Fundamental energy cost of finite-time computing

Michael Konopik,1 Till Korten,2 Eric Lutz,1 and Heiner Linke3

1 Institute for Theoretical Physics I, University of Stuttgart, D-70550 Stuttgart, Germany
2 B CUBE - Center for Molecular Bioengineering, Technische Universität Dresden, D-01307 Dresden, Germany
3 NanoLand and Solid State Physics, Lund University, S-22100 Lund, Sweden

The fundamental energy cost of irreversible computing is given by the Landauer bound of \( kT \ln 2 / \text{bit} \). However, this limit is only achievable for infinite-time processes. We here determine the fundamental energy cost of finite-time irreversible computing within the framework of nonequilibrium thermodynamics. Comparing the lower bounds of energy required by ideal serial and parallel computers to solve a problem of a given size in a given finite time, we find that the energy cost of a serial computer fundamentally diverges with increasing problem size, whereas that of a parallel computer can stay close to the Landauer limit. We discuss the implications of this result in the context of current technology, and for different degrees of parallelization and amounts of overhead. Our findings provide a physical basis for the design of energy efficient computers.

There is wide agreement that Moore’s law regarding the exponential growth of the number of components in integrated circuits [1] is coming to an end [2, 3]. One of the main physical reasons that prevents further miniaturization is unavoidable heat generation [2, 3]. A much-improved energy efficiency of computing is therefore a key requirement for any ‘More-than-Moore’ technology [4]. The fundamental limits to the work cost and the heat dissipation of computing are given by the Landauer bound of \( kT \ln 2 / \text{bit} \). Because it occurs in finite time, such a nonequilibrium process is necessarily accompanied by the dissipation of an amount of work \( W_{\text{dis}} \) into the environment [18].

We base our analysis on the following assumptions (Fig. 1): (i) A computing problem of size \( N \) should be solved in finite time \( T \). In order to stay within this time limit, (ii) an ideal serial computer dynamically adapts its processing frequency (time per operation \( \tau_p \); Fig. 1a left), whereas (iii) an ideal parallel computer adapts the number \( n \) of processors, keeping constant its processing frequency (time per operation \( \tau_p \); Fig. 1a right). We argue that these assumptions are well justified: While the available time is not exactly fixed, there is usually a limit on how long calculations can be run. For example, to be useful, the weather forecast for the next day should not run more than a few hours. Moreover, modern processors implement a number of mechanisms, such as dynamic frequency and voltage scaling [16, 17], as well as deactivation of cores using deep-sleep states [22], that make them behave in a manner very similar to the assumptions (i)-(iii) made above.

Let us first consider a single computation of finite duration \( \tau \). Because it occurs in finite time, such a nonequilibrium process is necessarily accompanied by the dissipation of an amount of work \( W_{\text{dis}} \) into the environment [18]. The energetic cost of a finite-time, logically irreversible bit operation may hence be written as a generalized Landauer bound,

\[
W(\tau) = kT \ln 2 + W_{\text{dis}}(\tau),
\]
FIG. 1. Assumptions for ideal serial and parallel computers. a) Schematic illustration of the three main assumptions: (i) The total time \( T \) available to solve a given problem requiring \( N \) computing operations is limited; (ii) an ideal serial computer (left, blue) reduces the time per operation \( \tau_s \) with increasing problem size \( N \); (iii) an ideal parallel computer (right, red) increases the number of processors \( n \) proportional to the size \( N \) in order to keep the time per operation \( \tau_p \) constant. b) Leading \( 1/\tau \) behavior (solid line) of the energy consumption of a single operation of duration \( \tau \) near equilibrium, Eq. (1), and its effects for serial (blue) and parallel (red) computers. Higher-order behavior for small \( \tau \) (dotted line) is discussed in the Supplemental Material [29].

where \( W_{\text{dis}}/T \geq 0 \) is the nonequilibrium entropy produced during the process [18]. Close to equilibrium, the dissipated work is predicted to take the generic form \( W_{\text{dis}} = a/\tau \) (Fig. 1b), for both classical [23] and quantum [27] dynamics, where \( a \) is an energy efficiency constant that depends on the system. Interestingly, such a \( 1/\tau \) behavior has been observed in experiments [6, 8, 28].

Equation (1) reduces to the Landauer limit for slow computation processes \( \tau \to \infty \) and illustrates that in general more work per operation is required for fast operations and, in turn, more heat is dissipated.

In view of Eq. (1), the total work cost associated with the solution of a computing problem that requires \( N \) bit operations within the finite time \( T \) is given by,

\[
W_{\text{tot}}(N, \tau) = NW(\tau) = NkT \ln 2 + NW_{\text{dis}}(\tau),
\]

where \( \tau = \tau(T) \) is in general a function of \( T \). The scaling of the dissipative term with the system size \( N \) depends on the type of computing considered. It may be con-
Fundamentally, one may argue that the lowest possible value for $a$ is quantum mechanically given by Planck’s constant, $h = 6.610^{-34}$ Js $\approx 1.6 \cdot 10^{-13}$ kTs (at room temperature) [15], 13 orders of magnitude lower than the above value for current electronic computers (Fig. 3) solid lines). However, to the best of our knowledge, no physical system has been proposed that would reach such a small value for $a$. In recent experimental studies of the thermodynamics of finite-time operations, much higher values have been found. The lowest measured value known to us is $a = 1.1 \cdot 10^{-29}$ Js, reported for memory operations using molecular nanomagnets [12], corresponding to $a \approx 10^{-6}$ kTs at the operation temperature of $T \approx 1$ K. For comparison, from experiments with optical traps one can estimate $a \approx 2$ kTs $= 8 \cdot 10^{-22}$ Js at room temperature [26].

Based on these insights it is illustrative to compare the fundamental energy cost of finite-time computing as a function of problem size $N$ for fully serial and fully parallel computers (Fig. 3). For a serial, electronic computer (blue dashed line) with representative $a = 2.5$ kTs, the finite-time energy cost is dominated by the term $aN/T$ in Eq. (3). A further increase in $N$ (corresponding to an increase in operation frequency $f_{op}N/T$ of a serial computer beyond the currently typical $f_{op} \approx 1$ GHz) thus leads to a continued increase in energy dissipation per operation. Given that thermal management is already now the limiting factor in processor design, this is not an option unless $a$ can be lowered, for example through transistor and circuit design. If, on the other hand, the quantum mechanical limit of $a \approx h$ were achievable for a serial computer, then the term $aN/T$ in Eq. (3) would become noticeable, compared to the frequency independent Landauer limit, only once $N$ exceeds $10^{13}$ operations, corresponding to $f_{op} \approx$ THz (blue line). By contrast, a fully (ideal) parallel computer does not increase its energy cost per operation (orange lines). For $a = 2.5$ kTs (Xeon Phi) and $\tau_p = 1$ s, the extrapolated energy cost per operation (orange dashed line) is only about one order of magnitude larger than the fundamental Landauer bound (orange solid line).

However, real-world algorithms are rarely completely parallelizable. Therefore, the ideal estimates, Eqs. (3) and (4), need to be refined. The impact of non-parallelizable parts of an algorithm on the speedup of parallel computing is commonly described by Amdahl’s law [19]. According to Amdahl [31], the time of the initial serial realization $T$ can be split into two contributions, a purely serial part $s$, that cannot be done by more than one processor at a time, and a parallel part $p$ that can, ideally, be split equally among all the used $n$ processors (Fig. 4 inset). We evaluate the energetic consequences of such a splitting for our ideal computers as follows: We assume that a given problem of size $N$ is comprised of a serial and parallel part, $N = N_s + N_p = sN + pN$. The total computation time is given by the sum of these two parts, $T = T_s + T_p$, where the serial part $T_s$ can be tuned
by adjusting the time per operation \( \tau_s \) and the parallel part \( \tau_p \) is solely controlled by the number of processors \( n \). We then optimize the combined energy cost function over \( \tau_p \) using the fixed total time constraint and obtain the minimal energy cost for partial parallelization \([29]\),

\[
\frac{W_{\text{tot}}^{\text{com}}}{N} = kT \ln 2 + \frac{a}{bT} \left( s\sqrt{bN} + \sqrt{p} \right)^2.
\]

(5)

It interpolates between the purely serial implementation \([3]\) (\( p = 0 \)) and the completely parallelizable processor \([4]\) (\( s = 0 \)). In particular, we observe that the quadratic energetic advantage of the parallel computer is weakened when the degree of parallelization is decreased (Fig. 4).

Another important aspect of real-world algorithms, that ought to be accounted for, is that of parallelization overhead. Parallelization indeed frequently requires the execution of additional overhead operations \( N_{\text{ove}} \). Usually, this overhead is a function of the number of processors used \([19]\). For concreteness and simplicity, we consider a linear overhead, \( N_{\text{ove}}(n) = cn \), that corresponds to the case where each processor requires a constant number of overhead operations (different overheads may be considered). Because of the constant \( T \) assumption, the overhead either means that each processor needs to work faster in order to compensate for the overhead (Fig. 4a inset), or that one might use a stronger degree of parallelization \( 1 > b' > b \). We shall assume that the maximal available parallelization is already used and that overhead may only be compensated by adjusting the calculation speed \( \tau_p \). We then obtain \([29]\),

\[
\frac{\tau_{\text{ove}}}{\tau_p} = \frac{\tau_p}{1 + N_{\text{ove}}(n)/N} = \frac{bT}{1 + bc}.
\]

(6)

Owing to the time dependence of the dissipated work in Eq. (1), the energy cost of the parallel execution not only increases with the number of additional operations \( N_{\text{ove}} \) but also because of the necessary increase in processing speed. As a result, we obtain from Eq. (1) the total energetic cost in the presence of a linear overhead \([29]\),

\[
\frac{W_{\text{tot}}^{\text{ove}}(N)}{N} = \left[ N + N_{\text{ove}}(n) \right] \frac{W(\tau_{\text{ove}})}{N} \left( 1 + \frac{N_{\text{ove}}(n)}{N} \right) \times \left[ kT \ln 2 + \frac{a}{bT} \left( 1 + \frac{N_{\text{ove}}(n)}{N} \right) \right].
\]

(7)

The overhead thus causes the parallel computer to be less efficient than the serial computer for small problem sizes. This is because the Landauer part adds a fixed cost to Eq. (7), while the dissipative part will only be dominant for large \( N \). However, the parallel implementation exhibits better scaling and becomes more energy efficient for larger problem sizes, even for large overhead (Fig. 4b). This result holds true as long as \( N_{\text{ove}}(n) \) scales better than \( n^{3/2} \) (or, equivalently, \( N^{3/2} \), since here \( n \propto N \)).

In conclusion, we have used insights from nonequilibrium thermodynamics to develop a general formalism to evaluate the fundamental energetic cost of finite-time computing. Our main result is that the finite-time energy cost per operation of a fully parallel computer is independent of problem size and can realistically operate close to the Landauer limit. This is in contrast to serial computers for which the finite-time energy cost per operation necessarily increases with problem size. We also provide a framework for including partial parallelization and the associated energy overhead into the analysis. For serial computers, the key limiting factor is the finite-time constant \( a \). To enable a drastic increase
in operation frequency without fundamentally necessary, prohibitive energy consumption, a needs to be strongly reduced below its current value of $a \approx 10^{-6}$ kTs in electronic computers. Whether physical systems are available that allow a smaller than the current record of $a \approx 10^{-6}$ kTs possibly down to a quantum limit of $a \approx h$, is an open question. At the same time, massively parallel computers can be realized in biological computing, such as DNA computing or network-based biocomputing (NBC), which use small DNA molecules or cytoskeletal filaments, respectively, as computing cores and memory. These are cheap to mass-produce and can be added to the computation in amounts matching the problem size. Both DNA computing and NBC have been estimated to be able to work very close to the Landauer limit per operation. As shown in our study, this finite-time energy cost is independent of problem size (Fig. 3). From the perspective of finite-time energy cost, biological computers thus offer a potentially large, fundamental advantage over electronic computers.

We acknowledge financial support from the German Science Foundation (DFG) (Contract No FOR 2724) and from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 732482 (Bio4Comp), and from the Knut and Alice Wallenberg Foundation (project 2016.0089).

[1] Moore, G. E. Cramming more components onto integrated circuits. *Electronics* **38**, 114–117 (1965).
[2] Theis, T. N. & Wong, H. S. P. The End of Moore’s Law: A New Beginning for Information Technology. *Computing in Science Engineering* **19**, 41–50 (2017).
[3] Waldrop, M. M. The chips are down for Moore’s law. *Nature* **530**, 144–147 (2016).
[4] Arden, W. et al. "More-than-Moore" White Paper, International Roadmap for Devices and Systems (IRDS) (2015).
[5] Landauer, R. Irreversibility and Heat Generation in the Computing Process. *IBM Journal of Research and Development* **5**, 183–191 (1961).
[6] Béruit, A. et al. Experimental verification of landauer’s principle linking information and thermodynamics. *Nature* **483**, 187–189 (2012).
[7] Orlov, A. O., Lent, C. S., Thorpe, C. C., Boechler, G. P. & Snider, G. L. Experimental test of Landauer’s principle at the sub-kB unit level. *Japanese Journal of Applied Physics* **51**, 06FE10 (2012).
[8] Jun, Y., Gavrilov, M. & Bechhoefer, J. High-precision test of Landauer’s principle in a feedback trap. *Phys. Rev. Lett.* **113**, 190601 (2014).
[9] Martini, L. et al. Experimental and theoretical analysis of Landauer erasure in nano-magnetic switches of different sizes. *Nano Energy* **19**, 108 – 116 (2016).
[10] Hong, J., Lambson, B., Dhuy, S. & Bokor, J. Experimental test of Landauer’s principle in single-bit operations on nanomagnetic memory bits. *Science Advances* **2**, e1501492 (2016).
[11] Yan, L. L. et al. Single-atom demonstration of the quantum Landauer principle. *PRL* **120**, 210601 (2018).
[12] Gaudenzi, R., Burzuri, E., Maegawa, S., van der Zant, H. S. J. & Luis, F. Quantum Landauer erasure with a molecular nanomagnet. *Nature Phys.* **14**, 565 (2018).
[13] Lutz, E. & Ciliberto, S. Information: From Maxwell’s demon to Landauer’s erasure. *Physics Today* **68**, 30–35 (2015).
[14] Lloyd, S. Ultimate physical limits to computation. *Nature* **406**, 1047 (2000).
[15] Meindl, J. D., Chen, Q. & Davis, J. A. Limits on Silicon Nanoelectronics for Terascale Integration. *Science* **293**, 2044–2049 (2001).
[16] Horvath, T., Abdelzaher, T., Skadron, K. & Liu, X. Dynamic Voltage Scaling in Multitier Web Servers with End-to-End Delay Control. *IEEE Transactions on Computers* **56**, 444–458 (2007).
[17] Cho, S. & Melhem, R. Corollaries to Amdahl’s law for energy. *IEEE Computer Architecture Letters* **7**, 25–28 (2008).
[18] Lebon, G. & Casas-Vásquez, D. J. J. Understanding Non-Equilibrium Thermodynamics (Springer, Berlin, 2008).
[19] Pacheco, P. An Introduction to Parallel Programming (Morgan Kaufmann, Burlington, 2007).
[20] Le Sueur, E. & Heiser, G. Dynamic voltage and frequency scaling: The laws of diminishing returns. In *Proceedings of the 2010 international conference on Power aware computing and systems*, 1–8 (2010).
[21] Samani, M. C. & Esfahani, F. S. A Review of Power Management Approaches Based On DVFS Technique in Cloud Data Centers. *Data Science Letters* **3**, 32–40 (2018).
[22] Rotem, E., Naveh, A., Ananthakrishnan, A., Weissmann, E. & Rajwan, D. Power-Management Architecture of the Intel Microarchitecture Code-Named Sandy Bridge. *IEEE Micro* **32**, 20–27 (2012).
[23] Weidlich, W. & Haag, G. Quasiadiabatic solutions of Fokker Planck equations with time-dependent drift and fluctuations coefficients. *Zeitschrift für Physik B Condensed Matter* **39**, 81–87 (1980).
[24] Sekimoto, K. & Sasa, S.-i. Complementarity relation for irreversible process derived from stochastic energetics. *Journal of the Physical Society of Japan* **66**, 3326–3328 (1997).
[25] Aurell, E., Mejía-Monasterio, C. & Muratore-Ginanneschi, P. Optimal protocols and optimal transport in stochastic thermodynamics. *Phys. Rev. Lett.* **106**, 250601 (2011).
[26] Proesmans, K., Ehrich, J. & Bechhoefer, J. Finite-time Landauer principle. *Phys. Rev. Lett.* **125**, 100602 (2020).
[27] Cavina, V., Mari, A. & Giovannetti, V. Slow dynamics and thermodynamics of open quantum systems. *Phys. Rev. Lett.* **119**, 050601 (2017).
[28] Ma, Y.-H., Zhai, R.-X., Chen, J., Sun, C. P. & Dong, H. Experimental test of the 1/τ-scaling entropy generation in finite-time thermodynamics. *Phys. Rev. Lett.* **125**, 210601 (2020).
[29] See supplemental material.
[30] Shao, Y. S. & Brooks, D. Energy characterization and instruction-level energy model of Intel’s Xeon Phi processor. In *International Symposium on Low Power Electronics and Design (ISLPED)*, 389–394 (2013).
[31] Amdahl, G. M. Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities. In
SUPPLEMENTAL MATERIAL

**Partially parallelizable algorithms.** The energy cost of partially parallelizable problems\(^5\) is derived from three observations: (i) the problem can be split into a serial and a parallel part, \(N = N_s + N_p = sN + pN\), (ii) corresponding to a total duration \(T = T_s + T_p\), where the respective serial and parallel time allocations, \(T_s\) and \(T_p\), can be freely chosen, except for the fixed total time constraint. Since we are interested in the optimal energy cost, it is important to further (iii) optimize the energy cost function over \(T_p\), respecting the time constraint. The computation times for a single operation, \(\tau_s\) and \(\tau_p\) (for notational simplicity, \(\tau_s\) denotes the time for the serial part, not the serial computation) are thus,

\[
T_s = \tau_s N_s \quad \text{and} \quad T_p = \tau_p N_p / n(N_p).
\]  

(S1)

The total energy cost follows from Eq. 1 as,

\[
W_{\text{tot}}^\text{com} = NkT \ln 2 + \frac{a}{\tau_s} N_s + \frac{a}{\tau_p} N_p
\]

\[
= N \left[ kT \ln 2 + \frac{s^2 Na}{T_s} + \frac{p^2 aN}{T_p n(N_p)} \right].
\]  

(S2)

Minimizing with respect to \(T_p\) using (iii), we obtain,

\[
\tau_p = \frac{aN[n + \sqrt{n(N_p)s}]^2}{n(N_p)T}.
\]  

(S3)

Inserting this result into (S2), we then find,

\[
W_{\text{tot}}^\text{com} = N \left( kT \ln 2 + \frac{a}{n(N_p)T} \left[ p + \sqrt{n(N_p)s} \right]^2 \right).
\]  

(S4)

Equation (S4) eventually follows with \(n(N_p) = bNp\).

**Parallelization overhead.** There are many possible kinds of overhead. We consider the simple linear form, which is linear in the number of processors and thus linear in the problem size for an ideal parallel computer. For simplicity, we assume that all available processors are used and therefore \(n(N)\) is fixed. The overhead is accordingly taken into account by modulating the computation speed. With \(N_g = N + N_{\text{ave}}(n)\) and \(\tau_p = N / n(N)\), the general time needed to solve the problem including the linear overhead is,

\[
T' = \tau' = \frac{N_g}{\tau_p n(N)} = \tau_p \left[ \frac{N}{n(N)} + \frac{N_{\text{ave}}(n)}{n(N)} \right].
\]  

(S5)

The given time constraint is \(T' = T = N\tau_p / n(N)\) for the overhead-free case. We thus have,

\[
\tau_p' = \frac{1}{1 + N_{\text{ave}}(n)/N}.
\]  

(S6)

The total energy cost is finally,

\[
W_{\text{tot}}^\text{ave} = N \left( 1 + \frac{N_{\text{ave}}(n)}{N} \right) \left[ kT \ln 2 + \frac{a}{bT} \left( 1 + \frac{N_{\text{ave}}(n)}{N} \right) \right].
\]  

(S7)

Equation (S7) can in principle be extended for any kind of overhead.

**Corrections to the 1/\(\tau\) behavior.** Equations (S5) and (S6) are derived using the leading-order finite-time correction to the quasistatic driving. In general applications, this limit may not be reached and it is thus of interest to consider the influence of higher orders \(W_{\text{intr}} \propto 1/\tau^m\), \(m > 1\), on the energetic bounds for serial and parallel computation. Higher orders lead to serial costs of the form \(W_{\text{ser}} / N \propto 1/\tau^m = N^m / T^m\). As a result, the larger the value of \(m\), the worse the energy scaling becomes. On the other hand, in the parallel case we have \(W_{\text{par}} / N \propto 1/\tau_p^m = 1/(bT)^m\), with \(b \leq 1\). Remarkably, the scaling is still independent of the number of operations \(N\). Hence, while there is no \(N\) dependence in the parallel case, the constant can be quite large compared to the Landauer bound, if \(b^m\) is very small. All in all, the leading-order comparison of ideal serial and parallel computation is, in a sense, a favorable comparison for the serial case, as higher order make the advantage of the (ideal) parallel computation more distinct.