Journal of Theoretical Biology 443 (2018) 1–9

Contents lists available at ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/jtbi

Performance limits and trade-offs in entropy-driven biochemical computers

Dominique Chu

School of Computing, University of Kent, Canterbury CT2 7NF, United Kingdom

ARTICLE INFO

Article history:
Received 25 September 2017
Revised 16 January 2018
Accepted 17 January 2018

Keywords:
Biological computing
Information thermodynamics
Cost of computation
Linear noise approximation

ABSTRACT

It is now widely accepted that biochemical reaction networks can perform computations. Examples are kinetic proof reading, gene regulation, or signalling networks. For many of these systems it was found that their computational performance is limited by a trade-off between the metabolic cost, the speed and the accuracy of the computation. In order to gain insight into the origins of these trade-offs, we consider entropy-driven computers as a model of biochemical computation. Using tools from stochastic thermodynamics, we show that entropy-driven computation is subject to a trade-off between accuracy and metabolic cost, but does not involve time-trade-offs. Time trade-offs appear when it is taken into account that the result of the computation needs to be measured in order to be known. We argue that this measurement process, although usually ignored, is a major contributor to the cost of biochemical computation.

1. Introduction

Computing architectures based on biochemistry, rather than semi-conductor technologies, are attracting increasing interest as alternative models of computation (Amos, 2004). Biochemistry can be used to engineer novel types of computers based on biological components. Examples include, DNA based computers (Lakin and Phillips, 2011; Seelig et al., 2006), robots controlled by slimemolds (Tsuda and Zauner, 2009), or logic gates implemented in living cells (Amos et al., 2015; Friedland et al., 2009; Silva-Rocha and de Lorenzo, 2011; Sole and Macia, 2013). Beside this technological importance of biochemical computers, there is now also an increasing appreciation that information processing may be an important fitness contributing function for natural organisms (Davies and Walker, 2016; Walker et al., 2016). There are a number of biosystems that have been studied as *in vivo* special purpose computations. For example, kinetic proofreading (Fluitt et al., 2007; Ninio, 1975) greatly enhances the copying fidelity during translation and is often interpreted as an *in vivo* computation. A classical example of biochemical computation is bacterial sensing (Berg and Purcell, 1977; Govern and Wolde, 2014; Gregor et al., 2007; Mehta and Schwab, 2012), whereby cells measure molecular concentrations in their environment and modify internal pathways and gene expression levels in response. Chemotaxis (Alon et al., 1999), for instance, depends on organisms sensing a molecular concentration gradient by computing the difference between several measurements, either in time or across the cell volume. Most recently even bacterial growth dynamics has been interpreted as a computational process (Chu, 2015; Chu and Barnes, 2016).

Detailed case studies of biological computers often find performance limits to biochemical computations. For a simple gene-switch, Zabet and coworker found a trade-off between the cost, the accuracy and the speed of the computation (Chu et al., 2011; Zabet and Chu, 2010). Similar trade-offs were established for other biological systems, including chemotaxis (Lan et al., 2012), regulation of nutrient uptake (Chu and Barnes, 2016), and translation (Johansson et al., 2012); lower limits on the cost of sensing (not involving trade-offs) have also been found recently (Govern and Wolde, 2014; Kaizu et al., 2014).

Intuitively such trade-offs are to be expected. Bio-chemical networks are stochastic systems and as such subject to noise. Overcoming this noise requires energy input and time. Energy-time-accuracy trade-offs are also implied by the classical results on the physics of computation (Bennett, 1982; Feynman and Hey, 2000; Landauer et al., 1990). While there does not seem to be a lower limit for the energy used during a computation, Bennett (1982) pointed out that in the zero energy limit the speed of computation goes to zero. Computations that complete within a finite time, therefore require finite energy resources.

The question is now whether one can go beyond both the individual case studies of biochemical computers and the intuitive arguments and establish a model which provides insights into the origins of the performance limits to biochemical computations. The

E-mail address: d.f.chu@kent.ac.uk

https://doi.org/10.1016/j.jtbi.2018.01.022
0022-5193/

task is a difficult one. For one, there is a wide variety of approaches to biochemical computation (only some of which are mentioned above). At the same time, there is no general definition of biological computation, i.e. it is not clear how to distinguish a reaction network that computes from one that does not.

For the purpose of this article, we will take a pragmatic approach with respect to the latter question and simply identify (in Section 2.1) computation with out-of-equilibrium biochemical processes. According to this, every biochemical process that is not in equilibrium performs a computation. As far as the wide variety of biochemical computations are concerned, we will abstract away from specific models and define the concept of *entropy driven computers* (EDC) in Section 2.2. This will capture many properties of *in vivo* computers as they appear in biological systems. EDCs are in many aspects different from real biological networks, but we will argue that they share important characteristics with a wide range of biochemical computers. It is perhaps best to think of EDCs as test-tube biochemistry, as it is frequently used in biological research to study reaction networks *in vitro*.

We model EDCs as continuous time Markov chain models of biochemical systems. We assume that each model is initialised in some state and then left to relax to equilibrium. We will then interpret this relaxation process as a computation. Throughout this article we will assume that the EDC is of mesoscopic scale. By this we mean that it is still affected by stochastic fluctuations, but that it is also within the range of validity of the linear noise approximation (van Kampen, 2007). Simply put, this assumption states that the stochastic system behaves like the deterministic equivalent, plus some noise. The linear noise approximation is a very good approximation for mesoscopic systems and holds true for a wide range of biochemical systems, and hence for a wide range of biological "computers" such as gene regulatory networks, protein-protein interactions or intra and inter-cellular signalling systems, although clearly there are systems that will not be captured by this assumption.

For the purpose of the present contribution, we will identify the cost in energy of a computation with the entropy produced during the computation. While this does not quantify the actual metabolic cost of this computation, it is directly related to it. We will first show that the linear noise approximation implies that the entropy production scales linearly with the system size, while the time-scale to approach equilibrium (which we interpret as the computing time) remains invariant. This means that there is a trade-off between the cost of the computation and its accuracy, but there is no trade-off involving the speed of the computation. Contrary to previous work (or apparently so), this suggests that speed-energy trade-offs are not a fundamental property of biochemical computation *per se*

A trade-off involving time emerges only when it is taken into account that the result of the computation must be measured in order for the computation to have any impact in the world. Any measurement of the outcome of the computation in turn requires a measurement device. This device needs to be brought into contact with the computer to determine its state. Device and computer then form a joint system, which initially will be out of equilibrium but relaxes to an equilibrium. This relaxation constitutes the measurement process. Formally a measurement is thus also an entropy driven computation. As we will show below, restoring the computer to its original state, while at the same time leaving the measurement device in a state that indicates the result of the computation, requires both energy input and time. It also leads to a trade-off between the energy used and the speed with which the restoration can be completed with a given confidence. A second trade-off involving time arises from the stochastic nature of the computer. A single measurement only indicates the correct result with a certain probability. Repeated measurements are necessary in order to sample the state of the computer reliably, thus leading to a trade-off between accuracy and time.

2. Results

2.1. Computation by biochemical systems

The current modus operandi in the field of biochemical computing is to identify a biological system (such as sensing or proof-reading) as a computation when it implements a function that is naturally interpreted as a computation. This approach enables deep insights into specific examples, but is likely to miss most instantiations of biochemical computation. It would be much more useful to have a concept of biochemical computation that is independent of its function, just as in computer science computation is defined with respect to a number of specific mathematical models, not by reference to what is computed.

The best known model of computation is the *Turing machine*. This is a mathematical construct consisting of a "reading head" that is reading and writing a tape, while changing its internal states in the process, until it reaches a "halting state," at which point the computation stops. It is believed that for every computable function there is a corresponding Turing machine that computes it. Based on this, one could be tempted to define a biochemical process as a computation if there is a Turing machine that simulates this process. This does not work however: The natural equivalent of a halting state in biochemical systems is the equilibrium state, i.e. the state of the biochemical system where reactions are in detailed balance. Unlike the halting state of a Turing machine, the equilibrium state is of a statistical nature. This means that on average there are no net-fluxes across the network of reactions (Beard et al., 2002; Qian and Beard, 2005). This does not mean, however, that reactions stop. Even in equilibrium there is an ongoing chemical activity. Crucially though, the sequence of reaction events is symmetric in time (van Kampen, 2007), such that an observer would not be able to tell apart an actual sequence of reactions from a (hypothetical) reversed sequence. Equilibrium is not time-directed. Computation, on the other hand, is necessarily time directed, mapping a particular input to a particular output. Equilibrium systems are therefore not able to compute. Sample paths of equilibrium biochemical systems can still be simulated and are thus computable by Turing machines, whether or not the system is in equilibrium. This demonstrates that not all processes that can be simulated by Turing machines are also themselves processing information.

For the purpose of this paper, we will adopt the simplest working hypothesis and postulate that the equilibrium state is the only halting state of biochemical computers. This implies that every biochemical system that is not in equilibrium is in the process of computing. By adopting this definition, we also accept that most biochemical computers will not do any useful calculations, just as almost all Turing machines do not compute anything of interest.

2.2. Entropy driven computation

In this section we define an EDC as a closed, stochastic, biochemical system, denoted by a fraktur S, $\mathfrak{S}$. The system does not exchange particles with the environment. We conceptualise $\mathfrak{S}$ as consisting of a (typically very large) number of discrete microstates $\mathfrak{s}_0, \mathfrak{s}_1, \ldots, \mathfrak{s}_m$ (see Supplementary Section 1 for more details). An EDC is initialised in a macrostate $M_0^{\mathfrak{S}}$ characterised by a specified abundance for each of its constituent biochemical species at time $t = 0$; see Supplementary Section 1 for a detailed explanation of what we mean by "macrostate." After a transient period, the biochemical system approaches an equilibrium state $M_\infty^{\mathfrak{S}}$ characterised by detailed balance. The approach to equilibrium is the