Association for Computing Machinery
Advancing Computing as a Science & Profession

Ubiquity, an ACM publication
February, 2011

Ubiquity Symposium

What is Computation?

Biological Computation

by Melanie Mitchell
Computer Science Department, Portland State University
and
Santa Fe Institute

Editor's Introduction

In this thirteenth piece to the Ubiquity symposium discussing 'What is computation?' Melanie Mitchell discusses the idea that biological computation is a process that occurs in nature, not merely in computer simulations of nature.

Peter J. Denning
Editor

Ubiquity Symposium

What is Computation?

Biological Computation

by Melanie Mitchell

What Is Meant By "Biological Computation"?

In this article, the term *biological computation* refers to the proposal that living organisms themselves perform computations, and, more specifically, that the abstract ideas of information and computation may be key to understanding biology in a more unified manner. It is important to point out that the study of biological computation is typically not the focus of the field of *computational biology*, which applies computing tools to the solution of specific biological problems. Likewise, biological computation is distinct from the field of *biologically-inspired computing*, which borrows ideas from biological systems such as the brain, insect colonies, and the immune system in order to develop new algorithms for specific computer science applications. While there is some overlap among these different meldings of biology and computer science, it is only the study of biological computation that asks, specifically, if, how, and why living systems can be viewed as fundamentally computational in nature.

It seems obvious to many that biological systems "compute". A brief perusal of the literature reveals an abundance of book and paper titles such as "Computation in Neurons and Neural Systems" (Eeckman, 1994), "Immunology as Information Processing" (Hofmeyr & Forrest, 2000), "Information Processing in Social Insects" (Detrain et al., 1999), "Computation in Cells and Tissues" (Paton et al., 2004), "Learning from Bacteria about Natural Information Processing" (Ben-Jacob, 2009), and "Evidence for Complex, Collective Dynamics and Emergent, Distributed Computation in Plants" (Peak et al., 2004), to name just a few. It is important to note that these authors are all talking about *computing* as it takes place in the biological system itself, not in any simulation of the system. The idea here is that *biological computation* is a process that occurs in nature, not merely in computer simulations of nature.

This widespread interest in biological computation reflects a strong intuition that the notions of *information* and *information processing* are building blocks that will shed new light on how living systems operate and the common principles underlying their operation. Biology has long suffered from being a science of specific details rather than abstractions and general laws. The theory of evolution serves as one grand organizing principle, but biology still lacks a general theory of how adaptive functionality emerges from large collections of individual, decentralized components. How, for example, do insect colonies, composed of thousands to millions of individual insects, collectively make decisions and accomplish complex tasks that seem to require the communication and processing of colony-wide information? How does the immune system, composed of trillions of cells and molecular components circulating in the body, collectively recognize patterns of infection and other organism-wide conditions, and collectively decide how to mount an appropriate response? How do the hundreds of billions of neurons in the brain work together to continually make sense of and respond to the opportunities and threats of the environment in real-time? These questions cry out for a unified theory involving information, communication, and computation. However, there is as yet little agreement among the people who ask such questions as to exactly what constitutes "information" and "computing" in such systems, and on whether common information-processing mechanisms might be identified across these disparate disciplines.

The notion that the framework of computation might be useful for biologists is not new. In the 1940s and 1950s, many of the earliest computer scientists were thinking about computation as a phenomenon that went beyond mechanical and electronic automata. Alan Turing's formalization of computation as "Turing machines" arose, in part, from his speculations about operations in the human brain. John von Neumann's studies of self-reproduction in machines sought to capture the "logic" of (in essence, the abstract algorithms governing) biological self-reproduction. Norbert Wiener, in developing the so-called "Cybernetics" movement, aimed at constructing a general theory of "control and communication" in animals and machines. However, this early work turned out to be much more influential in computer science—particularly in artificial intelligence—than in biology. The lack of influence for biologists was due to a number of factors, but perhaps the most important were the limited amount that was known about the mechanisms of biological information processing, and the lack of a formal theory of computation that was applicable to those mechanisms and that went beyond imprecise metaphors. These problems remain today in our understanding of biological computation.

Traditional Versus Biological Computing

In my view, to make the notion of biological computation more precise in any particular system, we need to answer the following questions (Mitchell, 2009):

- How is information represented in the system?
- How is information read and written by the system?
- How is it processed?
- How does this information acquire function (or "purpose", or "meaning")?

These questions are relatively straightforward to answer for traditional computing systems, which are based on the Turing-machine and von Neumann-style architectures. In such systems information is represented as collections of bits, which represent components of programs or data. Information is read and written by a central processing unit via "fetch" and "write" operations to and from memory. Information is processed via logical operations performed by the CPU. This information acquires "meaning" via the interpretations of human users. (The thorny question of what "meaning" is and how we humans perceive it is beyond the scope of this article, though, in my view, it will be a key question for understanding biological computation. Here I am using the term "meaning" informally and assuming that if a human uses a computer to perform a computation, then the information resulting from that computation has some meaning for the human.)

The answers are much less straightforward in the case of biological systems. One good illustration of this is the process of task allocation in ant colonies. In an ant colony, ants take on different specialized tasks, such as foraging for food, nest maintenance, patrolling the nest, and refuse-sorting. Ants do not always stick to the same task; instead they often switch tasks as needed, depending on the current state of their environment. Each ant has a limited view of the global nest environment, limited contact with other ants, and no central "controller" issuing commands as to what task to pursue. How do ants decide what task to take on at a given time so that the colony as a whole will have an optimal allocation of workers to various tasks, given that the optimal allocation continually changes?

According to the work of biologist Deborah Gordon and colleagues on Red Harvester ants (Gordon, 1999; Greene & Gordon, 2007), individual ants use not only their own local environmental observations but also observations of their *interaction rates* with other ants as a way to gain information about the global state of the environment and to decide what task to take on. In a colony, there is much random coming and going of ants, whether they are currently inactive or pursuing particular tasks. Thus an individual ant moving around will be likely to encounter other individuals pursuing a variety of task assignments, and can identify what task another ant is pursuing by direct contact with chemicals on the other ant's body. Suppose that at a given time there has been a disturbance to the nest and thus an increased need for nest-maintenance workers. Individuals who directly observe the disturbance may use that information to switch to maintenance tasks, but other individuals can do a form of statistical sampling, in which they measure the rates of their own encounters with maintenance workers versus encounters with ants performing other tasks, and, from their samples of these rates, decide whether to take on a new task. It seems that information about the global state of the colony is, in part, represented in the *rates* of interaction sampled by individual ants. Of course it is not any single decision by an individual ant that results in the colony achieving an ever-changing yet globally optimum allocation of tasks; rather it is a collective *computation* done via the statistics of thousands to millions of ants that results in this adaptive resource allocation.

Task switching in ant colonies is just one example of behavior for which notions of information and computation can provide insight. This collective computation clearly has a *function*, in that it results in an adaptive advantage for the colony that carries it out. However, in general it is difficult to formally characterize the functions being computed by such collectives, the "algorithms" by which they are computed, or even the *model* of computation one should use to understand the characteristics and limits of such computations.

This example also highlights a number of profound differences between the mechanisms of information processing in traditional computers and in living systems. In traditional computers, information is digital and of a single type (e.g., bits), fixed (unchanging unless explicitly rewritten), centralized (i.e., fetched to a central location to be processed), and typically noise-free. In living systems, by contrast, information is often analog in nature and of different types (e.g., reflected in real-valued rates of interaction or concentrations of different substances), continually changing, decentralized (distributed over large areas and over large numbers of system components), and fundamentally noisy.

The processing of information in traditional computers is centralized (i.e., performed by a CPU), typically serial, deterministic, exact, and terminating (i.e., there is an unambiguous final result of the computation). On the other hand, in biology, information processing is massively parallel, stochastic, inexact, and on-going, with no clean notion of a mapping between “inputs” and “outputs”.

Whereas traditional computing systems typically require synchronization in many aspects of their processing, biological systems often operate with asynchronous components. Traditional computing systems require components to be reliable, with very low error probabilities, whereas biological systems operate with unreliable components that are subject to frequent failures. In traditional computer science, the notions of *universal computation* and *programmability* are fundamental, whereas the relevance of these concepts for biological computing is unclear.

The differences I have sketched here, among others, present a challenge for adapting the traditional frameworks of computer science to formulate a theory of biological computation.

Is Computing a “Natural Science”?

In his article "Computing is a Natural Science" (Denning, 2007), Peter Denning describes how the relationship between computer science and the natural sciences has evolved over the past 70 years. Beginning in the 1940s, computer science began to be a provider of tools for natural scientists, allowing them to solve equations and analyze data. Later, computer science was able to provide methods for actually carrying out science via simulation and computational “experiments”. More recently, computing has increasingly been a source of inspiration to scientists looking at information processing as a natural phenomenon. In Denning’s view, this last stage has allowed computing to be counted as a natural science since it studies not only artificial systems but also natural phenomena, namely “informational processes” in nature.

I would agree that computing is, or at least has the potential to be, a natural science, which may eventually be as foundational for biology as physics has been for chemistry. That is, the science of computing may someday contribute the conceptual building blocks upon which is built a more unified understanding of biological phenomena. To do this, however, a number of fundamental advances must be made, not just in our understanding of biology but also in the development of theoretical computer science itself.

A theory of biological computation will need to incorporate new notions of information and information processing that are relevant for the characteristics of biological systems sketched above. A key strength of current-day computer science is its ability, via abstractions such as formal languages, abstract machines, programmability and model checking, measures of computability and complexity, and clear notions of *levels of description* to describe and design functionality, independent of specific mechanisms. Something like this is clearly needed for biology, which often focuses not on abstract structure or function but on detailed biochemical mechanisms, perhaps missing the forest for the trees. However, given the very different characters of traditional and biological information processing that I sketched above, it seems that new kinds of abstractions need to be formulated, for example, to make rigorous the apparent analogies among information processing in ant colonies, the immune system, the brain, genetic regulatory networks, and other systems (Mitchell, 2009).

What would these abstractions look like? It is not yet clear, though the development of frameworks that can capture functional aspects of the complex dynamics of networks (biological and otherwise) is an active area of research (e.g., Cardelli & Corrado, 2009; Klemm & Bornholdt, 2005, to name just two of the myriad recent examples on this subject). Interestingly, the field of computation itself has been evolving to become more “biological”, with the shrinking of computing elements to molecular scales, and the increasing focus in computer science on parallelism, asynchrony, stochasticity, and other properties seen in biological systems. It may be that a theory of biological computation will be a foundation not only for biology but also for a new era of more life-like computers. As pointed out in a recent NSF-sponsored workshop on “Shared Organizing Principles in the Computing and Biological Sciences” (Greenspan et al., 2010), it is only through the continuing close and equal collaboration between computer scientists and biologists, as well as cross-disciplinary education of a new generation of scientists, that the incipient field of biological computation may emerge as a source of foundational ideas for both fields.

About the Author:

Melanie Mitchell (mm@pdx.edu) is Professor of Computer Science at Portland State University, and External Professor and Member of the Science Board at the Santa Fe Institute.

Bibliography

Ben-Jacob, E. (2009). Learning from bacteria about natural information processing. *Annals of the New York Academy of Sciences, 1178*, 78-90.

Cardelli, L., & Corrado, P. (2009). Visualization in process algebra models of biological systems. In T. Hey, S. Tansley, & K. Tolle, Eds., *The Fourth Paradigm: Data Intensive Scientific Discovery*, 99-105.

Denning, P. J. (2007). Computing is a natural science. *Communications of the ACM, 50* (7), 13-18.

Detrain, C., Deneubourg, J. L., & Pasteels, J. M. (Eds.). (1999). *Information Processing in Social Insects.* Birkhäuser Basel.

Eeckman, F. H. (Ed.). (1994). *Computation in Neurons and Neural Systems.* Springer.

Gordon, D. M. (1999). Interaction patterns and task allocation in ant colonies. In C. Detrain, J. M. Pasteels, & J. L. Deneubourg, Eds., *Information Processing in Social Insects*, pp. 51-67.

Greene, M. J., & Gordon, D. M. (2007). Interaction rate informs harvester ant task decisions. *Behavioral Ecology, 18* (2), 451-455.

Greenspan, R. J., Mitchell, M., & Wise, J. (2010). Shared Organizing Principles in the Computing and Biological Sciences. Unpublished Report.

Hofmeyr, S., & Forrest, S. (2000). Immunology as information processing. In L. Segel, & I. Cohen (Eds.), *Design Principles for the Immune System and Other Autonomous Distributed Systems.* Oxford University Press.

Klemm, K., & Bornholdt, S. (2005). Topology of biological networks and reliability of information processing. *Proceedings of the National Academy of Sciences, USA, 102* (51), 18414-18419.

Mitchell, M. Complex systems: Network thinking. *Artificial Intelligence, 170* (18), 1194-1212.

Mitchell, M. (2009). *Complexity: A Guided Tour.* Oxford University Press.

Paton, R., Bolouri, H., Holcombe, W. M., Parish, J. H., & Tateson, R. (Eds.). (2004). *Computation in Cells and Tissues: Perspectives and Tools of Thought.* Springer.

Peak, D., West, J. D., Messinger, S. A., & Mott, K. A. (2004). Evidence for complex, collective dynamics and emergent, distributed computation in plants. *Proceedings of the National Academy of Sciences, USA, 101* (4), 918-922.