq$) We show that there are at least $2^{(d+\delta(n))/2}$ elements y such that Theorem 4 holds. Let $y = x^* z$ with $|z| = \delta(n)$ and x^* is the first program for x which we find by dovetailing all computations on programs of length less than n. We can retrieve z from y using at most $O(\log n)$ bits. There are $2^{\delta(n)}$ different such y's. For each such y we have $K(x|y) = O(1)$, since x can be retrieved from y using x^*. Now suppose we further divide $y = uw$ with $|u| = l/2$ and choose u arbitrary. Then, the total number of such y's increases to $2^{\delta(n)+l/2}$.

These choices of y must satisfy $E(x, y) \leq d$. Clearly, $K(y|x) \leq \delta(n) + l/2$. Moreover, $K(x|y) \leq l/2$ since we can retrieve x by providing $l/2$ bits. Therefore,

$$K(x|y) + K(y|x) \leq l/2 + \delta(n) + l/2.$$

Since the lefthand side has at most value d, the largest l we can choose is, up to the suppressed additional term $O(\log n)$, given by $l = d - \delta(n)$.

This puts the number of y's such that $E(x,y) \leq d$ at least at $2^{(\delta(n)+d)/2 \pm O(\log n)}$. (Since l must be nonnegative, we can at most choose $\delta(n) \leq d$, which gives a greatest number 2^d of y's for $\delta(n) = d$. This corresponds to Theorem 5)

($\leq$) Assume, to the contrary, that there are at least $2^{(d+\delta(n))/2+c}$ elements y such that Theorem 4 holds, with c some large constant. Then, for some y,

$$K(y|x) \geq \frac{d+\delta(n)}{2} + c. \tag{4}$$

By assumption

$$K(x) = n - \delta(n), K(y) \leq n.$$

By symmetry of information [26]

$$K(x) + K(y|x) = K(y) + K(x|y),$$

and substituting we find

$$n + \frac{d-\delta(n)}{2} + c \leq n + K(x|y).$$

But this means that

$$K(x|y) \geq \frac{d-\delta(n)}{2} + c, \tag{5}$$

which, by Equations 4 and 5, contradicts

$$K(x|y) + K(y|x) \leq d.$$

□

If we consider strings of all lenghts, then there is a fixed approximate number of y's within optimal thermodynamic cost d of x. This property will be very useful in defining cognitive distance.

Theorem 5 (Topology) *Let $x, y \in \mathcal{N}$. For all x the number of y's satisfying $E(x,y) \leq d$ is at least $\Omega(2^d)$ and at most 2^d.*

Proof. ($\leq$) Assume the converse. Then for some x and d, the number of y's for which $E(x,y) \leq d$ exceeds 2^d. Consequently, there is a y_0 such that $K(y_0|x) \geq d$. Since $E(x,y_0) \geq K(y_0|x) + K(x|y_0)$ and $K(x|y_0) > 0$ we have a contradiction.

($\geq$) Consider the set of y's of the form xp^R, where p^R is the reversal of a self-delimiting program p. Then $K(x|y) = O(1)$, $K(y|x) = K(p|x) + O(1)$. Hence,

$$d \geq E(x,y) \geq K(x|y) + K(y|x) = K(p|x) + O(1).$$

For any fixed x there are at least $\Omega(2^d)$ distinct p's which satisfy $K(p|x)+O(1) \leq d$, and hence there are also that many y's. $\square$

It turns out that in every set of low complexity almost all elements are far away from each other. The complexity $K(S)$ of a set is the length of the shortest binary program that enumerates S and then halts.

Theorem 6 (Diameter of sets) *Let S be a set with $K(S) = O(\log d)$ and with $|S| = 2^d$. Almost all pairs of elements $x, y \in S$ have distance $E(x,y) \geq 2d$, up to an additive logarithmic term.*

Proof. By the conditions in the theorem, for all $x \in S$ we have $K(x) \leq d \pm O(\log d)$. There are $2^d - 2^{d-c}$ elements x in S which have complexity $K(x) \geq d - c$. Let S_c be the set of these x's. For any x in S_c, there are at least $2^d - 2^{d-c+1}$ elements $y \in S_c$ such that $K(y|x) \geq d - 2c$. By symmetry of information,

$$K(x) + K(y|x) = K(y) + K(x|y) \pm O(\log d).$$

Hence, for $c = \log d$ we have $K(x|y) \geq d - O(\log d)$. That is, by Theorem 1, we have

$$E(x,y) \geq 2d - O(\log d),$$

for at least

$$(2^d - 2^{d-\log d})(2^d - 2^{d-2\log d}) = 2^{2d}(1 - \frac{1}{d} - \frac{1}{d^2} + \frac{1}{d^3}).$$

pairs $x, y \in S$. $\square$

A similar statement can be proved for the distance of an x (possibly outside S) to the majority of elements y in S. If $K(x) \geq n$, then for almost all $y \in S$ we have $E(x,y) \geq n + d - O(\log dn)$.

3 Cognitive distance

Let us identify digitized black-and-white pictures with binary strings. There are many distances defined for binary strings. For example, the Hamming distance and the Euclidean distance. Such distances are sometimes okay. For instance, taking a binary picture, and you change a few bits on that picture, then the changed and unchanged pictures have small Hamming or Euclidean distance, and they do look similar. However, this is not always the case. The positive and negative prints of a photo have the largest possible Hamming and Euclidean distance, yet they look similar in our eyes. Another example, if we shift a picture one bit to the right, again the Hamming distance may increase by a lot, but the two pictures remain similar.

Many books on pattern recognition try to define picture similarity. But up to now, no definition is even close to a satisfactory definition of cognitive similarity.

Indeed, we do not even know what we mean by cognitively similar. This is an intuitive notion which has no mathematical meaning. In fact, the intuitive 'cognitive distance' can hardly be regarded as one definite measure since different people see things differently. One man's meat may be another man's poison.

Given two pictures x and y, a first thought to us was to define the distance from x to y as $K(y|x)$. But then one immediately notices that this is not symmetric. A few years ago, one of the authors and Zurek [25] proposed measure $K(y|x)+K(x|y)$. But why? The solution is in thermodynamics of computation.

To make our approach relevant for human cognition we will assume that human eyes (and brains) process information like a computer. Such an assumption suffices for artificial intelligence and pattern recognition in which we deal only with artificial eyes and computer vision. We shall equate the work done to compare x with y with the thermodynamic cost of comparing x with y. $E(x,y)$ is the minimum thermodynamic cost which *any* effective process needs to spend in comparing x with y. It is a recursively invariant objective notion and cannot be improved.

We are ready to define *cognitive distance.* A distance measure must be nonnegative, symmetric, and satisfy the triangle inequality. This is not sufficient since a distance measure like $d(x,y)=1$ for all $x \neq y$ must be excluded. For a distance to make sense, it must be able to discriminate minority of strings that are similar to a given string x from majority of strings that are not similar to x.

We consider only effectively computable distances. Our definition will not be suitable to situations when a distance function is not computable. For human vision, it may be unclear if man can compute more than computers. But in pattern recognition and computer vision, our assumption is simply a consequence of universally accepted notions of effectiveness. This leads to Item 4 in the definition of cognitive distance.

The thermodynamic cost of the shortest effective distance between x and y is $E(x,y)$. According to Theorem 5, for each x, the number of elements that are within distance d to x is at most 2^d. We want cognitive distance to be thermodynamically computable, and hence set this number as an upper bound on the number of elements in a d-ball around x. This requires Item 5 in the definition of cognitive distance.

Definition 4 *Cognitive distance*, D, is a total function, $\mathcal{N} \times \mathcal{N} \rightarrow \mathcal{N}$, such that

1. $\forall x, y, D(x,y) \geq 0$, with equality holding only for $x = y$;
2. Symmetry: $\forall x, y, D(x,y) = D(y,x)$;
3. Triangle Inequality: $\forall x, y, z, D(x,y) \leq D(x,z) + D(z,y)$;
4. for each x, the set $\{y : D(x,y) \leq d\}$ is recursively enumerable; and
5. for each x, $|\{y : D(x,y) \leq d\}| \leq 2^d$.

Can the thermodynamic distance metric be expected to behave as a cognitive metric? Let $\mathcal{D}$ be the class of cognitive distances.

Definition 5 $\Gamma \in \mathcal{D}$ *is a* universal cognitive distance *if for each* $D \in \mathcal{D}$, *we have* $\Gamma(x,y) = O(D(x,y))$ *for all* $x, y \in \mathcal{N}$.

Theorem 7 (Optimal cognitive distance) *The function* $E(x,y)$ *is a universal cognitive distance function.*

Proof.

We first prove that E is a cognitive distance. Items (1)—(3) of the definition are trivially satisfied since E is a distance function. Item (4) is satisfied since E is enumerable. Item (5) is satisfied by Theorem 5.

Secondly, it needs to be proved that E is optimal. For any $D \in \mathcal{D}$, by Item 5 of Definition 4, we have $|S| \leq 2^d$, where $S = \{y : D(x,y) \leq d\}$. Given x, we can recursively enumerate the set S, and identify y by its index in S. Therefore, $K(y|x) \leq d + O(1)$. Since D is symmetric, we derive similarly $K(x|y) \leq d + O(1)$. Hence,

$$E(x,y) = K(x|y) + K(y|x) \leq 2D(x,y) + O(1).$$

□

In this sense, $E(x,y)$ is the *optimal* cognitive distance. If x and y are d-close under any cognitive distance D, then they are also $2d$-close under E. That is, E will account for similarity according to *any* cognitive measure. This distance is the ultimate effective similarity criterion.

References

[1] P.A. Benioff. Quantum mechanical Hamiltonian models of discrete processes that erase their histories: applications to Turing machines. *Int. J. Theoret. Physics*, 21:177–202, 1982.

[2] P.A. Benioff. Quantum mechanical Hamiltonian models of computers. *Ann. New York Acad. Sci.*, 480:475–486, 1986.

[3] C.H. Bennett. Logical reversibility of computation. *IBM J. Res. Develop.*, 17:525–532, 1973.

[4] C.H. Bennett. Dissipation-error tradeoff in proofreading. *BioSystems*, 11:85–90, 1979.

[5] C.H. Bennett. The thermodynamics of computation—a review. *Int. J. Theoret. Physics*, 21:905–940, 1982.

[6] C.H. Bennett and R. Landauer. The fundamental physical limits of computation. *Scientific American*, pages 48–56, July 1985.

[7] R.P. Feynman. Simulating physics with computers. *Int. J. Theoret. Physics*, 21:467–488, 1982.

[8] R.P. Feynman. Quantum mechanical computers. *Optics News*, 11:11–20, 1985. Reprinted in: *Foundations of Physics*, 16(1986), 507-531.

[9] E. Fredkin and T. Toffoli. Conservative logic. *Int. J. Theoret. Physics*, 21(3/4):219–253, 1982.

[10] P. Gács. On the symmetry of algorithmic information. *Soviet Math. Dokl.*, 15:1477–1480, 1974. Correction, Ibid., 15:1480, 1974.

[11] A. Goldberg and M. Sipser. Compression and ranking. *SIAM J. Comput.*, 20:524–536, 1991.

[12] R.W. Keyes and R. Landauer. Minimal energy dissipation in logic. *IBM J. Res. Develop.*, 14:152–157, 1970.

[13] A.N. Kolmogorov. Three approaches to the quantitative definition of information. *Problems Inform. Transmission*, 1(1):1–7, 1965.

[14] R. Landauer. Irreversibility and heat generation in the computing process. *IBM J. Res. Develop.*, 5:183–191, 1961.

[15] M. Li and P.M.B. Vitányi. Kolmogorov complexity and its applications. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, chapter 4, pages 187–254. Elsevier and MIT Press, 1990.

[16] K. Likharev. Classical and quantum limitations on energy consumption on computation. *Int. J. Theoret. Physics*, 21:311–326, 1982.

[17] N. Margolus. Parallel quantum computation. In W.H. Zurek, editor, *Complexity, Entropy and the Physics of Information*, pages 273–287. Addison-Wesley, 1991.

[18] C. Mead and L. Conway. *Introduction to VLSI Systems.* Addison-Wesley, 1980.

[19] D. Schweizer and Y. Abu-Mostafa. Kolmogorov metric spaces. Manuscript, Computer Science, 256-80, California Institue of Technology, Pasadena, 1988.

[20] J. Storer. *Data Compression: Method and Theory.* Computer Science Press, 1988.

[21] T. Toffoli. Reversible computing. In *7th International Colloquium Automata, Languages, and Programming*, volume 85 of *Lecture Notes in Computer Science*, pages 632–644. Springer-Verlag, 1980.

[22] J. von Neumann. *Collected Works*, volume 5. MacMillan, 1963.

[23] S. Watanabe. *Pattern recognition.* John Wiley & Sons, 1985.

[24] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate encoding. *IEEE Trans. Inform. Theory*, IT-24:530–536, 1978.

[25] W.H. Zurek. Thermodynamic cost of computation, algorithmic complexity and the information metric. *Nature*, 341:119–124, 1989.

[26] A.K. Zvonkin and L.A. Levin. The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. *Russian Math. Surveys*, 25(6):83–124, 1970.