

Proceedings

# Thermodynamic Computing: An Intellectual and Technological Frontier <sup>†</sup>

Todd Hylton 

Department of Electrical and Computer Engineering, University of California, San Diego, CA 92093, USA; thylton@ucsd.edu

<sup>†</sup> Conference Morphological, Natural, Analog and Other Unconventional Forms of Computing for Cognition and Intelligence (MORCOM), Berkeley, CA, USA, 2–6 June 2019.

Published: 10 June 2020



**Abstract:** Concepts from thermodynamics are ubiquitous in computing systems today—e.g., in power supplies and cooling systems, in signal transport losses, in device fabrication, in state changes, and in the methods of machine learning. Here we propose that thermodynamics should be the central, unifying concept in future computing systems. In particular, we suppose that future computing technologies will thermodynamically evolve in response to electrical and information potential in their environment and, therefore, address the central challenges of energy efficiency and self-organization in technological systems. In this article, we summarize the motivation for a new computing paradigm grounded in thermodynamics and articulate a vision for such future systems.

**Keywords:** thermodynamics; computation; thermodynamic computing; artificial intelligence; self-organization; evolution

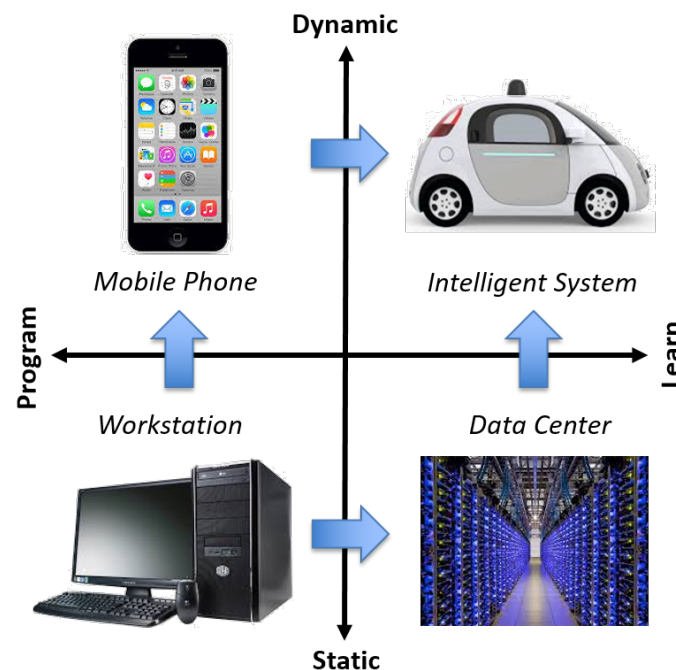
---

## 1. Introduction

The current computing paradigm, which is the foundation of much of the current standards of living that we now enjoy, faces fundamental limitations that are evident from several perspectives. In terms of hardware, devices have become so small that we are struggling to eliminate the effects of thermodynamic fluctuations, which are unavoidable at the nanometer scale. In terms of software, our ability to imagine and program effective computational abstractions and implementations are clearly challenged in complex domains like economic systems, ecological systems, medicine, social systems, warfare, and autonomous vehicles. Somewhat paradoxically, while avoiding stochasticity in hardware, we are generating it in software for various machine learning techniques at substantial computational and energetic cost. In terms of systems, currently 6% of global power generation is consumed by computing systems [1]—this astonishing figure is neither ecologically sustainable nor economically scalable. Economically, the cost of building next-generation semiconductor fabrication plants has soared past \$10 billion and therefore eliminated all but a few companies as the sources of future chips. However, even as we face these constraints, we are ~1000X above the Landauer limit [2] of thermodynamic efficiency, our computing systems possess far more computational potential than we can ever program, and our computers are many orders of magnitude less capable and less energy efficient than brains. All of these difficulties—device scaling, software complexity, adaptability, energy consumption, and fabrication economics—indicate that the current computing paradigm has matured and that continued improvements will be limited, even while we are still far from fundamental limits. If technological progress is to continue and corresponding social and economic benefits are to continue to accrue, we contend that the computing paradigm itself must be reconsidered.

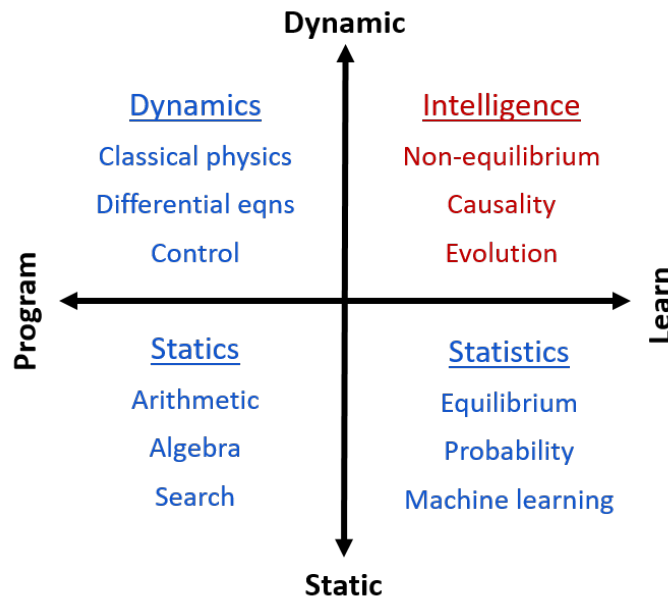
## 2. Motivation

To motivate our thesis, we begin with a comparison of computing applications illustrated as a “technology landscape” in Figure 1. The figure summarizes four application domains divided along two axes of characterization—programmed versus learned and static versus dynamic. Over the last several decades applications have progressed from *static-programmed* applications on desktop workstations (lower left) to, more recently, *dynamic-programmed* applications on mobile phones (upper left) and *static-learned* applications on large data centers (lower right). Today, we aspire to address *dynamic-learned* applications including robotics, self-driving cars, smart grids, etc. (upper right). These applications are profoundly difficult in large part because they must perform in complex, real world environments with high reliability, which require learning, adaptation and generalization in real time. We call this the domain of *intelligent systems*.



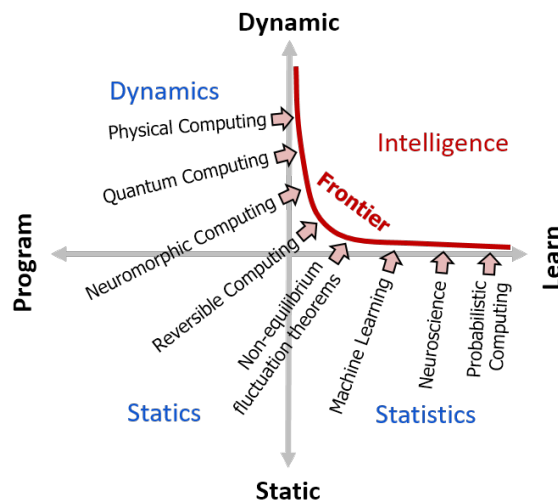
**Figure 1.** The technology landscape is divided into four application domains separated by two features: (1) programmed vs learned (horizontal axis) and (2) dynamic or static in time (vertical axis). A few decades ago applications were largely confined to the lower left corner as illustrated by the desktop workstation and applications like spreadsheets and word processors. More recently, advances in computing and communication technologies have made it possible to take programmed computation into the real world in the form of mobile phones (upper left) and to build large data centers capable of serving large, offline, machine learning applications (lower right). Tantalizing but as yet largely unrealized applications like driverless cars, smart cities, and robotics exist in dynamic environments that must be learned because they are too complex to program (upper right). We call this domain “intelligent systems”.

In Figure 2 we describe a “conceptual landscape” that parallels the technology landscape of Figure 1. Static-programmed applications correspond to concepts from *statics*; dynamic-programmed applications correspond to concepts from *dynamics*; static-learned applications correspond to concepts from *statistics*; and dynamic-learned applications correspond to concepts from the yet to be understood domain of *intelligence*. We emphasize that these assignments of terms are intended to illustrate differences and not to rigidly assign meaning; many applications will draw ideas from more than one of these domains.



**Figure 2.** The conceptual landscape is divided into the same four domains described in Figure 1. The lower left domain of *Statics* is primarily concerned with quantification and counting. The upper left domain of *Dynamics* is primarily concerned with systems that change deterministically with time and are not too complex. The lower right domain of *Statistics* is primarily concerned with correlations in data sets collected from “noisy” environments which are not changing in time. The upper right domain of *Intelligence* is considerably less well developed and understood than the other domains. We suppose that this understanding will emerge from work in non-equilibrium thermodynamics, causality and evolution.

We hypothesize that the challenges in computing today as well as its inability to approach fundamental limits are associated primarily with the conceptual challenges of the Intelligence domain in Figure 2. In Figure 3 we illustrate this gap of understanding as “frontier” that is well recognized and correspondingly addressed by many technical communities with roots in the other, better understood domains.



**Figure 3.** Intelligence stands at the frontier of our understanding, illustrated in the upper right corner of the same four domains used in Figures 1 and 2. Multiple approaches with roots in the other quadrants address this frontier, but a comprehensive understanding is yet to be achieved.

To summarize, we suppose that primary problem in computing today is that computers cannot organize themselves—they cannot address the domain of Intelligence as illustrated in Figures 1–3. Organization of computing systems today is left entirely to the engineers that design and program them. However, no matter their skill, humans will never be able to program machines with trillions of degrees of freedom to realize either (1) even a small fraction of their potential applications or (2) optimal computational efficiency in a complex, dynamic task. We can see these limitations today in our computing infrastructure consuming massive amounts of power in largely repetitive tasks. The whole universe apparently organizes itself, but our computers do not. What are we missing? We offer the following observations.

- Thermodynamics is universal. There is no domain of science, engineering or everyday life in which thermodynamics is not a primary concern. As far as we know, ideas from thermodynamics are the only ones that are universally applicable.
- Thermodynamics is the problem in computing today. As described in Section 2, challenges of power consumption, device fluctuations, fabrication costs, and organization are fundamentally thermodynamic.
- Thermodynamics organizes everything. While many would argue that this idea is unproven, many have speculated on the role of thermodynamics in the organization of systems as diverse as life [3] and the basic physics of Newton and Einstein [4].
- Thermodynamics is temporal. The second law of thermodynamics is the only explanation for the existence of time. If our objective is to build technologies in the dynamic-learned-intelligence domain of Figures 1–3 then an incorporation of this understanding should be a primary concern.
- Thermodynamics is efficient. All systems spontaneously evolve to achieve thermodynamic efficiency. This is particularly evident in living systems, but also in human built technologies and organizations, which are almost always under pressure to improve their efficiency. The most likely explanation for this observation, of course, is that thermodynamics drives their evolution.
- Thermodynamics is ancillary to the current computing paradigm. Thermodynamics is currently viewed as a practical constraint in the implementation of a programmed state machine and memory technology, rather than as a fundamental concept. In computing systems today, physics ends at the level of the device and is replaced by bits, gates and algorithms thereafter. We suppose that physics should extend throughout the computing system, as it evidently does in all natural systems.
- Electronic systems are well-suited for thermodynamic evolution. The energy, time and length scales of today's basic computational elements (e.g., transistors) approach that of biological elements (e.g., proteins). Instead of struggling to prevent the spontaneous evolution of systems of these devices (i.e., eliminate fluctuations), we should learn how to guide their evolution in the solution of problems that we specify at high levels.

### 3. Vision

In this section, we offer a high-level vision for a future of computing grounded in thermodynamics. More details of this vision are available in a recently released report [5].

A Thermodynamic Computer (TC) is an open, evolving, thermodynamic system that globally organizes to efficiently transport energy provided by external potentials, locally adapts to improve transport efficiency with time, equilibrates constantly with a thermal bath, and can be constrained by a user to evolve a solution to a task. Figure 4 illustrates the basic concept of a TC system. A conventional computing system hosts a TC, which is connected directly to an environment of electrical and information potentials. Humans can direct the evolution of the TC through the host computer by programming constraints that influence the connections to the environment and the evolution of the TC. The TC can also provide feedback to the conventional computing system; for example, it may evolve a representation of its environment that can be used as an input.

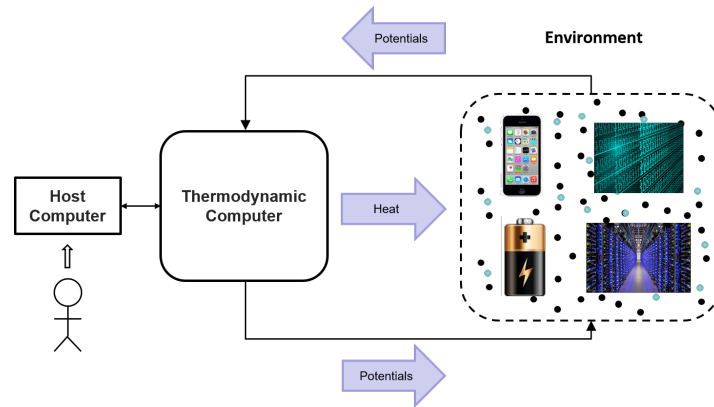


Figure 4. Conceptual schematic for a Thermodynamic Computing System.

Figure 5 illustrates in more detail the evolution of the TC component of the system in Figure 4. The TC architecture is illustrated as an array of nodes connected in a network. Each node may assume a variety of states/functions and the connections in the network may adapt as the system evolves. The TC is connected to an external environment that represents the “problem” to be solved as a collection of potentials that will drive the evolution of the network. A programmer may fix the states/functions of some of the nodes and connections to constrain the evolution of the network and/or to help define the problem. Unconstrained nodes fluctuate as the network evolves to a low energy configuration that is effective at transporting energy among the external potentials. The network may stop fluctuating if the environment is stable and converge upon a “solution”, but may also repeatedly fluctuate and stabilize to varying degrees if the environment is constantly changing. The key idea driving the evolution of the system is the positive feedback between dissipation and fluctuation. Energy dissipation associated with high energy configurations (i.e., poor energy transport across the TC) creates fluctuations that sample different configurations. If these fluctuation-generated configurations decrease dissipation (i.e., improve energy transport) they will be preferentially stabilized. In this way the TC evolves to move energy efficiently as it equilibrates with its environment. The environment includes a thermal reservoir, which plays an active role in the evolution of the TC by providing fluctuations.

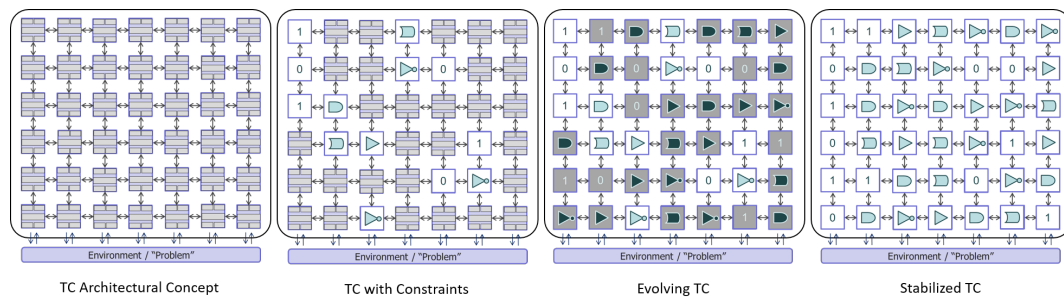


Figure 5. Conceptual schematic for the evolution of a Thermodynamic Computing System. (Left) The TC architecture is conceived as an array of nodes connected in a network. Each node may assume a variety of states/functions and the connections in the network may adapt as the TC evolves. The TC is connected to an external environment that represents the “problem” to be solved as a collection of potentials that will drive the evolution of the network. (Center Left) A programmer may fix the states/functions of some of the nodes (shown in white) and connections in order to constrain the evolution of the network and/or to help define the problem. (Center Right) Unconstrained nodes (shown in grey) fluctuate as the network evolves to a low energy configuration that is effective at transporting energy among the external potentials. (Right) The network may stop fluctuating if the environment is stable and converge upon a “solution”, but may also repeatedly fluctuate and stabilize to varying degrees if the environment is constantly changing.

To put this vision in a larger context, Figure 6 divides computing into domains according to their relationship to fluctuation scales. Spatial and temporal fluctuation scales are estimated in terms of thermal energy ( $kT$ , i.e., Boltzmann constant times temperature) and corresponding free electron quantum coherence times and lengths. We divide the computing paradigm into three qualitatively different domains that we label as “Classical”, “Thermodynamic”, and “Quantum”. In the classical domain, fluctuations are small compared to the smallest devices in a computing system (e.g., transistors, gates, memory elements), therefore separating the scales of “computation” and “fluctuation” and enabling abstractions such as device state and the mechanization of state transformation that underpin the current computing paradigm. In the quantum domain, fluctuations in space and time are large compared to the computing system. While the classical domain avoids fluctuations by “averaging them away”, the quantum domain avoids them by “freezing them out”. In the thermodynamic domain, fluctuations in space and time are comparable to the scale of the computing system and/or its devices. This is the domain of non-equilibrium, mesoscale thermodynamics, cellular operations, neuronal plasticity, genetic evolution, etc.—i.e., it is the domain of self-organization and the evolution of life. This is the domain that we need to understand in order to build technologies that operate near the thermodynamic limits of efficiency and spontaneously self-organize, but paradoxically it is also the domain that we assiduously avoid in our current classical and quantum computing efforts. Both classical and quantum computing struggle to scale—smaller devices in the classical domain and larger systems in the quantum domain—as they approach the thermodynamic domain.

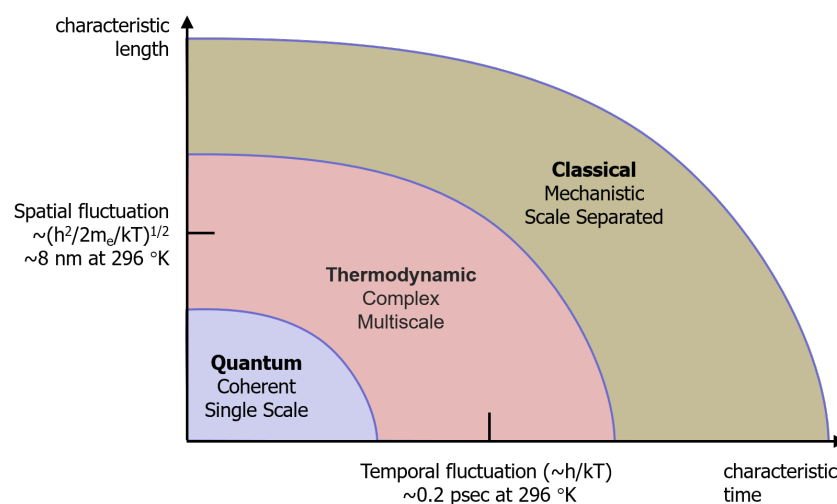


Figure 6. Three Major Domains of Computing by fluctuation scale.

Many of the concepts described in this vision have been captured in a model system called Thermodynamic Neural Network [6]. In this model networks of nodes connected by adapting, weighted edges self-organize to transport charge from internal and external potentials while equilibrating with a thermal reservoir. The evolution of selected networks can be visualized as videos (Figures 4–8, 10–13 and 15 in [6]) that clearly illustrate complex multiscale dynamics and self-organization to improve transport efficiency among external potentials.

#### 4. Conclusions

As we approach the limits of the current computing paradigm, we advocate a re-examination of that paradigm and a shift to a new paradigm in which thermodynamics and associated physical concepts become its foundations. We argue that we are currently facing a technological and intellectual frontier of understanding. We present a vision for a new technology that we call Thermodynamic Computing as we cross that frontier.

**Funding:** This research received no external funding.

**Acknowledgments:** The author thanks the Computing Community Consortium of the Computing Research Association for its support of the Thermodynamic Computing workshop, which influenced many of the ideas presented here.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Gent, E. To make smartphones sustainable, we need to rethink thermodynamics. *New Sci.* **2020**. Available online: <https://www.newscientist.com/article/mg24532733-300-to-make-smartphones-sustainable-we-need-to-rethink-thermodynamics/> (accessed on 11 March 2020)
2. Landauer, R. Irreversibility and heat generation in the computing process. *IBM J. Res. Dev.* **1961**, *5*, 183–191.
3. Schrödinger, E. *What is Life?: The Physical Aspect of the Living Cell*; Cambridge University Press: Cambridge, UK, 1944.
4. Verlinde, E. On the Origin of Gravity and the Laws of Newton. *J. High Energy Phys.* **2011**, *2011*, 29.
5. Hylton, T.; Conte, T.; DeBenedictis, E.; Ganesh, N.; Still, S.; Strachan, J.P.; Williams, R.S.; Alemi, A.; Altenberg, L.; Crooks, G.; et al. *Thermodynamic Computing: Report Based on a CCC Workshop Held on January 3–5, 2019*; Technical Report; Computing Community Consortium: Washington, DC, USA, 2019.
6. Hylton, T. Thermodynamic Neural Network. *Entropy* **2020**, *22*, 256, doi:10.3390/e22030256.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).