

Article

Dynamics of Information as Natural Computation

Gordana Dodig Crnkovic

Mälardalen University, School of Innovation, Design and Engineering, Sweden;

E-Mail: gordana.dodig-crnkovic@mdh.se

Received: 30 May 2011; in revised form: 13 July 2011 / Accepted: 19 July 2011 /

Published: 4 August 2011

Abstract: Processes considered rendering information dynamics have been studied, among others in: questions and answers, observations, communication, learning, belief revision, logical inference, game-theoretic interactions and computation. This article will put the computational approaches into a broader context of natural computation, where information dynamics is not only found in human communication and computational machinery but also in the entire nature. Information is understood as representing the world (reality as an informational web) for a cognizing agent, while information dynamics (information processing, computation) realizes physical laws through which all the changes of informational structures unfold. Computation as it appears in the natural world is more general than the human process of calculation modeled by the Turing machine. Natural computing is epitomized through the interactions of concurrent, in general asynchronous computational processes which are adequately represented by what Abramsky names “the second generation models of computation” [1] which we argue to be the most general representation of information dynamics.

Keywords: philosophy of information; information semantics; information dynamics; natural computationalism; unified theories of information; info-computationalism

1. Introduction

This paper has several aims. Firstly, it offers a computational interpretation of information dynamics of the info-computational universe [2,3]. Secondly, it argues for the necessity of generalization of the model of computation beyond the traditional Turing machine model [4]. In doing so, it addresses a number of frequent misunderstandings when it comes to the models of computation

and their relationships to physical computational systems. Thirdly, I suggest that info-computationalism as a new philosophy of nature provides a basis for unification of knowledge from currently disparate fields of natural, formal and technical sciences [5]. The on-going developments of among others Bioinformatics, Computational Biology, Neuroscience, Computational Evolutionary Biology and Cognitive Science show that in practice biological systems already are studied as information processing systems and modeled by the advanced computational tools, theories and machinery which belong to the new generation paradigms of computing related to networks of concurrent computational processes, unlike Turing machines, which represent formalization of a human calculation process.

Both information and computation are concepts and phenomena still intensely researched from the multitude of perspectives. In order to be able to meaningfully model the universe as a computational network, the model of computation must be adequate. When it comes to generalization of the idea of computation, there are several confusions. The recurrent one is about the relationship between the universe and the computer. It must be emphasized that the universe is not equivalent to a PC in any interesting way and when we talk about computing universe, the notion of computation must be generalized to be able to reflect the richness of phenomena found in nature. The attempt to represent vastly complex systems comprising a huge number of different organizational levels by a Turing machine model is inadequate, and indeed presents too powerful a metaphor, as already noticed by critics [6]. Among typical criticisms of naturalist computationalism are those grounded in the belief that computationalism necessarily implies digital (discrete) computation. Of course if the computing system is continuous then computation should be continuous at least on some level of abstraction. The continuous/discrete and analog/digital are interesting questions with deep consequences and there has been no consensus on this topic in the scientific community to date [7]. Some would claim that space-time can be discretized without loss of explanatory power, while the others argue that discretization necessarily breaks symmetry in such a fundamental way that Lorentz invariance is lost. The article presents some of the questions about the nature of computation in terms of discrete/continuum and analog/digital dichotomies and answers criticisms regarding the ability of computational models to adequately represent the dynamics of physical systems.

2. Different Approaches to the Dynamics of Information

Van Benthem and Adriaans [8] define several dynamic information stances: *the logical stance* (dynamics of information update), *the Shannon view* (dynamics of information communication/transmission), and *the Kolmogorov view* (dynamics of encoding and decoding). Processes considered representing information dynamics are: *questions and answers* (Kamp and Stokhof), *observations* (Baltag, van Ditmarsch and Moss), *communication* (Devlin and Rozenberg), *learning* (Adriaans, Kelly), *belief revision* (Rott), *inference* (van Benthem and Martinez), *game-theoretic interactions* (Walliser) and *computation* (Abramsky). Which of them will be in focus depends on what aspect of information dynamics we are interested in.

The early developments of the field of dynamic of information such as seminal work of Dretske [9] (information flow as linguistic regularities) and Barwise and Seligman [10] (informational relation between situations) are described in [8], as well as [11] and [12]. Information was considered to be a building block of knowledge and thus supposed to always be true. For Dretske, even though false

information contains no information at all, it still can have a meaning. Burgin on the other hand argues in [11] (Sections 2.4 and 2.5) that both false information and false knowledge should be taken into account and studied as real-world phenomena in knowledge production. He also points out that information comes in degrees and it is often fuzzy, all of which must be accounted for in modeling of information dynamics.

Depending on the understanding of information, different types of information dynamics are defined by Floridi [13] as: “(i) the constitution and modeling of information environments, including their systemic properties, forms of interaction, internal developments, applications, *etc.*; (ii) information life cycles, *i.e.*, the series of various stages in form and functional activity through which information can pass, from its initial occurrence to its final utilization and possible disappearance; and (iii) computation, both in the Turing-machine sense of algorithmic processing, and in the wider sense of information processing. This is a crucial specification. Although a very old concept, information has finally acquired the nature of a primary phenomenon only thanks to the sciences and technologies of computation and ICT (Information and Communication Technologies). Computation has therefore attracted much philosophical attention in recent years”.

The first two types of the dynamics of information refer to the classical notion of information as constituent of knowledge. Information processing is a more general type of dynamics and can be applied to any type of information.

A study of information dynamics within a framework of logic is presented in van Benthem’s recent book [14] developed as a theory of *information-based rational agency* including *intelligent interaction* between *information-processing agents*. Van Benthem describes *dynamic logics for inference, observation and communication*, with update of knowledge and revision of beliefs, changing of preferences and goals, group action and strategic interaction in games.

Informational dynamics is by Hofkirchner [15] characterized as *a process of self-organization* of a system. According to this view, whenever a self-organizing system relates to the environment, it *creates* information, rather than *processes* it. This concept might be called *emergent information*. Hofkirchner’s “triple-c” model [16] describes information generation as consisting of *cognition* (information generation of a self-organizing system vis-à-vis its environment); *communication* (the coupling of cognitive processes of self-organizing systems) and *cooperation*.

3. The Dynamics of Information as Computation: Info-Computationalism

Info-computationalism (ICON) [2,3,17,18] is one of the approaches that takes information dynamics to be a process of computation. It is built on a dual-aspect foundation with information and computation as basic elements. Within this framework, reality is a hierarchy of levels, starting from the basic proto-information as a stuff of the universe, and building a number of higher levels of organization through computational processes.

Info-computationalism is a unifying approach that brings together *informationalism* (Informational Structural Realism of Floridi [19]; Informational Realism of Sayre [20]) with the *naturalist computationalism/pancomputationalism* (Zuse, Fredkin, Wolfram, Chaitin, Lloyd) [21–25]. Informationalism argues that the entire existing physical universe is an informational structure, while (pan)computationalism takes the universe to constantly be computing its own next state. With the

universe represented as a network of computational processes at different scales or levels of granularity, information is both a substrate and a result of natural computation [26].

In sum: if the physical universe is an informational structure, natural computation is a process governing the dynamics of information. Information and computation are two mutually defining concepts [3]. Being inextricably connected, information and computation are often conceptually mixed together. Burgin [11] (p. 462) quotes Nonaka as claiming that “information is the flow, and knowledge is the stock”—while more accurate description would be that knowledge represents informational architecture whereas the flow of information is a process of computation.

Information dynamics expressed as computation is related to information by Burgin in the article *Information Dynamics in a Categorical Setting*. It is based on two types of representation of information dynamics, the categorical representation and the functorial representation. This approach enables building a common framework for information and computation. Burgin emphasizes that category theory is also “used as unifying framework for physics, biology, topology, and logic, as well as for the whole mathematics. This provides a base for analyzing physical and information systems and processes by means of categorical structures and methods” [27].

Information is fundamental as a basis for all knowledge and its processing lies behind all our cognitive functions. In a wider sense of proto-information, information represents every physical/material phenomenon, [28]. Information is for an agent created by the process of differentiation. Bateson [29] famously defined information as “a difference which makes a difference”.

Instructive in this context is connecting the information flow in computation, interactive process and games (representing the rules or logic) as steps towards a “fully-fledged dynamical theory” [30]. The idea is to study the processing of information as it appears in computational systems. It discusses the information flow in computation, using game-based models of interactive processes, and explicating the *interplay of information and computation*. Nondeterministic concurrent games are used in [31] to formalize games in which both Player(s) and Opponent(s) can interact in a *distributed way*, and *without the assumption that their moves alternate*.

The claim that “information **is** what information **does**” is made several times by van Benthem in [8], paraphrasing Lewis who famously asserted: “In order to say what a meaning **is**, we may first ask what a meaning **does**, and then find something that does that” [32].

Abramski [30] takes the process stance and studies information in action. He makes clear that “in a computational perspective, *it makes little sense to talk about static data structures in isolation from the dynamic processes that manipulate them*”. This is in agreement with van Benthem who declares that “*structure should always be studied in tandem with a process!*” and “*No information without transformation!*”

From everyday experiences we know that computation can provide information. Some typical examples are search functions and automatic translation. On the web, a search engine finds by computation the information we are interested in. Wolfram’s Alpha computes even further by filling in the gaps in the information that exists on the web. Or, in short, as [30] points out: a useful output of a computation has two aspects: *making information explicit*, *i.e.*, extracting the normal form (so that we can understand) and *reducing the data*—eliminating redundant/irrelevant information. Both of the aspects are important for the user of processed information, who in general is an agent—biological (typically human), software or a robotic system. In many informational theories agents are supposed to

be humans and information is exchanged and processed among humans. However when studying evolution and the development of different kinds of natural agents (organisms) and their information processing capabilities and behaviors it becomes evident that the idea of an agent must be generalized to encompass the whole spectrum of agency, from very rudimentary to the most complex.

In agent-based models, which are a class of computational models for simulating the behavior of autonomous agents, not only the notion of an agent but also the idea of their interactions must be generalized. What is exchanged during communication between agents can be different, not always necessarily words or written symbols. “*Agents and their interactions are intrinsic to the study of information flow and increase in computation. The classical theories of information do not reflect this adequately*” [30].

4. The Turing Machine Model of Computation

The notion of computation as a formal (mechanical) symbol manipulation originates from discussions in mathematics in the early 20th century. The most influential program for formalization was initiated by Hilbert, who treated formalized reasoning as a symbol game in which the rules of derivation are expressed in terms of the syntactic properties of symbols. As a result of Hilbert’s program large areas of mathematics have been formalized. Formalization implies the establishment of the basic language which is used to formulate the axioms and derivation rules defined such that the important semantic relationships are preserved by inferences defined only by the syntactic form of expressions. Hilbert’s *Grundlagen der Mathematik*, and Whitehead and Russell’s *Principia Mathematica* are examples of such formalization.

The basic idea was that any operations that are sensitive only to syntax can be simulated mechanically. What the human following a formal procedure/algorithm does by recognition of syntactic patterns, a machine can be made to do by purely mechanical means. Formalization and computation are closely related and together entail that reasoning which can be formalized can also be simulated by the Turing machine. Turing assumed that a machine operating in this way would actually be doing the same thing as the human performing computation.

The Turing machine is supposed to be given from the outset—its logic, its physical resources, and the meanings ascribed to its actions. The Turing machine model essentially presupposes a human as a part of the system—the human is the one who provides material resources, poses the questions, and interprets the answers.

5. The Church-Turing Thesis

The Church-Turing thesis states that Turing machine can perform any effective computation, *i.e.*, calculation defined by “purely mechanical” procedure or “rule of thumb”. In its original formulation [33,34], the thesis says that *effectively calculable* function of positive integers is identical with a recursive function of positive integers or of a lambda-definable function of positive integers [34] (p. 356). Actually, the Church-Turing thesis has served as a definition for computation. Computation was considered to be a process of *computing the function of positive integers*. There has never been a proof, but the evidence for its validity comes from the equivalence of several computational models such as cellular automata, register machines, and substitution systems.

The Church-Turing thesis has been extended to a proposition about the processes in the natural world by Stephen Wolfram in his *Principle of computational equivalence* [23] in which he claims that there are only a small number of intermediate levels of computing before a system is universal and that *most natural systems can be described as universal computational mechanisms in a Turing sense*. If the physical systems quickly reach the expressive power of universal Turing machine, the question is if that is all computational expressive power physical systems may possess. A number of computing specialists and philosophers of computing (Siegelman, Burgin, Copeland, and representatives of natural computing) question the claim that all computational phenomena in all relevant aspects are equivalent to the Turing machine.

One of the problems in the discussion about the Turing machine model is the word “machine” as the *model of computing* is often confused with a *physical machine*. Turing’s original nomenclature was more apt. He used the expression LCM (*logical computing machines*) which makes it clear that they are *not physical machines*, but *logical models*. The next source of confusion (apart from understanding of Turing machine as a *machine*) is the fact that *models can be expressed in terms of each other*. That does not imply that they possess the same behavior or the same expressive power. It is not only the *function* that is important to be calculated when a program executes and stops, delivering a result, but sometimes it is actually the *behavior* (which is often not supposed to cease) that we expect from computing machines today. Internet does not calculate a function and it is not expected to stop. Even though some would object that particular programs may well be executing and delivering specific results, *the system as a whole* is not. Also, even though one may connect physical implementations of Turing machines in a network, *a network itself cannot be modeled as a Turing machine in any interesting way*.

6. Paradigms and Models of Computation

Ever since Turing proposed his machine model which identifies computation with the execution of an algorithm, there have been questions about how widely the Turing machine model might be applicable. Church-Turing Thesis establishes the equivalence between a Turing machine and an algorithm, interpreted as to imply that all of computation must be algorithmic. However, with the advent of computer networks, the model of a computer in isolation, represented by a Turing machine, has become insufficient. Today’s software-intensive and intelligent computer systems have become large, consisting of massive numbers of autonomous and parallel elements across multiple scales. At the nano-scale they approach programmable matter; at the macro scale, huge numbers of cores compute in clusters, grids or clouds, while satellite data are used to track global-scale phenomena. The common for these modern computing systems is that they are *ensembles* (composed of the parts which act together to achieve a common goal) and they are *physical* (as they act in the physical world and communicate with their environment through sensors and actuators).

At the moment, the closest to common acceptance is the view of *computation as information processing*, found in different mathematical accounts of computation as well as Cognitive science and Neuroscience [35]. *Basically, for a process to be a computation a model must exist such as algorithm, network topology, physical process or in general any mechanism which ensures definability of its behavior* [4].

Unlike the Turing machine model which originates from Hilbert's program for logics and mathematics (proposed in the early 1920s), other types of models of computation such as process models (Petri nets, Process Algebra, and Agent-Based models) appeared in the past decades (the theory of Petri nets in 1962), specifically in Computer Science. Indicatively, the present day formal methods in Systems Biology include Rule-Based Modeling of Signal Transduction, Process Algebras, Abstract Interpretation, Model Checking, Agent-Based Modeling of Cellular Behavior, Boolean Networks, Petri Nets, State Charts and Hybrid Systems. However, concurrency models have emerged in a bottom-up fashion in order to tackle present day networks of computational systems and it will take a few years until they reach the shared world view as tools of thinking in a new paradigm of computation.

7. Natural Computation/Physical Computation

Natural computation/physical computation is a new paradigm of computing which deals with computability in the physical world, that has brought a fundamentally new understanding of computation [36].

Kampis for example, in his book *Self-Modifying Systems in Biology and Cognitive Science* [37] claims that the Church-Turing thesis applies only to *simple systems*. According to Kampis, complex biological systems must be modeled as *component-systems* which are self-referential, self-organizing and self-generating and whose behavior is computational in a general sense which goes beyond Turing machine model:

*“a component system is a computer which, **when executing its operations (software) builds a new hardware**.... [W]e have a computer that re-wires itself in a hardware-software interplay: the hardware defines the software and the software defines new hardware. Then the circle starts again”* [37] (p. 223).

Natural computing is characterized by a *bidirectional research* [38] as the natural sciences are rapidly absorbing ideas of information processing, and computing at the same time assimilates ideas from natural sciences.

Compared with new computing paradigms, Turing machines form the proper subset of the set of information processing devices, in much the same way that Newton's theory of gravitation is a special case of Einstein's theory, or Euclidean geometry is a limit case of non-Euclidean geometries.

Natural computation is a study of computational systems including the following: (1) Computing techniques that take inspiration from nature for the development of novel problem-solving methods; (2) Use of computers to simulate natural phenomena; and (3) Computing with natural materials (e.g., molecules, atoms).

Natural computation is well suited for dealing with large, complex, and dynamic problems. It is an emerging interdisciplinary area closely related with the artificial intelligence and cognitive science, vision and image processing, Neuroscience, Systems Biology, Bioinformatics—to mention but a few.

Fields of research within natural computing are among others Biological Computing/Organic Computing, Artificial Neural Networks, Swarm Intelligence, Artificial Immune Systems, computing on continuous data, Membrane Computing, Artificial Life, DNA computing, Quantum computing,

Neural computation, Evolutionary computation, evolvable hardware, self-organizing systems, emergent behaviors, machine perception and Systems Biology.

Evolution is a good example of natural computational process. The kind of computation it performs is Morphological computation [39,40]. The result of this computation is the body shape and material optimized for the class of organisms in a given type of environment.

For computationalism, *interactive computing* (such as, amongst other, agent-based) is the most appropriate model, as it naturally suits the purpose of modeling a network of mutually communicating processes [2,3].

According to pancomputationalism (naturalist computationalism) [2–5,17,18,26,27,41–50] one can view the time development (dynamics) of physical states in nature as information processing, and learn about its computational characteristics. Such processes include self-assembly, developmental processes, gene regulation networks, gene assembly in unicellular organisms, protein-protein interaction networks, biological transport networks, and similar.

Natural computing has specific criteria for the success of a computation. Unlike the case of the Turing model, the halting problem is not a central issue, but instead *the adequacy of the computational behavior*. Organic computing system, e.g., adapts dynamically to the current conditions of its environments by self-organization, self-configuration, self-optimization, self-healing, self-protection and context-awareness.

In many areas, we have to computationally model emergence not being algorithmic [51–53] which makes it interesting to investigate computational characteristics of non-algorithmic natural computation (sub-symbolic, analog).

Much like the research in other disciplines of Computing such as AI, SE, and Robotics, Natural computing is interdisciplinary research, and has a synthetic approach, unifying knowledge from a variety of related fields. Research questions, theories, methods and approaches are used from Computer Science (such as Theory of automata and formal languages, Interactive computing), Information Science (e.g., Shannon’s theory of communication), ICT studies, mathematics (such as randomness, Algorithmic theory of information), Logic (e.g., pluralist logic, game logic), Epistemology (especially naturalized epistemologies), evolution and Cognitive Science (mechanisms of information processing in living organisms) in order to investigate foundational and conceptual issues of Natural computation and information processing in nature.

“(O)ur task is nothing less than to discover a new, broader, notion of computation, and to understand the world around us in terms of information processing” [38].

This development necessitates what [42] calls computational research “beyond the constraints” of “normal science”. In other words, a paradigm shift or [30] “*the second generation models of computation*” (“the first generation” being the Turing machine model).

8. The Computing Universe: Naturalist Computationalism/Pancomputationalism

Konrad Zuse was the first to suggest (in 1967) that the physical behavior of the entire universe is being computed on a basic level, possibly on cellular automata, by the universe itself which he referred to as “Rechnender Raum” or Computing Space/Cosmos. Consequently, Zuse was the first pancomputationalist (naturalist computationalist). Here is how Chaitin explains pancomputationalism:

“And how about the entire universe, can it be considered to be a computer? Yes, it certainly can, it is constantly computing its future state from its current state, it's constantly computing its own time-evolution! And as I believe Tom Toffoli pointed out, actual computers like your PC just hitch a ride on this universal computation!” [24].

Fredkin in his *Digital Philosophy* [54] suggests that particle physics can emerge from cellular automata. For “computational universe” according to Fredkin, “reality is a software program run on a cosmic computer. The universe is digital, time and space are discrete. Humans are software running on a universal computer”.

Even Wolfram in his *A New Kind of Science* advocates pancomputationalist view, a new dynamic kind of reductionism, in which complexity of behaviors and structures found in nature are derived (generated) from a few basic computational mechanisms. Natural phenomena are thus the products of computation processes. In a computational universe new and unpredictable phenomena emerge as a result of simple algorithms operating on simple computing elements such as, e.g., cellular automata, and complexity originates from the bottom-up driven emergent processes. Cellular automata are equivalent to a universal Turing machine. Von Neumann provided a proof that an automaton consisting of cells with four orthogonal neighbors and 29 possible states may simulate a Turing machine for some configuration of about 200,000 cells [55] (p. 227). Cook has proven Wolfram’s conjecture that one of the simplest possible cellular automata (number 110) is capable of universal computation [56]. This result was first described in [23].

Wolfram’s critics remark however that cellular automata do not evolve beyond a certain level of complexity. The mechanisms involved do not necessarily produce evolutionary development. Actual physical mechanisms at work in the physical universe appear to be quite different from simple cellular automata. Critics also claim that it is unclear if the cellular automata are to be thought of as a metaphor or whether real systems are supposed to use the same mechanism on some level of abstraction. Wolfram meets this criticism by pointing out that cellular automata are models, and as such, surprisingly successful ones.

9. Criticisms of the Computational Views of the Universe

In his article on Physical Computation for Stanford Encyclopedia of Philosophy, Piccinini [57] presents several critical arguments against Pancomputationalism/Naturalist computationalism. The unlimited Pancomputationalism, the most radical version of Pancomputationalism, according to Piccinini asserts that “every physical system performs every computation—or at least, every sufficiently complex system implements a large number of non-equivalent computations”. I find these to be two substantially different claims. The first one, that every system executes every computation, has little support in physics and other natural sciences. Different sorts of systems perform different sorts of dynamical behaviors. The second claim, that a sufficiently complex systems implement a large number of different computations, is in accordance with natural sciences and essentially different from the claim that every system performs every computation.

The *limited pancomputationalism* on the other hand is a belief that “every physical system performs *some* (as opposed to every) computations. A slightly stronger version maintains that everything performs a few computations, some of which encode the others in some relatively unproblematic

way (...)"'. Depending on how the difference between “*large number*” (second type of unlimited pancomputationalism) and “*some*” or “*few*” (in limited pancomputationalism) are defined, all versions (except for the “every system-every computation” type of unlimited pancomputationalism) are more or less similar.

As for the *sources* of pancomputationalism, Piccinini identifies several. One holds that it is “a matter of *relatively free interpretation*” which computation a system performs. This may well be true of human computational devices like fingers, pebbles, abacuses, and computers even though interpretations once chosen are kept constant (and thus no longer free) in order to allow social communication of computational results. Another source of pancomputationalism is the *causal structure* of the physical world. That claim goes one step further than the first one about free interpretation, actually searching for the basis of free interpretation. We can freely chose systems used for calculation/computation, but the computational operations performed are predictable because of the laws of physics which guarantee that physical systems behave in the same way and according to physical laws so that we can predict their behavior.

Info-computationalism is in Piccinini’s scheme kind of limited computationalism based on the third source:

*“A third alleged source of pancomputationalism is that every physical state carries information, in combination with an information-based semantics plus a liberal version of the semantic view of computation. According to the semantic view of computation, **computation is the manipulation of representations**. According to information-based semantics, a representation is anything that carries information. Assuming that every physical state carries information, it follows that every physical system performs the computations constituted by the manipulation of its information-carrying states (cf. Shagrir 2006). Both information-based semantics and the assumption that every physical state carries information (in the relevant sense) remain controversial” [57].*

The use of the word “manipulation” seems to suggest a conscious intervention (“the practice of manipulating”, Wiktionary), while computation in general is a dynamical process that drives (through the interaction mechanisms) changes in informational structures. Notwithstanding Piccinini’s skepticism there are well established theories in computer science, see [30] which do exactly the job of connecting computational processes and informational structures as suggested by Info-computationalism.

10. Discrete vs. Continuous and Digital vs. Analog

One of the frequent criticisms of computational approaches is based on the understanding that computation always must be discrete and that some continuous processes (as human mind) can never be adequately represented by computational models. Here, several confusions act together, and it is useful to disentangle them.

Wolfram and Fredkin assume that the universe on a fundamental level is a discrete system, and so a suitable basis for an all-encompassing digital computer. But the hypothesis about the discreteness of the physical world is not decisive for pancomputationalism/naturalist computationalism. As is well known, besides digital there are analog computers. On a quantum-mechanical level, the universe

performs computation [25] on characteristically dual wave-particle objects. There are interesting philosophical connections between digital and analog processes. Moreover, even if some representations may be purely digital (and thus conform to Pythagorean ideal of number as a principle of the world)—computation in the universe is performed on many different levels of organization, such as bio-computing, membrane computing, spatial computing, morphological computing, *etc.* which presuppose continuous representations. Toffoli in [58] (p. 374) claims that whenever there is an infinite lattice there is continuity, which means the dynamics of continuous functions. Calude [58] (p. 376) supports the similar view arguing that e.g. in non-standard analysis both continuum and discreteness appears. In *The Age of Intelligent Machines*, Kurzweil discusses the question if the ultimate nature of reality is analog or digital, and points out that:

“as we delve deeper and deeper into both natural and artificial processes, we find the nature of the process often alternates between analog and digital representations of information. As an illustration, I noted how the phenomenon of sound flips back and forth between digital and analog representations. (...) At a yet deeper level, Fredkin, and now Wolfram, are theorizing a digital (i.e., computational) basis to these continuous equations. It should be further noted that if someone actually does succeed in establishing such a digital theory of physics, we would then be tempted to examine what sorts of deeper mechanisms are actually implementing the computations and links of the cellular automata. Perhaps, underlying the cellular automata that run the Universe are yet more basic analog phenomena, which, like transistors, are subject to thresholds that enable them to perform digital transactions” [59].

Now it should be emphasized that “computational” is not identical with “digital”, and Maley [7] demonstrates that it is necessary to distinguish between analog and continuous, and between digital and discrete representations. Even though typical examples of analog representations use continuous media/formats, this is not what makes them analog; rather, it is the relationship that they maintain with what they represent. Similar holds for digital representations. The lack of proper distinctions in this respect is the source of much confusion, especially in cognitive science.

Lloyd makes a statement equivalent to Calude’s [58] about digital/analog in case of quantum mechanics:

“In a quantum computer, however, there is no distinction between analog and digital computation. Quanta are by definition discrete, and their states can be mapped directly onto the states of qubits without approximation. But qubits are also continuous, because of their wave nature; their states can be continuous superpositions. Analog quantum computers and digital quantum computers are both made up of qubits, and analog quantum computations and digital quantum computations both proceed by arranging logic operations between those qubits. Our classical intuition tells us that analog computation is intrinsically continuous and digital computation is intrinsically discrete. As with many other classical intuitions, this one is incorrect when applied to quantum computation. Analog quantum computers and digital quantum computers are one and the same device” (Lloyd, 2006).

Thus establishing a digital basis for physics at certain level of granularity, will not resolve the philosophical debate as to whether physical universe is ultimately digital or analog. Nonetheless, establishing a feasible computational model of physics would be a major achievement.

11. Continuum as a Result of Interaction. Asynchronous Communicating Systems

Let us start from the assertion that a dichotomy exists between the discrete and continuous nature of reality in classical physics. From the cognitive point of view it is clear that most of the usual dichotomies are coarse approximations. They are useful, and they speed up our perception and reasoning considerably. Following Kant, however, we can say that “Ding an Sich” (thing-in-itself) is nothing we have knowledge of. This is also true in the case of the discrete-continuous question. Human cognitive categories are the result of natural evolutionary adaptation to the environment. Given the bodily morphology we have, they are definitely strongly related to the nature of the physical world which we live in, but they are by no means general tools for understanding the universe as a whole at all levels of organization and for all types of phenomena that exist. If we adopt the dichotomy as our own epistemological necessity, how could the continuum/discrete universe be understood?

In what follows I will argue that discrete and continuous are dependent upon each other—that logically there is no way to define the one without the other. Let us begin by assuming that the basic physical phenomena are discrete. Let us also assume that they appear in finite discrete quanta, packages, amounts or extents. If the quanta are infinitely small then they already form a continuum. So in order to get discretization, quanta must be finite.

Starting with finite quanta one can understand the phenomenon of continuum as a consequence of the processes of (asynchronous) communication between different systems. Even if the time interval between two signals that one system produces has always some definite value different from zero, (discrete signals), two *communicating* phenomena can appear arbitrarily in time, so that the overlap is achieved, which means that a continuum is realized in a communicative (interactive) process such as computation. It is interesting to note the related insight of Sloman about computation in asynchronous communicating systems. Sloman [52] points out, that concurrent and synchronized machines are equivalent to sequential machines (Turing machine is such), but some concurrent machines are asynchronous. Turing machines can in principle approximate machines with continuous changes, but cannot implement them exactly. This argument may also be found in [60]. A continuous machine with non-linear feedback loops may be chaotic and impossible to approximate discretely, even over short time scales. If a machine is composed of asynchronous concurrently running subsystems and their relative frequencies vary randomly then such a machine cannot be adequately modeled by Turing machine [4].

12. A Paradigm Shift in Modeling of Computing

Abramsky summarizes the process of changing paradigm of computing as follows:

*“Traditionally, the dynamics of computing systems, their unfolding behavior in space and time has been a mere means to the end of computing the **function** which specifies the algorithmic problem which the system is solving. In much of contemporary computing, the*

*situation is reversed: the purpose of the computing system is to exhibit certain **behaviour**. (...) We need a theory of the dynamics of informatic processes, of interaction, and information flow, as a basis for answering such fundamental questions as: What is computed? What is a process? What are the analogues to Turing completeness and universality when we are concerned with **processes and their behaviours, rather than the functions which they compute?**” [30].*

This also suggests a possibility that a lack of an “adequate structural theory of processes” may explain the lack of fundamental progress in theory of complexity. For example, Calude in [58] argues about the P vs. NP problem that it is “a very challenging and deep and interesting mathematical question”, but “one that has no computer science meaning whatsoever. For the simple fact that P is not an adequate model of feasible computation, and there are lots of results—both theoretical and experimental—which point out that P does not model properly what we understand as feasible computation. Probably the simplest example is to think about the simplex algorithm which is exponentially difficult, but works much better in practice than all known polynomial solutions”. This is in accordance with the view expressed by Abramsky that we lack an adequate theory of computation and the emphasis on the importance of agent models and explicit dynamics. Game Semantics and Geometry of Interaction are developed as dynamical theories of computation explicitly based on interaction between agents, and they expose both geometrical and logical structure of information flow.

According to [30], there is the need for *second generation models of computation*, and in particular process models such as Petri nets, Process Algebra, and similar. The first generation models of computation were originating from problems of formalization of mathematics and logic, while processes or agents, interaction, and information flow are genuine product of the developing of computers and Computer Science. In the *second generation models of computation*, previous isolated systems with limited interactions with the environment are replaced by processes or agents for which the interactions with each other and with the environment are fundamental. As a result of interactions among agents and with the environment, complex behavior emerges. The basic building block of this interactive approach is the agent, and the fundamental operation is interaction. This approach works at both macro-scale (processes in operating systems, software agents on the Internet, transactions, *etc.*) and on micro-scale (program implementation, down to hardware). The same conceptual model is appropriate both for design/synthesis and for description/analysis of both artificial and natural information processing systems. This view of the relationship between information and computation presented in [30] agrees with the ideas of Info-computational naturalism which is based on the same understanding of computation and its relation to information.

13. Complexity and Emergence

If computation is understood as a physical process, if Nature computes with physical bodies as objects (informational structures) and physical laws governing process of computation, then the computation necessarily appears on many different levels of organization. Natural sciences provide such a layered view of Nature. One sort of computation processes will be found on the quantum-mechanical levels of elementary particles, atoms and molecules; yet another on the level of classical physical objects. In the sphere of biology, different processes (computation, information

processing) are going on in biological cells, tissues, organs, organisms, and eco-systems. Social interactions are governed by still another kind of communicative/interactive processes. In short, computation on a given level of organization is implementation of the laws that govern the interaction between different parts that constitute that level. Consequently, what happens on every next level of organization is that a set of rules governing the system switches to the new level. How exactly this process goes on in practice, remains yet to learn. Recently, simulation tools are being developed which allow for study of the behavior of complex systems modeled computationally. For the analysis of the time development of dynamic systems various simulation techniques are developed, from purely mathematical approaches, e.g., equation based modeling, simulated by iterative evaluation, to formal modeling approaches, such as Petri Nets and Process Algebra together with Object-oriented and Agent-oriented simulation methods based on the emulation of constituent system elements.

One of the criticisms of pancomputationalism based on cellular automata is presented by Kurzweil, and concerns complexity. Cellular automata are a surprisingly fruitful model, and they have led to the development of a new kind of scientific method—generative modeling [23]—but they do not evolve beyond a certain limit.

“Wolfram considers the complexity of a human to be equivalent to that a Class 4 automaton because they are, in his terminology, “computationally equivalent.” But Class 4 automata and humans are only computationally equivalent in the sense that any two computer programs are computationally equivalent, i.e., both can be run on a Universal Turing machine. It is true that computation is a universal concept, and that all software is equivalent on the hardware level (i.e., with regard to the nature of computation), but it is not the case that all software is of the same order of complexity. The order of complexity of a human is greater than the interesting but ultimately repetitive (albeit random) patterns of a Class 4 automaton” [61].

My comments are the following: What seems to be missing in cellular automata is the dependence of the rules (implemented in the process of computation) on the underlying structure of the system. In a physical system such as magnetic iron, the mean electromagnetic field of the whole system affects each of its parts (magnetic atoms in a crystal lattice), so interactions are not only local, with the closest neighbors but reflect also global properties of the system. Cellular automata are synchronously updated, which according to Sloman makes them computationally less expressive than systems with asynchronous interactions. Agent based models which are currently developed are generalizations of cellular automata and they can avoid those limitations. They are essentially decentralized, bottom-up, in general asynchronous models. (Synchronous communication where agents exchange information all at the same time is a special case of asynchronous information exchange. The behavior is defined at individual agent level, and the global behavior emerges as a result of interaction among numerous individuals communicating with each other and with the environment. For a quick and concise introduction to Agent Based Models, see for example Castiglione’s article in Scholarpedia.)

In short, solutions are being sought in natural systems with evolutionary developed strategies for handling complexity in order to improve modeling and construction of complex networks of massively parallel autonomous engineered computational systems. The research in the theoretical foundations of

Natural computing is needed to improve understanding on the fundamental level of computation as information processing which underlie all of the computing in nature.

14. Summary

The mutually interrelated objectives of this paper were to:

- Offer computational interpretation of information dynamics of the info-computational universe;
- Suggest the necessity of generalization of the models of computation beyond the traditional Turing machine model and acceptance of “second generation” models of computation;
- Elucidate a number of frequent misunderstandings when it comes to the models of computation and their relationships to physical computational systems;
- Argue for info-computationalism as a new philosophy of nature providing a basis for unification of currently disparate fields of natural, formal and technical sciences;
- Answer some of the most prominent criticisms of naturalist computationalism.

The developments supporting info-computational naturalism are expected from among others Complexity Theory, Theory of Computation (Organic Computing, Unconventional Computing), Cognitive Science, Neuroscience, Information Physics, Agent Based Models of social systems and Information Sciences as well as Bioinformatics and Artificial Life [5].

References

1. Abramsky, S. Information, processes and games. In *Philosophy of Information*; van Benthem, J., Adriaans, P., Eds.; North Holland: Amsterdam, The Netherlands, 2008; p. 517.
2. Crnkovic, G.D. *Information and Computation Nets. Investigations into Info-Computational World*; VDM Verlag Dr. Müller: Saarbrücken, Germany, 2009.
3. Crnkovic, G.D. *Investigations into Information Semantics and Ethics of Computing*; Mälardalen University Press: Västerås, Sweden, 2006; pp. 1–133.
4. Crnkovic, G.D. Significance of models of computation from turing model to natural computation. *Minds Mach.* **2011**, *21*, 301–322.
5. Crnkovic, G.D.; Müller, V. *A Dialogue Concerning Two World Systems: Info-Computational vs. Mechanistic*; Crnkovic, G.D., Burgin, M., Eds.; World Scientific Publishing Co., Inc.: Singapore, 2010.
6. Greco, G.M.; Paronitti, G.; Turilli, M.; Floridi, L. How to do philosophy informationally? In *WM2005: Professional Knowledge Management*; Althoff, K.-D., Dengel, A., Bergmann, R., Nick, M., Roth-Berghofer, T., Eds.; Springer-Verlag: Kaiserslautern, Germany, 2005; pp. 623–634.
7. Maley, C.J. Analog and digital, continuous and discrete. *Philos. Stud.* **2010**, *155*, 117–131.
8. Van Benthem, J.; Adriaans, P. *Philosophy of Information*; North Holland: Amsterdam, The Netherlands, 2008.
9. Dretske, F. *Knowledge and the Flow of Information*; Cambridge University Press: New York, NY, USA, 1999.

10. Barwise, J.; Seligman, J. *Information Flow: The Logic of Distributed Systems*; Cambridge University Press: Cambridge, UK, 1997.
11. Burgin, M. *Theory of Information: Fundamentality, Diversity and Unification*; World Scientific Publishing Co., Inc.: Singapore, 2010.
12. Floridi, L. *The Philosophy of Information*; Oxford University Press: Oxford, UK, 2011.
13. Floridi, L. What is the philosophy of information? *Metaphilosophy* **2002**, *33*, 123–145.
14. Van Benthem, J. *Logical Dynamics of Information and Interaction*; Cambridge University Press: Cambridge, UK, 2011.
15. Hofkirchner, W. *Emergent Information. An Outline Unified Theory of Information Framework*; World Scientific Publishing Co.: Singapore, 2011.
16. Hofkirchner, W. *Twenty Questions About a Unified Theory of Information: A Short Exploration into Information from a Complex Systems View*; Emergent Publications: Litchfield Park, AZ, USA, 2010.
17. Dodig Crnkovic, G., Mueller, V. A dialogue concerning two world systems: info-computational vs. mechanistic. In *Information and Computation*; Dodig Crnkovic, G., Burgin, M., Eds.; World Scientific Publishing Co., Inc.: Singapore, 2009; pp. 149–184.
18. Dodig Crnkovic, G. *Biological Information and Natural Computation*. In *Thinking Machines and the Philosophy of Computer Science: Concepts and Principles*; Vallverdú, J., Ed.; Information Science Reference (IGI Global): Hershey, PA, USA, 2010.
19. Floridi, L. A defence of informational structural realism. *Synthese* **2008**, *161*, 219–253.
20. Sayre, K.M. *Cybernetics and the Philosophy of Mind*; Routledge & Kegan Paul: London, UK, 1976.
21. Zuse, K. *Calculating Space*; Friedrich Vieweg & Sohn: Braunschweig, Germany, 1969.
22. Fredkin, E. Finite Nature. In *Proceedings of the XXVIIth Rencontre de Moriond*, Les Arcs, Savoie, France, 22–28 March 1992.
23. Wolfram, S. *A New Kind of Science*; Wolfram Media: Champaign, IL, USA, 2002.
24. Chaitin, G. Epistemology as information theory: From leibniz to Ω . In *Computation, Information, Cognition—The Nexus and the Liminal*; Dodig Crnkovic, G., Ed.; Cambridge Scholars Publisher: Newcastle, UK, 2007; pp. 2–17.
25. Lloyd, S. *Programming the Universe: A Quantum Computer Scientist Takes on the Cosmos*, 1st ed.; Knopf: New York, NY, USA, 2006.
26. Dodig Crnkovic, G. Info-computational philosophy of nature: an informational universe with computational dynamics. In *Festschrift for Søren Brier*; Sørensen, B., Cobley, P., Thellefsen, T., Eds.; CBS University Press: Copenhagen, Denmark, 2011.
27. Dodig Crnkovic, G.; Burgin, M. *Information and Computation*; World Scientific Publishing Co., Inc.: Singapore, 2011.
28. Dodig Crnkovic, G. The Cybersemiotics and Info-computationalist research programmes as platforms for knowledge production in organisms and machines. *Entropy* **2010**, *12*, 4878–4901.
29. Bateson, G. *Steps to an Ecology of Mind: Collected Essays in Anthropology, Psychiatry, Evolution, and Epistemology*; University Of Chicago Press: Chicago, IL, USA, 1972; pp. 448–466.
30. Abramsky, S. Information, processes and games. In *Philosophy of Information*; van Benthem, J., Adriaans, P., Eds.; North Holland: Amsterdam, The Netherlands, 2008; pp. 483–549.

31. Rideau, S.; Winskel, G. Concurrent Strategies. In *Proceedings of the Twenty-Sixth Annual IEEE Symposium on Logic in Computer Science (LICS 2011)*, Toronto, Ontario, Canada, 21–24 June 2011.
32. Lewis, D. General semantics. *Synthese* **1970**, *22*, 18–67.
33. Church, A. An unsolvable problem of elementary number theory (abstract). *Bull. Am. Math. Soc.* **1935**, *41*, 332–333.
34. Church, A. An unsolvable problem of elementary number theory. *Am. J. Math.* **1936**, *58*, 348–363.
35. Burgin, M. *Super-Recursive Algorithms Monographs*; Springer-Verlag: New York, NY, USA, 2004.
36. Dodig Crnkovic, G. Significance of models of computation from turing model to natural computation. *Minds Mach.* **2011**, *21*, 301–322.
37. Kampis, G. *Self-Modifying Systems in Biology and Cognitive Science: A New Framework for Dynamics, Information, and Complexity*, 1st ed.; Pergamon Press: Amsterdam, The Netherlands, 1991.
38. Rozenberg, G.; Kari, L. The many facets of natural computing. *Commun. ACM* **2008**, *51*, 72–83.
39. Pfeifer, R.; Fumiya, I. Morphological computation: Connecting body, brain and environment. *Jan. Sci. Mon.* **2005**, *58*, 48–54.
40. Pfeifer, R.; Fumiya, I.; Gabriel, G. Morphological computation for adaptive behavior and cognition. *Int. Congr. Ser.* **2006**, *1291*, 22–29.
41. Dodig Crnkovic, G. *Epistemology as Computation (Information Processing)*; Calude, C., Ed.; World Scientific Publishing Co., Inc.: Singapore, 2007; pp. 263–279.
42. Dodig Crnkovic, G.; Stuart, S. *Computation, Information, Cognition: The Nexus and the Liminal*; Cambridge Scholars Publisher: Newcastle, UK, 2007.
43. Dodig Crnkovic, G. Knowledge Generation as Natural Computation. In *Proceedings of the International Conference on Knowledge Generation, Communication and Management (KGCM 2007)*, 8–11 July 2007, Orlando, FL, USA, 2007.
44. Dodig Crnkovic, G. Knowledge generation as natural computation. *J. Syst. Cybern. Inf.* **2008**, *6*, 12–16.
45. Dodig Crnkovic, G. Info-Computationalism and Philosophical Aspects of Research in Information Sciences (keynote). In *Philosophy's Relevance in Information Science*; Hagengruber, R., Ed.; Springer: Berlin Heidelberg, Germany, in press.
46. Dodig Crnkovic, G. Knowledge generation as natural computation. *J. Syst. Cybern. Inf.* **2008**, *6*, 12–16.
47. Dodig Crnkovic, G. Semantics of Information as Interactive Computation. In *Proceedings of the Fifth International Workshop on Philosophy and Informatics (WSPI 2008)*, Kaiserslautern, Germany, 1–2 April 2008; Roth-Berghofer, T., Moeller, M., Neuser, W., Eds.; Springer: Berlin, Germany, 2008.
48. Dodig Crnkovic, G. Empirical modeling and information semantics. *Mind Soc.* **2008**, *7*, 157.
49. Dodig Crnkovic, G. The cybersemiotics and info-computationalist research programmes as platforms for knowledge production in organisms and machines. *Entropy* **2010**, *12*, 878–901.
50. The Organic Computing Page. Available online: <http://www.organic-computing.org/> (accessed on 19 July 2011).

51. Cooper, S.B.; Löwe, B.; Sorbi, A. *New Computational Paradigms*; Springer: Berlin, Germany, 2008.
52. Sloman, A. Beyond turing equivalence. In *Machines and Thought: The Legacy of Alan Turing (Vol I)*; Clark, A., Millican, P.J.R., Eds.; Clarendon Press: Oxford, UK, 1996; pp. 179–219.
53. Cooper, S.B.; Löwe, B.; Sorbi, A. *New Computational Paradigms: Changing Conceptions of What is Computable*; Springer: New York, NY, USA, 2008.
54. Fredkin, E. Digital mechanics: An information process based on reversible universal cellular automata. *Phys. D* **1990**, *45*, 254–270.
55. Gardner, M. The game of life, Parts I–III. In *Wheels, Life, and other Mathematical Amusements*; W.H.Fr.eman & Co., Ltd.: New York, NY, USA, 1983; pp. 20–22.
56. Cook, M. Universality in elementary cellular automata. *Complex Syst.* **2004**, *15*, 1–40.
57. Piccinini, G. Computation in physical systems. *The Stanford Encyclopedia of Philosophy*, Fall 2010 ed.; Stanford University. Available online: <http://plato.stanford.edu/> (accessed on 19 July 2011).
58. Zenil, H. *Randomness Through Computation: Some Answers, More Questions*; World Scientific Publishing Co., Inc: Singapore, 2011.
59. Kurzweil, R. *The Age of Spiritual Machines, When Computers Exceed Human Intelligence*; Viking Adult: New York, NY, USA, 1999.
60. Copeland, J.; Sylvan, R. Beyond the universal turing machine. *Australas. J. Philos.* **1999**, *77*, 46–67.
61. Kurzweil, R. *Reflections on Stephen Wolfram's A New Kind of Science*; 2002. Available online: <http://www.kurzweilai.net/articles/art0464.html?printable=1> (accessed on 19 July 2011).

© 2011 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 license (<http://creativecommons.org/licenses/by/3.0/>).