Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012

Some Computational Aspects of Essential Properties of Evolution and Life

Hector Zenil and James A.R. Marshall
Behavioural and Evolutionary Theory Lab
Department of Computer Science / Kroto Research Institute
University of Sheffield, UK

Abstract

While evolution has inspired algorithmic methods of heuristic optimisation, little has been done in the way of using concepts of computation to advance our understanding of salient aspects of biological evolution. We argue that under reasonable assumptions, interesting conclusions can be drawn that are of relevance to behavioural evolution. We will focus on two important features of life—robustness and fitness optimisation–which, we will argue, are related to algorithmic probability and to the thermodynamics of computation, subjects that may be capable of explaining and modelling key features of living organisms, and which can be used in understanding and formulating algorithms of evolutionary computation.

Keywords: evolutionary computation, information theory, thermodynamics of computation, biological robustness, algorithmic probability, fitness.

1 Introduction

> It is raining DNA outside. On the bank of the Oxford canal at the bottom of my garden is a large willow tree, and it is pumping downy seeds into the air... It is raining instructions out there; it's raining programs; it's raining tree-growing, fluff-spreading, algorithms. That is not a metaphor, it is the plain truth. It couldn't be any plainer if it were raining floppy disks.
>
> Richard Dawkins, 1986
> *The Blind Watchmaker*
> Longman Pub

With the work of Darwin it became clear that there were natural processes that could in fact shape features of biological organisms. Evolutionary computation was inspired by this idea, prompting us to posit a correspondence between algorithms and biology. What evolutionary computation does is to take natural selection as an algorithmic mechanism as an iterative procedure for optimisation in problem solving.

Insofar as evolutionary computation models important aspects of evolution, it does so because evolution itself behaves very much like a computational process. In evolutionary computation, biological evolution is matched with algorithms, and if biological evolution is in turn taken to be computational in nature, we can expect it to be driven by the same forces and to be constrained by the same laws as govern computation and information.

In her contribution to the Ubiquity symposium on the question "What is computation?" Melanie Mitchell pondered the possibility that biological computation was a process that occurred in nature, and that computing would eventually prove as fundamental for biology as physics had been for chemistry. Wolfram [36] has advocated a paradigm shift in proposing that when modelling nature, traditional mathematics be replaced with computational models based on systems such as cellular automata. Biologists have also drawn formal parallels between evolution by natural selection and optimisation [13, 14, 24].

Gregory Chaitin, one of the founders of algorithmic information theory [8] (together with R. Solomonoff [31], A. Kolmogorov [19] and L. Levin [22]), has strongly suggested that key features of evolution could be fully captured by mathematics, and approached using algorithmic information-theoretic models. Chaitin points out [9, 10]:

> DNA is essentially a programming language that computes the organism and its functioning; hence the relevance of the theory of computation for biology.

A central element in living systems turns out to be digital: DNA sequences refined by evolution encode the components and the processes that guide the development of living organisms. It is therefore natural to turn to computer science, where the concepts of programming languages, data structures, and algorithms are used to organise and characterise digital information.

We believe that important concepts in the theory of computation (in particular from algorithmic complexity and computational thermodynamics) can explain aspects of behaviour and evolution. We will focus on two pivotal features of life–robustness and fitness–which, we will argue, can be related to properties of computational systems.

2 Algorithmic Probability and Biological Robustness

Computer simulations performed as part of research into artificial life reproduced various known features of biological evolution [12, 20, 36], all of which turned out to be profoundly connected to the concept of (Turing) universal computation. Evolution seems to manifest some fundamental properties of computation, and not only does it seem to resemble an algorithmic process, it often seems to produce the kind of output distribution a computation could be expected to produce [25], as witness, for instance, the form in which the most important instructions for life are stored–in digital form (the DNA), even if in a convoluted fashion, and, despite all its intricate paths and complicated connections, in identifiable units of information (genes).

Among several key properties of living organisms is their robustness in the face of genetic change, yet organisms are highly evolvable [33]. We propose that the tools of algorithmic probability may be fruitfully applied to the task of framing a computational perspective on life and evolution. The algorithmic probability [31, 22] of a string s can be defined as

$$Pr(s) = \Sigma_{p:M(p)=s} 2^{-|p|} \tag{1}$$

Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012

Where $Pr(s)$ is the sum over all the programs for which M with p outputs the string s and halts. As p is itself a binary string, $Pr(s)$ is the probability that the output of M is s when provided with a sequence of random (uniformly-distributed) bits as input taken as a program p running on a universal (prefix-free[1]) Turing machine.

Consider a biological version, a container for each of the four nucleotides of DNA: A, T, C, and G. One connects the containers to a machine, and into the machine comes–as input–a random program with instructions for putting these nucleotides together to form a sequence of DNA. The machine binds the genetic material from scratch. This scenario looks much like what actually happens inside a cell, with ribosomes compiling DNA into proteins, and is consistent with an information-processing framework, where principles of computation would apply.

Biological processes may not behave or correspond to the mechanism of a Turing machine. But having learned directly from James D. Watson (the co-discoverer of the structure of DNA), Charles Bennett realised [3] that a special-purpose Turing machine existed in nature: RNA polymerase. An enzyme that crawls along a gene analogous to a machine "tape" transcribing DNA stepping left and right just as a Turing machine would do (this is not a minor thing, it is the possibility to access to any part of the machine tape that provides the machine with its computational power). The logical state of the RNA polymerase changes according to the chemical information written in the sequence and Bennett used this information to calculate the energy required for this biological processes [3].

Trying to feed a machine with random sequences of nucleotides will certainly lead to a frustrating experience. But for those outcomes with biological meaning, algorithmic probability will describe their frequency distribution in terms of their complexity. Not all programs will produce a biological valid output, most will be disappointing, if anything comes out, just as it would be for Turing machines that may or may not even stop or stop after only 1 step. If it is true that any valid rule would produce a valid computation, whether an output is valid from a biological perspective, even in the context of an automaton, is subject to interpretation and does not interfere with the fact that biology would follow information and algorithmic laws, just as should follow physical and chemical laws.

This is the main property of $m(s)$, that random computer programs produce structure by filtering out a portion of what one feeds them. Start with a random-looking string and run a randomly chosen program on it, and there is a good chance the random-looking string will be turned into a regular, highly structured one. The programs producing these structured strings may be the kinds of building blocks used in nature [29] thought to be common to living beings as basic units. For example, in an experiment with digital organisms designed to show how complex features originate from random mutation and natural selection it was found that most evolutionary paths build over prior intermediate states to arrive at the same evolved complex functions [24].

Robustness, in many contexts, means that a system's behaviour will remain qualitatively unchanged even if it is perturbed. In biology this translates into the ability of an organism to preserve its phenotype in spite of environmental perturbations or mutations. If many computations produce the same outcome, the outcome becomes less perturbable

[1] That is, a machine for which a valid program is never the beginning of any other program, a technicality that allows us to define a probability over strings produced by programs for which the sum is at most 1.

Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012

Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012

and more robust. Levin shows [22] that there is a universal (semi[2]) measure m that dominates any other probability measure Pr. In a world of computable processes, $m(s)$ (1) establishes that simple patterns which result from simple processes are likely, while complicated, random-looking patterns produced by complicated processes (long programs) are relatively unlikely. This means that if one tampers with a non-negligible number of programs (e.g. disarrangement or random removal) the output of the remaining ones will still produce the same frequency of patterns as the original. $m(s)$ is in fact also known as the Universal Distribution because it has the remarkable properties [22, 18].

Rodney Brooks [6] notes that

> . . . neither AI [Artificial Intelligence] nor Alife has produced artifacts that could be confused with a living organism for more than an instant. AI just does not seem as present or aware as even a simple animal and Alife cannot match the complexities of the simplest forms of life.

Wolfram has pointed out [36] that it is because we have been shaping nature in such a way as to produce engineered things that are meant to be understandable and within our full control, unlike most complex natural systems around us. It is perhaps with the hope of closing this gap that evolutionary computation has been used as an alternative to problem solving.

Strengthening the hypothesis that algorithmic probability may model certain biological phenomena some systems show preference for certain features, such as dihedral over cyclic symmetry, and phenotypic robustness [16], which seem to agree with the outcome expected of a bias of the kind produced in computational processes explained by algorithmic probability.

Robustness can also be observed in artificial self-assembly dynamics [15], suggesting that it is not exclusively a property of living systems but rather of simple rule systems which evolve persistent structures of high complexity.

This kind of "algorithmic robustness" may also be consistent with models where random point mutation is less important than mutations that occur at the level of the instructions determining biological functions, and may eventually explain processes of speciation favouring phenomena such as *epistasis* and models of *punctuated equilibria*. In this regard, we have found that there is also a computational approach for studying possible rates of change in evolutionary time-scales, such as stasis and rapid phenotypic evolution [5], investigated in terms of slow-down and speed-up phenomena in small computer programs competing to compute a function [17].

3 The Cost of Information Processing and Fitness

If biological systems can be regarded as computational systems one may inquire into biological systems as one would into computational systems. If algorithmic concepts can be successfully adapted to the biological context, key concepts developed in computation theory may not only be relevant to biology but may also explain fundamental aspects of biological processes.

[2] "semi" because, as an uncomputable function, it is approachable from below.

To grasp the role of information in biological systems think of a computer as an idealised information-processing system. Converting energy into information is today fairly easy to understand from a practical point of view [21, 3, 11]. Computers may be thought of as engines transforming energy into information. One need only connect one's computer to the electrical grid in order to have it perform a task and produce new information.

It is clear that animals convert information into energy–for example, by using information to locate food, to navigate within a habitat, or by learning how to hunt. Similarly, it is clear that energy can be used to extract even greater quantities of energy from the environment–through investment in the construction and operation of a brain. From this it follows that the extent to which an organism is adapted to or able to produce offspring in a particular environment depends on a positive exchange ratio between information and energy.

This works in reverse as well, that is the conversion of information into energy, and has been studied in the thermodynamics of computation. A good way to understand the process is by using the well-known thought experiment known as Maxwell's demon paradox [3, 4]. The basic idea is that one can use the knowledge about the microstate of a system to make it *hotter*. Using bits to produce energy is well explained in [11] and [30]. The connections to information theory and algorithmic complexity, however, are less well understood, but we have good ideas to find how to deeply connect these concepts using algorithmic information theory.

In exchanging information for energy and energy for information some energy inevitably escapes to the environment, hence it is also a subject important to ecology and ultimately to climate changing. Dissipation is a general phenomenon in the real world and it tells us that something is lost in the exchange process, something which itself interacts with the environment, affecting other organisms. In accordance with the second law of thermodynamics, one can see this as information about the system's irreversibility. In biology this kind of exchange happens all the time. Neurons, for example, dissipate about $10^{11}kT$ per discharge [3]. Computers (mainly because of their volatile memory devices–the RAM memory) also dissipate energy by at least $1kT$ per bit [27, 21, 3] (with k the Boltzmann constant and T the ambient temperature) which is the reason computers heat and require an internal fan.

In a recent paper [23] it was shown that, under certain assumptions, information processing by individuals could only be a fitness enhancing property. But it follows that any system with access to a finite amount of resources (e.g. memory) has to incur in a cost, assuming nothing other than information processing. Hence, if an organism aims to gather information about the world, updating its previous state by replacing old information with new (learning), a cost is unavoidable, and takes the form of dissipation. The connection to fitness is then quite natural. Living systems can be regarded as systems with access to finite resources and subject to the same costs as any other computational and physical system.

As pointed out in [32], natural selection can be seen as extracting information from the environment and coding it into a DNA sequence. But current efforts in the direction of quantifying information content typically do not venture beyond Shannon's communication theory. Nevertheless, as we are suggesting, analogies with computation and algorithmic complexity may help to understand how selection accumulates and exchanges information

Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012

and energy from the environment in the replacement of populations, as it has already help to connect computation and physics before in thermodynamics.

Information as a fitness enhancing property can only be so, however, if energy return is greater than the cost of processing the information. From the thermodynamics of computation we know that "forgetting" takes work and learning saves energy. Therefore, organisms cannot survive in a random world or without information storing resources, because some predictability is necessary for organisms to survive. The existence of living systems is therefore a thermodynamical proof that the world is more predictable than random.

4 Concluding Remarks

After the discovery of DNA, it was clear that at least an important part of biological systems and evolution was digital and computational in nature, involving code and devices for copying and reusing code, and for mechanically producing new components from specific instructions. As a complement to evolutionary computation, and indeed in recompense for what biology has bequeathed to computation in the form of nature-inspired models, we ought to acknowledge that such an exchange between disciplines (the theory of computation and evolutionary biology) is possible, not only because of the existence of strong similarities, but also because computation may explain fundamental aspects of life, as in itself it may explain important aspects of biological processes. We have reason to hope, therefore, that this approach will prove fruitful for investigating issues in biology using the theory of computation and algorithmic information theory.

We have yet to see how these key concepts may also shape new evolutionary computation algorithms, refining techniques and perhaps improving their understanding and application.

Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012

References

[1] L. Adleman. Toward a mathematical theory of self-assembly. *USC Tech Report*, 2000.

[2] N.A. Barricelli. Esempi numerici di processi di evoluzione. *Methodos.* 45–68, 1954.

[3] C.H. Bennett. The Thermodynamics of Computation–A Review. *International Journal of Theoretical Physics.* vol. 21, no. 12, pp. 905–940, 1982.

[4] C.H. Bennett, P. Gács, M. Li, P.M. B. Vitányi, W.H. Zurek. Thermodynamics of computation and information distance. *STOC '93 Proceedings of the twenty-fifth annual ACM symposium on Theory of computing.* 1993.

[5] K.D. Bennett. *Evolution and ecology: the pace of life.* Cambridge University Press, 1996.

[6] R. Brooks. The relationship between matter and life. *Nature.* 409:409–411, 2001.

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

[8] G.J. Chaitin, On the length of programs for computing finite binary sequences: Statistical considerations, *Journal of the ACM,* 16(1):145–159, 1969.

[9] G.J. Chaitin. Metaphysics, Metamathematics and Metabiology. in H. Zenil (ed.), *Randomness Through Computation.* pp. 93–103, World Scientific, 2011.

[10] G.J. Chaitin. Life as evolving software. in H. Zenil (ed.), *A Computable Universe.* World Scientific, forthcoming 2012.

[11] R.P. Feynman, A.J.G. Hey, D. Pines. *Feynman Lectures on Computation.* Westview Press, 2000.

[12] M. Gardner. Mathematical Games - The fantastic combinations of John Conway's new solitaire game "life". *Scientific American.* 223. pp. 120–123, 1970.

[13] A. Grafen. The formal Darwinism project: a mid-term report. *Journal of Evolutionary Biology.* 20: 1243–1254, 2007.

[14] A. Grafen. The simplest formal argument for fitness optimization. *Journal of Genetics.* vol. 87, no. 4, pp. 1243–1254, 2008.

[15] S. Griffith, D. Goldwater and J.M. Jacobson. Robotics: Self-replication from random parts. *Nature.* 437, 636, 2005.

[16] I.G. Johnston, S.E. Ahnert, J.P.K. Doye, and A.A. Louis. Evolutionary dynamics in a simple model of self-assembly. *Phys. Rev. E.* 83, 066105, 2011.

[17] J. Joosten, F. Soler-Toscano and H. Zenil. Program-size Versus Time Complexity, Speed-up and Slowdown Phenomena in Small Turing Machines. *Int. Journ. of Unconventional Computing.* special issue on Physics and Computation, vol. 7, pp. 353-387, 2011.

[18] W. Kirchherr, M. Li and P. Vitányi. The Miraculous Universal Distribution. *The Mathematical Intelligencer.* vol. 19, no. 4, pp. 7–15, 1997. http://www.cs.sjsu.edu/~kirchher/Publications/inductive.pdf

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

[20] C.G. Langton. Studying artificial life with cellular automata. *Physica D: Nonlinear Phenomena* 22 (1-3): 120–149, 1986.

[21] R. Landauer. Irreversibility and heat generation in the computing process, *IBM Journal of Research and Development.* vol. 5, pp. 183–191, 1961.

[22] L. Levin. Laws of information conservation (non-growth) and aspects of the foundation of probability theory. *Problems in Form. Transmission* 10. 206–210, 1974.

[23] J.M. McNamara and S.R.X. Dall. Information is a fitness enhancing resource, *Oikos.* 119: 231–236, 2010.

[24] R. Lenski, C. Ofria, R.T. Pennock and C. Adami. The Evolutionary Origin of Complex Features, *Nature* 423, 139–144, 2003.

[25] H. Li, R. Helling, C. Tang and N. Wingreen, Emergence of Preferred Structures in a Simple Model of Protein Folding. *Science.* New Series, vol. 273, no. 5275, pp. 666–669, 1996.

[26] M. Mitchell. What Is Meant By "Biological Computation"? *ACM Communications Ubiquity.* vol 2011, number, Pages 1–7, February, 2011.

[27] J. von Neumann. in A. Burks (ed). *Theory of Self-Reproducing Automata.* University of Illinois Press, 1966.

[28] P.W.K. Rothemund, N. Papadakis, E. Winfree. Algorithmic Self-Assembly of DNA Sierpinski Triangles. *PLoS Biol* 2(12): 2004. http://www.plosbiology.org/article/info:doi/10.1371/journal.pbio.0020424

Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012

[29] R.P Shetty, D. Endy and T.F. Knight Jr., Engineering BioBrick vectors from BioBrick parts. *Journal of Biological Engineering.* 2:5, 2008.

[30] J. Sethna. *Statistical Mechanics: Entropy, Order Parameters and Complexity.* Oxford University Press, 2006.

[31] R. Solomonoff. A Preliminary Report on a General Theory of Inductive Inference. (Revision of Report V-131), Contract AF 49(639)-376, *Report ZTB*–138, Zator Co., Cambridge, Mass., Nov, 1960.

[32] H.P. de Vladar, N.H. Barton. The contribution of statistical physics to evolutionary biology, *Trends in Ecology & Evolution.* 26(8) pp. 424–432, 2011.

[33] A. Wagner. *Robustness and Evolvability in Living Systems.* Princeton University Press, 2005.

[34] E. Winfree. *Algorithmic Self-Assembly of DNA Thesis.* in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy, Caltech, 1998.

[35] E. Winfree. Simulations of computing by self-assembly. *Technical Report CS-TR:1998.22.* Caltech, 1998.

[36] S. Wolfram. *A New Kind of Science.* Wolfram Media, 2002.

Nature Precedings : hdl:10101/npre.2012.6867.1 : Posted 3 Feb 2012