Found Sci (2017) 22:863–881
DOI 10.1007/s10699-016-9503-x

Emergence, Computation and the Freedom Degree Loss Information Principle in Complex Systems

Ignazio Licata[1] **· Gianfranco Minati**[2]

Published online: 1 September 2016

Abstract We consider processes of emergence within the conceptual framework of the Information Loss principle and the concepts of (1) systems conserving information; (2) systems compressing information; and (3) systems amplifying information. We deal with the supposed incompatibility between emergence and computability *tout-court*. We distinguish between computational emergence, when computation acquires properties, and emergent computation, when computation emerges as a property. The focus is on emergence processes occurring within computational processes. Violations of Turing-computability such as non-explicitness and incompleteness are intended to represent partially the properties of phenomenological emergence, such as logical openness, given by the observer's cognitive role; structural dynamics where change regards rules rather than only values; and multi-modelling where multiple non-equivalent models are required to model such structural dynamics. In this way, we validate, from an epistemological viewpoint, models and simulations of phenomenological emergence where the sequence of events constitutes the natural, analogical non-Turing computation which a cognitive complex system can reproduce through learning. Reproducibility through learning is different from Turing-like computational iteration. This paper aims to open a new, non-reductionist understanding of the conceptual relationship between emergence and computability.

Keywords Computational · Emergence · Information · Iteration · Uniqueness · Violations

✉ Ignazio Licata
Ignazio.licata@ejtp.info

Gianfranco Minati
gianfranco.minati@AIRS.it

1 Institute for Scientific Methodology (ISEM), Via Ugo La Malfa 153, 90146 Palermo, Italy

2 Italian Systems Society (AIRS), Via Pellegrino Rossi 42B, 20161 Milan, Italy

1 Introduction

This contribution is addressed to researchers in interdisciplinary areas of systemics where such technical aspects and concepts are often misused, for example, by assuming that emergence and computability are incompatible.

On the contrary, we are convinced that many levels of compatibility exist between emergent phenomena and computation theory, and each level requires proper strategies. In particular, we consider that a one-to-one correspondence, in a reductionist view, between an evolutionary process of a complex system and computational steps is intractable.

We believe that research into high level computational models of systems is a tractable and fruitful approach. We know from transition phase theories (which are largely independent of specific components of the system, so their indications are more general and precious) that away from a critical point, a system finds a rearrangement between its internal dynamics and the environment. Such new phases can be investigated as forms of mesoscopic configurations (Minati and Licata 2013).

As a matter of fact such presumed incompatibility affects, even annuls, the suitability to model and simulate processes of emergence.

There are lively debates about the meaning of emergence, its possible levels and bottom-up nature, and the relationship between the concepts of self-organization and emergence. In this context which, for brevity, we call it here *Philosophy of Computation*, there are various contrasting positions, from the Church–Turing thesis onwards (Batterman 2011; Bedau 2011; Brunner 2002; Butterfield 2011; Chalmers 2006; Crutchfield 1994, 1999).

Emergence and computability are well-known in the literature. Here we outline their relationships, with particular attention to models and simulations of collective phenomena.

The Information Loss principle is introduced with the concepts of (a) systems conserving information; (b) systems compressing information, and (c) systems amplifying information.

Some concepts are then given: Turing machine (TM); Turing's observer; phenomenological emergence; emergent computation with references to theoretical issues such as non-explicitness (non-analyticity) and incompleteness; computational emergence given by properties acquired by processes of emergent computation and simulation, properties analogous to those acquired by processes of phenomenological emergence. Finally we mention problems and proposals for future research.

Many of these issues relate to logical openness[1] (Minati et al. 1998; Licata 2008a); the DYnamic USAge of Models (DYSAM) when different, non-equivalent models are required for albeit incomplete descriptions of the system (Minati and Pessa 2006: 64–88);

[1] A model may be defined as *logically closed* when (1) a formal description of the relationships between all the state variables is available in the model; (2) a complete and explicit description of system-environment interactions is available; (3) all possible structural features and asymptotic states are deducible from the information in (1) and (2). For instance, thermodynamically open systems such as *dissipative structures* may be described by *logically closed* models. Since the description of a given system is equivalent to assertions about its input and output processing, we may distinguish between (a) *logically closed models* related to explicit and completed input processing modalities; and (b) *logically open models* related to non-completed, non-explicit description of the system in case it is impossible to know, *in principle*, how the input–output will be processed. Therefore, it is impossible to know the asymptotic states of the system, if any. Examples are given by a computer program playing a game with a player and by the evolutionary paths of complex systems like ecosystems and biological collective behaviours where the environment plays a crucial role. *Logical open models* may be introduced on the basis of violation of as least one of the three criteria (1), (2), and (3) listed above to describe *logical closed models*.

as well as research on meta-structures (MS), not discussed here, to model processes of emergence (Minati and Licata 2012, 2013; Minati et al. 2013). Comments and examples of possible overlap between emergent computation and computational emergence are presented. There is also an "Appendix" with the conceptual differences between complete and incomplete systems considered relevant in the systemic treatment of processes and problems.

2 The Information Loss Principle

There is a general belief that complex systems are such due to the large number of variables involved. This is not true, because there may be cases in which the system can be effectively described by a handful of macro-variables, as in the classic case of statistical physics, where a huge number of particles can be managed using only a few thermodynamic variables.

On the other hand, non-linear terms in interactions can actually make a problem of only a few bodies unpredictable, e.g., the three-body problem, or drive the entire system towards asymptotically predictable situations, as for the *falls of Feigenbaum* (see, for instance, Sethna 2006).

We focus here on the issue of *degrees of freedom* and the non-trivial dynamics depending upon them. In recent years, 't Hooft suggested the possibility that "weirdness" aspects of quantum physics are due to its emergent characteristics, developing a "thermodynamic" approach to quantum phenomena ('t Hooft 1993, 2015; Blasone et al. 2001; Acosta et al. 2012). In particular 't Hooft assumed that at quantum gravity level, namely "beables",[2] the world is local and deterministic, modelled with cellular automata (CA), while at the level of observables the quantum behaviours emerge as effect of correlations. This implies the loss of local information in favor of the formation of states at laboratory level.

The key idea is extraordinarily simple and of intense interest for every complex dynamics, regardless of their classical or quantum nature. Indeed, in principle, this approach makes use mainly of thermodynamic *arguments* and thus is general, largely independent of system components, which is why the principle of "Information Loss", can effectively realize the program of Anderson "More is Different" (Anderson 1972). In fact, if the emergence is largely independent of microscopic details, this happens precisely because during the process of forming new correlations individual information gradually loses importance.

Consider a population where each entity has a certain number of degrees of freedom, finitely or infinitely enumerable. This assumption is reasonable although not strictly necessary. Simply we focus our interest on a system model having *well-defined* variables, degrees of freedom and relationships with the environment. Interactions and constraints can profoundly change the number of degrees of freedom. We know that a system can belong to three basic categories (or to a mix of these three), which depends on the equation $dI/dt = (1/V)dV/dt$, where I is the 'countable' Shannon-Turing information and V the volume of the phase space (Licata 2008b). We indicate with Vi the initial and with Vf the final volume:

[2] In the sense of Bell (1987).

1. Systems conserving information ($Vi = Vf$): these isolated systems follow the Liouville theorem, the ratio between information variation and volume is zero. A situation of this kind is given by a perfect Brownian motion, with "immutable" aspects on any scale. We note that in this case the interaction between particles is very simple, e.g., elastic collision, and therefore the information on all scales is invariant. This means that an ideal zoom with a renormalization group will show a repetition of the information structures from the average free path of each molecule up to the entire system. In other words, whether we consider a few particles or a large number of the order of Avogadro's number, there is no need to introduce new models and therefore new information to describe behaviours.
2. Systems *compressing* information ($Vf < Vi$): these closed systems undergo a progressive weakening of correlations (entropy), with a finite number of asymptotic states. Here there is a novelty with respect to systems of the first type, given by non-isolation. An exchange from macroscopic to microscopic information is possible, e.g., the melting of ice. Taken together, these two types of systems can be seen from a common point of view: information is always conserved, but it passes from macroscopic forms, related to collective behaviours, to microscopic forms, relative to the particle and its degrees of freedom.
3. Systems *amplifying* information ($Vf > Vi$): these are open systems where the exchange with the environment, driven by interactions and constraints, allows the opposite passage, namely from microscopic to macroscopic information. This means that we must bring into play new models (and therefore a new kind of information) to characterize the emergent structures. Were every variable to acquire degrees of freedom, the information would then remain microscopic. In order to generate macroscopic emergent information, it is necessary to decrease the degrees of freedom, or even that the role of some variables become secondary or even disappears, see Haken's *slaving principle* in Zhang (1991) and Kroger (2014). The loss of information about the microscopic degrees of freedom is then a necessary condition for the emergence of organized structures. Of course, while there is no ambiguity about the concept of microscopic information intended as the number of bits needed to pinpoint a particle event, the notion of macroscopic information is much more vague, but in a virtuous way. The model chosen for a description will depend critically on the type of emerging structure and its dynamics. Note, finally, that the reasoning is largely independent of the type of "statistics" assumed for the constituent parts, even though in recent times it has been shown that "classic" structures may exhibit statistics of a quantum type (Bianconi and Barabási 2001; Pastor-Satorras and Vespignani 2004). The loss of degrees of freedom is closely related to the processes of symmetry breaking, which is a general scenario, proposed to describe emergent processes (for biological systems, see Pessa 2008; for neurodynamics, see Vitiello 2001).

Now, it seems we have all the elements to address the issue of computation in complex systems without falling into facile reductionism. Imagine a Turing Observer (Licata 2006) able to recognize and "count" any degree of freedom of a system, therefore ideally without problems of computational capacity. An observer of this type thus concentrates on the microscopic aspects of a system. After the phase transition, it cannot say how many bits of information are missing (bit by bit, as the Laplace demon would do), and thus is unable to analyze which structures have emerged.

With the expression "Turing Observer" we mean a natural or artificial system able to detect values over a well-defined range. A case can be given by the number of particles, or detections of an eigenvalue, or information related to a degree of freedom. In the case of the authentic complexity of a system, with emerging aspects, this information may be extremely poor. As a matter of fact, radical emergence implies strong overall system readjustments with changes related to the significant variables, as in the case of systems which are assumed to be compatible with or which follow the “slaving principle” introduced by Haken. We therefore speak of (software) “code” because the situation of the Turing Observer is similar to that of a program which can process data with the same code, i.e., for which it was designed to record. In the case of physics, this is a specialized observer on a “code” failing to capture global changes. An elementary but evident case is given by a Turing Observer concentrating on considering positions and moments but not detecting entropy, which is a global property. More generally, in emergent systems, code belonging to micro-, meso- and macro-levels intervenes. The hypothesis which we consider and which is supported by observations (see, for instance, Gorban et al. 2009, 2010; Scheffer et al. 2012), is that in stages where something interesting happens, e.g., pre-transitions, significant correlations between these levels, or “codes” occur. There is a large research on tipping points, indicators of changes underway in the system. On a network, for example, the strengthening or weakening of a node can be a prelude to a critical reassessment, a point of no return to a power law distribution (Sornette 2006; some interesting situations are illustrated with simplicity in Buchanan 2000).

Ideally, however, it is possible to also provide it with (a) details on the interactions between constituents; and (b) information on constraints between system/environment. This endowment should soften its reductionist vocation and its “blindness” towards emerging structures. However, the appearance of the truly cognitive problem shows up here. Actually, the Turing Observer does not have the tools to connect the missing microscopic information with points (1) and (2). In complex systems of any type, it is in general the dynamic play of the boundary conditions which bind organically the different levels of information and therefore allow the modeller to appropriately characterize an aspect of the emerging structure. We also emphasize that the same system, under the same conditions, may have a plurality of emergent histories, particularly in cases where the constituent is an agent provided with an articulate cognitive interface (Minati and Pessa 2006). The importance of the constraints can be understood also with easy examples: the formation of dissipative structures are heavily dependent on boundary conditions; molecules in Brownian motion in free space are reversible in time; in an open small bottle their produce an arrow of time (inside-out the bottle, irreversibly); a quantum particle in a box show the energy level distribution, and so on. In a complex system the boundary condition are more important that the “laws” and this could be a good definition of complex system. In general, the possibility of many stories equivalent was called by biophysicist Mario Ageno “principle of indifference” (Licata 2010): this concept is associated with random fluctuations, i.e., *everything that is not prohibited will occur*, as physicists say. At the end, the behaviour of a single particle or agent is subject to a strong *contextuality*, as it depends on the relations with other constituents, on the global relationship system/environment and it can take part in very different dynamics stories due to the principle of indifference. Strictly reductionist approach proves its short-sightedness; the *complexity is contextuality* (Kitto 2014).

This is why, in our opinion, a software program predicting emergence is not conceptually possible since, from the computational point of view, phase transitions, symmetry breaking and loss of information are equivalent to *an unpredictable change of code*. Such news, however, also contains a positive implication. In fact, the reduction of degrees of freedom in general produces a large number of structures, but most of them will be very similar to each other. In other words, classes of macroscopic information will emerge, and this suggests the adoption of useful strategies to investigate them through normal computational approaches. We make here a heuristic conjecture we call *weak ergodicity*, and which is somewhat equivalent to those of equivalence classes in 't Hooft work. In a fundamental Physics context, these notions are very well formalized (Liu and Sun 2001), but in the wild kingdom of "middle way" of complex systems this rigor is unworkable (Laughlin et al. 2000). It is reasonable to assume however that, apart from small fluctuations, the same conditions will produce similar configurations, as we see in simulation of affinity polarization between social agents (Shafee 2010). Our suggestion is that *the emergence is an equivalence class* of system behaviours that show "coherence" with respect to a given choice of variables. Again, in Physics it is possible to define formally the notion of coherence. Here we mean rather a cognitive connotation of "interesting for the observer", as in meta-structures (MS) research program (Minati and Licata 2012, 2013; Minati et al. 2013; Licata and Minati 2010).

In conclusion, only *natural computation* (see Sect. 3) can "follow" the emergence of a process. Although a "program" which can predict *authentic*[3] emergence cannot exist, it is possible to try to explore the results of processes of emergence through computational analysis of the "historical" sequences produced by the system, by assuming that forms of emergence of a system, under given conditions, are classifiable within *macro classes of equivalence*.

Of course, we do not suggest in any way that synergic mixed modelling *increases* predictability. Systems with radical emergence, i.e., with a strong continuous redefinition of the system/environment relationship and with layering of "historian" constraints, are unpredictable by nature, and even a suitable asymptotic evaluation is impossible. We may ask ourselves whether it would have been possible to predict biological evolution from the primordial soup. Of course not! What we want to suggest is that in many cases of practical interest the "predictability" is much less interesting than the "manageability", i.e., speculating on local trends of the system to design *interventions*. It is possible to *control* many "coarse grain" situations by exploring the appearance of correlations in a pre-transition phase or other situation "where something is happening." The meta-structures (MS) are therefore a general framework for the schemas of simulation of certain behaviours in systems which structurally change.

The issues we considered in the previous case (3) can be dealt with using various approaches, such as networks or meta-structures intended as dynamical sets of simultaneous, superimposed and *interfering* structures of interactions characterised by their properties represented using mesoscopic variables.

In the following we deal with the *compatibility* of the conceptual scenario introduced above and the computational approaches for explore emerging structures and their dynamics (Bedau 2011; Chalmers 2006). As we will see the analytic and non-explicit intractability of the scenario in case (c) is not equivalent to *non-computability* having this non-equivalent important, liberating, and epistemological significance for models and

[3] The cases where we consider to have *authentic*, so-called *intrinsic* or *radical*, *emergence* are (a) those in which the relationship with the environment and related processes of acquisition of emergent properties can *not* be modelled a priori in a single formal model. This is the case of *structural dynamics* given by the changing of variables to be considered, i.e., degrees of freedom. Structural dynamics can only be locally modelled by sequences of *unrelated* models. We have sequences of different but coherent different uniqueness; (b) ones in which the bonds are entirely independent of the rules (therefore, literally, "the rules of the game *change*"). In other words when, simply, something *compatible* with the "grid" of the laws *happens,* allowing, however, the emergence of properties.

simulations to be *validated* but conceptually *admissible*, overcoming the supposed incompatibility between emergence and computability *tout-court*.

3 Computational Emergence and Emergent Computation

In this Section we focus on the Turing machine (TM) concept often misunderstood as an "objective" limit for any computational processing.

In this regard, there is the well-known *Church–Turing thesis* for which each function effectively computable by an algorithm can be calculated with an appropriate TM, i.e., *each algorithm is Turing-computable*.

The general features of an algorithm are its *completeness* and *explicitness*, i.e., each step of the program specifies the process's causal chain, thus defining univocally the set of inputs and outputs.

Crucial general questions for understanding the theoretical validity, the levels of suitability, and the approaches for validation of models and simulations start from deciding whether the Turing-computability is a definitive limit of any possible computation or, rather, presumably the bottom level in a possible hierarchy of levels of computation.

Here, we must first mention the concept of *TM with Oracle* introduced by Turing. An oracle is any device (therefore also a TM) able to intervene externally on the operations of another TM to act as a constraint upon the computational process (see, for example, Soare 2009). Furthermore, particular classes of neural networks show non-Turing behaviours and this suggested the utility of constructing a theory of *natural computation* (MacLennan 2004), with sub- and super-Turing behaviours. In hypercomputational approaches (Toby 2006; Syropoulos 2008) special cooperative configurations of TMs investigate the possibility of physically realizing an oracle, which guides computing beyond the Turing limit.

The crucial point is that it is sufficient to have a TM or a set of TMs as constraints where some play the role of Oracle to obtain a computation different from the one obtained with a set of TMs without the Oracle set. In this way, non-Turing computation is not anything strange or exceptional and falls within the concept of classical computing. The difference is due to the Oracles. This is a well-known fact in Physics, equivalent to the diverse behaviours of a physical system when immersed in different environments. *Oracles are the computational environment*. In particular in the natural processes, the metastable arrangements between the internal complexity of a system and environment make that complex systems do not suffer the halting problem. They know exactly when to stop. In contrast to the purely algorithmic situations, complexity and emergence realize an *oracle* based on the fitness of the system. All this is not necessarily in conflict with a micro description of the components in Turing terms. It is well known that a set of Turing machines that cooperate is not subject to the constraints of a single TM. Realistically, however, it must be said that the complex system of "every day" are not at all TM: a firm, an organism, a mind and so on. The question of models of computation arises so naturally: the model that works at a level is not working on another level. As in t' Hooft work, where at beables level the AC calculus applies and at observables one run the quantum logic and computing.

In the following we focus on two less controversial aspects, i.e., violations of Turing-computability such as non-explicitness and incompleteness of information.

3.1 Violations

Consider the cases where computational programs, Turing-computable, are explicit and complete whereas the processing based on them is not.

We distinguish between the cases where:

- The program explicitly and completely represented, *is* the processing. The processing is performed through single sequential executions of single instructions.
- The execution of the program builds up the incomplete and non-explicit processing, even in *different*, *possible equivalent ways*.

3.1.1 Non-explicitness

A case of non-explicitness is sub-symbolic computing in Artificial Neural Networks (ANN).

For example, we may consider the case of a *learning machine* performed by an ANN, with given inputs and outputs whose correspondences are established by the researcher. The program, the ANN, *represent*s (may we say *computes*?) the corresponding processing (machine learning) which when applied to that input will generate the corresponding output. This processing is represented in a non-analytic way through weighted connections (the weights may change during the process) and levels.

Because of this the processing is considered non-explicit, non-analytically represented, and that is why it is called *sub-symbolic*, whereas the ANN program is an explicit algorithm.

Actually, the whole set of weights and levels used cannot be *zipped*, analytically represented into individual general formulae or functions, being *instead* a dynamical process. The term “zipped” applies to the possibility of algorithmic compression according to Chaitin, Kolmogorov and Solomonoff (see, for instance, Faloutsos and Megalooikonomoum 2007; Li and Vitányi 2009). By extension, a system can be zipped when it is closed, that is, all its significant aspects can be described by a finite group of variables and parameters. This indicates that there are no significant changes in the structure of the system nor in its relationships with the environment. In this case predictability is high. Conversely, a system with strong emergence can not, in this sense, be “zipped”, i.e., described by a single formal model.

The sub-symbolic processing leading to the same result may also not be unique.

However, the repetition of the process, under the same conditions, leads to the *same* result.

3.1.2 Incompleteness

When we speak of *violation* of completeness, we refer to the incomplete identification of computational resources, whatever the programs or hardware set-up. However, incomplete computations will allow processing consistency even though more equivalent configurations of resources are possible. Consider, for example, the conceptual correspondence with self-organization processes, where coherence and consistency are acquired (e.g., swarms or traffic) compared to those where they are *fixed* by a structure (e.g., electronic circuits).

- A typical case is given by the well-known *network paradigm*.
  With the he so-called *Networked Software* (Schmidt et al. 2000), the processing of an

input occurs through a networked *path*, depending on the nature of the input or network *skills* such as: availability, security and efficiency, robustness (fault-tolerance, resistance to attacks, non-intrusion, etc.).
The network paradigm also relates to data representation (nodes) and then the fact that their current network may represent a phenomenon at various levels, e.g., accuracy, timing, and exhaustiveness. The network may consist of software on a single computing platform, or also distributed over a network of nodes of computing platforms, even identical but more probably *equivalent* with regard to some computational properties, e.g., the Internet. *Because of the variability of the path*, *the processing is considered incomplete*, *whereas the programs used are constituted by complete algorithms*.
- Another case is *Cloud computing*.
Cloud computing (Erl et al. 2013) occurs when the processing is performed by using sets of computational resources *from* populations of hardware and software resources available and which may be (1) identical, but in different states of availability; (2) functionally *equivalent*, but having, for instance, different time or energy effectiveness, being in different states of availability or otherwise; (3) partially equivalent or non-equivalent. The identification of the set of resources can be made in a variety of criteria such as considering safety, efficiency, and optimization.
- Another case is related to the *nature of variables* represented by data.
The variables may be macroscopic classes or clusters whose value and properties do not represent nor are univocally prescribed at the microscopic level. In this case *the result of the processing prescribes properties which may be respected in various ways by data classes and can therefore be considered as equivalent and allowing incompleteness*.

However, the repetition of the process under suitable constraints leads to the same results (consistency of the process).

3.2 Computational Emergence: When Computations Acquire Properties

In general, *computational emergence* consists of *acquiring properties by computations*. Properties are considered *acquired*, *emergent* when they are the outcome of not purposely designed computational processes. Such acquired, emergent properties are *unpredictable* unless the process can be repeated in the same way.

This is different from the execution of purposely designed programs simulating processes of phenomenological emergence and their acquired properties.

A simple case of computational emergence is given by the emergence of patterns deriving from step-by-step computation of cellular automata (CA), as in the *Game of Life* introduced by Conway (Gardner 1970). Patterns emerging from very simple computational rules, as for CA and properties acquired by ANN, are examples of the emergence of properties acquired by processes of computation (see Fig. 1).

This brings us to the classification introduced by Wolfram-Langton (Wolfram 2002; Langton 1990):

- Class 1: evolving into stable, homogeneous structures towards a spatially homogeneous equilibrium state.
- Class 2: evolving into stable or oscillating structures. Local randomness. Evolution toward stable or periodic attractors, always of finite spatial extent.
- Class 3: acquiring chaotic evolution. Spread randomness with unlimited spatial growth of initial patterns.

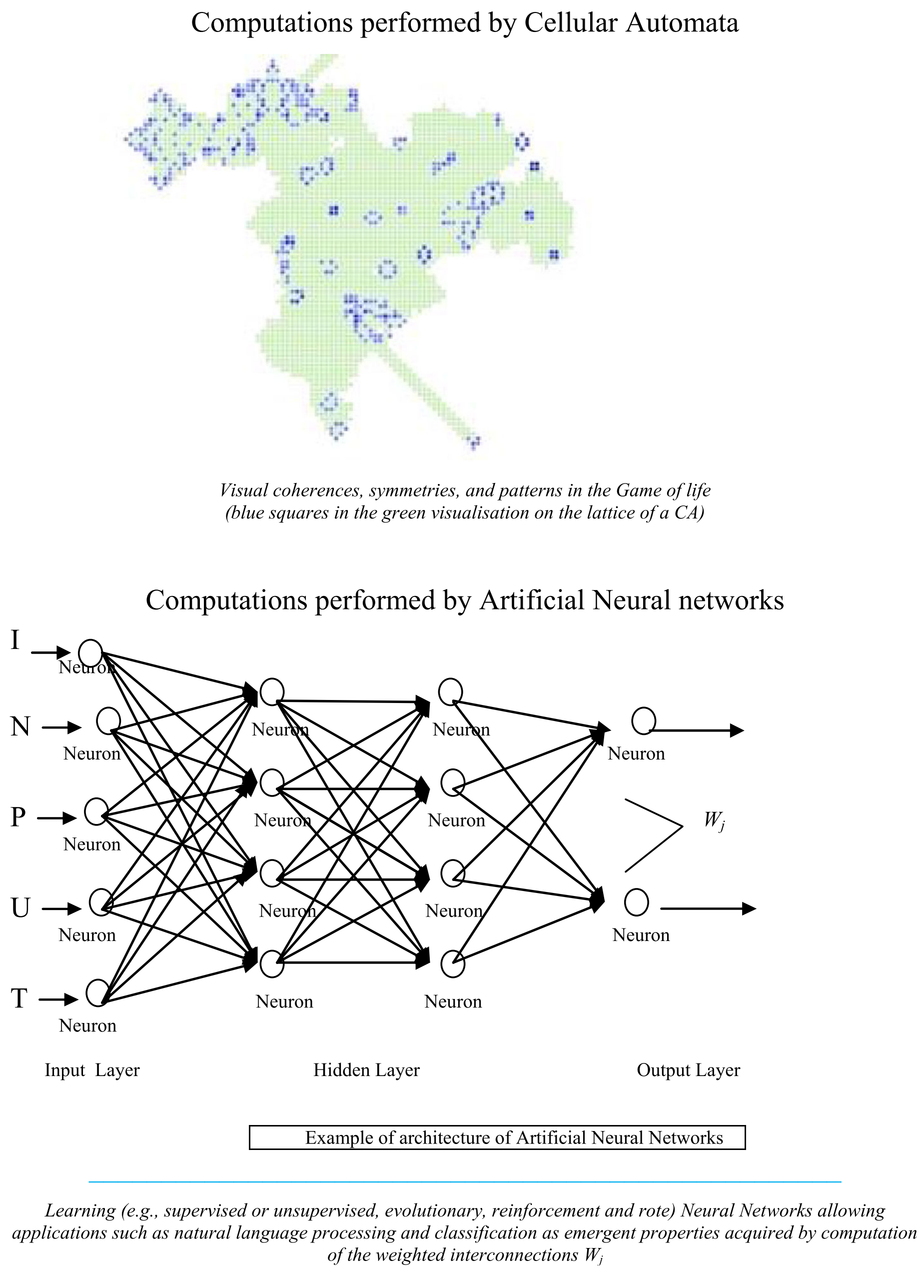

Fig. 1 Examples of computational emergence given by formation of coherences, symmetries, periodicities, structures, patterns and other properties within Turing-compatible processes of computation when they not only represent complex phenomena but they are computational complex phenomena

- Class 4: acquiring emergence of local and surviving dynamic structures, localized patterns of great complexity which can alternatively grow and contract. This is the most interesting class, *at the edge of order and chaos*. Whether such a perfect continuous, dynamical equivalent of this class exists is still an open question.

However, when the acquired property is *merely* the ability to compute, the processes of computational emergence and emergent computation (see below) coincide, e.g., usage of ANN to compute statistics, CA to simulate partial differential equations, and CA to model the behaviour of complex systems.

3.3 Emergent Computation: When Computation Emerges

We consider here the case where the property acquired by processes of emergence is the ability to compute (see, for instance, Forrest 1990) as considered, for instance, in Bio-computing (Altman et al. 2014; Simon 2005) and nanocomputing (Anderson and Bhanja 2014; Waldner 2010; MacLennan 2012), performed through molecular, network or genetic processing.

Furthermore, emergent computation can also be a particular case of computational emergence when the property acquired by processes of computation is a computation *of a different kind from the generative computational processes*. In other words, it is as if the computing system is able, because of its extreme complexity, to change the rules of the game which generated it. It is a typical feature of good biological models and it improves the adaptive fitness.

We consider the emergent computation performed by using, for instance, CA able in their turn to perform computations of models otherwise modelled by using differential equations. Since CA can simulate a universal Turing machine their computational power is at least equal to that of a TM. Accordingly, there must be aspects of their evolutionary behaviour which are undecidable, as in the case of Turing machines in the early history of cellular automata theory (see, for instance, Burks 1970). It was discovered that the computational ability emergent from CA is excellent to simulate, on discrete temporal and spatial scales, the behaviour of many physical systems governed by partial differential equations. This means that the discretization of CA aims to produce high level patterns which are described by differential equations. In general, computation itself is considered as emergent (Brunner 2002) when it is implemented through cooperative processes, networks of software resources or processing platforms such as computers, or populations of available processors.

Computational processing occurs through emergent computation (Korotkikh 2014) which occurs in any case having at the bottom TMs, but also through *decentralised*, *distributed* (Fokkink 2014), or *parallel* (Pacheco 2011) processing as for programs running on computing platforms, namely TMs.

Processing, contrary to programs which consist of complete algorithms, is considered to be incompletely represented because of the different paths and sets of the computational resources used, e.g., cloud or network computing.

However, emergent computation is a stable and consistent emergent property as should be those acquired by phenomenological emergence processes (see Sect. 4). Emergent computation may occur in various ways, but it gives the same results *as does an internet search through a browser or sending the same email message several times to the same address which will be reached through different paths*.

There are various cases of phenomenological emergence simulations performed through emergent computation, such as pedestrian flow simulation by CA (see, for instance, Ishii and Morishita 2010; Nagatani 2012) or complex systems in general (see, for instance, Hoekstra et al. 2010).

On the other hand, we point out how properties of phenomenological emergence may be acquired by processes of simulation of emergence, *performed through non-emergent computation*.

A typical case, extensively considered in the literature, regards the simulation of properties of collective behaviours. Here, the software generates phenomena having some characteristics in silicio of phenomenological emergence. A classic example is the approach introduced by Reynolds (1987) and used in software and in various approaches, developed with non-emergent computation following other methods of a stochastic nature investigating the formation of, and behaviour in, fish schooling (Aoki 1982; Takagi et al. 2004). For a review of other approaches to simulation of the more general *collective motion* see Vicsek and Zafeiris (2012).

We consider here the simulation of a flock-like collective behaviour with interacting agents acquiring:

(a) Mathematical properties observed in real flocks, such as scale invariance or topological properties (Ballarini et al. 2008; Cavagna et al. 2010), power-laws (Pinto et al. 2002); where *incompleteness* is related to some properties, *necessity* can be detected in a phenomenological way, but *sufficiency* should be demonstrated.
(b) *Behavioural phenomenological* properties of real flocks, such as emergence of patterns where the observer and a suitable level of description are required. Internal events and external perturbations are considered as countable, if not finite in number (to avoid turbulence).

Repetition of the same process of simulation of phenomenological emergence generates the *same* simulation (in which one uses, for instance, random elaborations in the same way as the Mersenne Twister algorithm for generating pseudo-random numbers), unless the usage of random parameters affects the initial conditions. See Table 1 for a concise view.

In conclusion, we can say that the TM level is a molecular-atomic level of computation, according to statistical mechanics language. Computational emergence has strong limits of complexity, because the emergent levels are greatly influenced by the elementary level of computation. That is why CA are so useful for simulating fluid-dynamics processes, with a very accurate degree of precision. On the other hand, emergent computation is based on a

Table 1 Computational emergence and emergent computation

Computational emergence	Emergent computation
Computational emergence consists of the *acquisition of properties by processes of computation*	*Emergent computation* consists of the *acquisition of the property to compute through processes of emergence* of which computational emergence is a particular case
Properties acquired are, for instance, coherences, symmetries and patterns	The property acquired by the process of emergence is the ability to compute
Examples	Examples
Classic examples are given by the establishment of deterministic complexity, such as deterministic chaos and the evolution of CA	The computation itself is considered as emergent when it is implemented through CA, cooperative processes, networks of software resources on populations of available processors, or cloud computing

different cognitive design: the high level of system patterns are the constructive key of the computational model. In this sense, it is the pattern which leads the configuration and no longer the elementary constituents of computation. Once again, as we have already said, the collective side calls into play the computational environment.

4 Phenomenological, Non-computational Emergence

The concept of emergence in the literature refers to the acquisition of novelty, in short, showing new properties which are non-deducible and not predictable from the previous ones, such as the establishment of multiple, local synchronizations, or coherences between phase transitions and self-organization processes (Goldstein 1999; Minati and Licata 2012, 2013; Minati et al. 2013; Minati and Pessa 2006; Ronald et al. 1999; Ryan 2006). From what we have said, there are forms of emergence which cannot be reduced to computational processes. We refer to phenomenological (or observational) emergence, events which can be detected experimentally, which often modify the nature of a system. It can neither be described algorithmically nor individuated in a univocal way by means of a precise set of inputs/outputs. Obviously, there is nothing magic about this. Once again we have to keep in mind the physical difference between laws (classes of events) and limiting conditions (initial and boundary conditions). The apparent non-algorithmic nature is equivalent to saying that a specific limiting condition has changed the system nature and the inter-relationships between its components. All of which cannot be forecasted by an algorithm. It means neither renouncing an ideal chain of causes and effects nor, where possible, its computational representation. These cases are very few indeed and would imply a huge quantity of information both in a synchronic sense (all the information related to a single event at time t) and diachronic sense (all the information related to the history of that single event at time Δt).

This is what usually happens with cognitive systems or systems where cognitive interaction between agents is implicit. Other examples of phenomenological emergence include Ecosystems, road traffic, flows of data and telecommunications signals, industrial districts, the morphology of cities, and the behaviour of markets.

However, the *non-computational nature* of phenomenological emergence processes should not be identified with the *absolute* absence of *any* computational processes. In collective behaviours the individual, unique sequences and combinations of events *correspond* to rules of interaction and parameters used in the individual histories by each agent (Minati and Licata 2013). It is a matter of *individual histories* which can not be standardized nor *zipped* into equations. The only choice is to try alternative approaches based on the search for significant configurations. Empirically, it is a work in stages: emerging collective behaviour are observed (macro); assumptions about the type of relationships between agents are done (micro); so, the system is explored using computational simulations whose goal is to show a mesoscopic level behaviours that combine the conjectured micro level with the observed macro level. Despite the fact that such work implies a vast amount of data (configuration, histories at the edge of emergence) it should not be mistaken for statistics. It is instead necessary to create strategies to selectively explore vast amounts of heterogeneous data. An example is the recent work of Barabási (2011). In these cases, our *computational design* is based on conjecture about the nature of examined processes. It is what we mean by “observer’s choice” (all models are built by observers with a purpose!). This is a human activity. Currently the machines do not do best (Claude and Longo 2016).

The idea is to work with clusters, largely mesoscopic, which correlate the macro-level with the micro-level. This recalls the *order parameters* of Synergetics (Haken 1987, 1988), but with an essential difference: synergetics treats systems which are highly non-linear, but perfectly describable by evolutionary equations, while in this case, we have to find the emergent tendencies of hidden logics beneath the apparent “randomness” of a multiplicity of interconnected events. This is the typical challenge of financial markets (Gorban et al. 2009).

Modelling and simulating related to phenomenological emergence are *incomplete* because *not all processes* are represented or even *known*, such as individual cognitive processing, inhomogeneous perturbations and turbulences, due to heterogeneity, all with their own dynamics. Moreover, the dynamic and coherent system of processes, constituting the process of emergence (e.g., ecosystems) establishes so-called *natural computation* consisting of the physical process relating the input values to the output values of specific variables. An instrumental example is work in a laboratory, which can be seen as a piloted natural computation (let *nature compute*).

Finally, incompatibility between emergence and iteration seems related to a necessary *uniqueness-non repeatability-non predictability* of emergence processes. Thus we should consider *non-equivalence* as the typical mark of unpredictability and un-repeatability of emergence processes. The alternative would be to assume that the same states under the same conditions univocally fix the evolutionary fate of a system. It is clear that such an assumption is completely artificial since in cases of effective phenomenological emergence the boundary conditions are never the same. This makes it necessary to adopt a descriptive level based on observer choices. This is related to the well-known halting problem in TMs. Halting a computation is a problem of decision which cannot be presented in terms of low level computation, but requires considering at a high level the meaning of the emergent patterns. It implies a choice by the programmer in modelling activities, but it is also a known process in Nature. In fact, the whole of biological evolution may be regarded as an emergent phenomenon where imposing the limiting conditions has spontaneously solved the halting problem. Phenomenological emergence can be considered as *subsequent uniqueness* by *coherence* of subsequent phase transitions (Minati and Licata 2012, 2013; Minati et al. 2013).

Step-by-step computability may correspond to step-by-step *frames* of a radical emergence process where the *sequence* is *the computation itself*, is non-Turing, but a natural, analogue one, possibly repeatable through *learning* by a cognitive complex system. *Repeatability by learning* is different from Turing-like computational iteration.

5 Conclusions

Long tradition has spread two conceptual opposite visions of the relationship between computation and emergence. The first argues that there is no emergence which cannot be “captured” by a computational schema. The philosophy behind this view is purely reductionist and based on the idea that any natural process may correspond to a suitable, however *complicated*, algorithm based on the “right” choice of the information variables. This is, in practice, equivalent to asserting that there is no *real* emergence.

This first vision operates an identification *tout court* of "physical law" with "algorithm". The misadventures of *strong* Artificial Intelligence (AI) should be sufficient to show the limits of this vision, generally supported by computationalists.

On the opposite side, the second conceptual vision argues, as many physicists of emergence believe, that natural processes are too complex to be "zipped" into an algorithm. In support of this position it should be noted that a complex system is generally a process in which numbers and relationships between the entities rapidly change, as well as the relationship with the environment and its constraints, in an inextricable weaving of micro, meso and macro levels. In general, the description of a system of this type requires a multi-modelling strategy targeted to the plurality of emergent behaviours.

In this paper we have tried to provide a new basis for the relationships between emergence and computation, inspired by the scheme of 't Hooft on the emergence of QM from beables modelled through CA in a Hilbert space applied to complex systems. Indeed, in a world of strict local rules it would be impossible to obtain authentic emergence, and what we call as such is only a "practical" strategy due to the fact that we can not follow the progress of each constituent step by step and therefore of each overall state of the system. However, the real world is much more complex than that. The characteristic complexity of a system derives from the fact that during its evolutionary history it loses degrees of freedom through dissipation, and a considerable part of this contributes to the creation of emergent structures. Strictly speaking, a reductionist and algorithmic approach could not even predict a simple asymptotic state such as achieving maximum entropy in an isolated system! The loss of degrees of freedom, in computational terms, is equivalent to a drastic *change of code*, which makes the single TM "blind" when faced with phenomena of emergence and requires multi-modelling strategies.

The world processes information, but only a small part of this qualifies for the Turing scheme. Genuinely hyper-computational aspects (i.e., beyond the single TM), appear even within traditional Turing Computation, when one of the TMs, or in a real system relations with the environment, behaves as oracle acting on constraints. As always in complex systems, constraints are more important than the laws which should be seen as "grids" of possibilities which alone, therefore, do not uniquely define the "fate" of a system. This is particularly true for systems which have been defined as characterised by high levels of logical openness (Licata 2008a; Minati et al. 1998; Minati and Pessa 2006: 111–112) and which, within this context, can be seen as systems with sudden changes in the general organisation of degrees of freedom.

If this seems like *bad news* for the relationship between emergence and computation, it is only so when we think in terms of an impossible *algorithmization* of the world, without any considerations of scale or environment. But there are strong indications that in both natural and artificial systems emerging forms occur in classes, and this allows refined forms of simulation by the *learning* of a cognitive complex system. Repeatability through learning is something other than Turing-like computational iteration. In conclusion, the pact between emergence, computation, and complexity is restored if we keep in mind the fundamental lesson of Von Foerster that a description of the world is never the passive picture of a *state of affairs*, but the action of *cognitive observing systems* (Von Foerster 1984). We hope these reflections allow one to consider in a new light aspects of compatibility between emergence and computability.

Appendix

In this appendix we will briefly illustrate the concepts of completeness and incompleteness for systems.

For *completeness*, one can say that such a system is completely described by one or a finite number of models. The problem becomes more complicated when we consider dynamic systems, i.e., systems of differential equations describing the evolution of the system, often intractable *analytically* and thus we have to look at global properties of a system (families of solutions and structural facets).

For *incompleteness*, we have to use multiple non-equivalent models *dynamically*, i.e., for different instants and *locally*. In this case, however, models may describe the behaviour of the system in an *incomplete* way as with DYSAM (Minati and Pessa 2006). The dynamic usage of the models corresponds to the *structural dynamics of the system*.

Another way to deal with *completeness* and *incompleteness* consists of considering the *states* (configurations, parameters, etc.) reachable by the system. A system can be understood as *complete* when the number of reachable states is finite.

A system can be understood as *incomplete* when the set of achievable states is non-finite, when the next state is *invented* (not *chosen* among the available ones) by the system, e.g., by means of broken symmetry, given by *logical openness*, and they are not equivalent to, nor linearly deducible from, previous ones. Today, the focus is on coherence rather than on completeness. In other words, we use cognitive strategies to look for *interesting* configurations. In this case, the problem is not *forecasting* in a mathematical sense, but *betting* in the de Finetti sense (de Finetti 2008; Pavlov and Andreev 2013; Hosni et al. 2011). Computing is an essential tool for evaluating our chances of success.

References

Acosta, D., Fernandez de Cordoba, P., Isidro, J. M., & Santander, J. L. G. (2012). An entropic picture of emergent quantum mechanics. *International Journal of Geometric Methods in Modern Physics, 9*(5), 1250048–1250053.

Altman, R. B., Dunker, A. K., & Hunter, L. (2014). *Biocomputing 2014: Proceedings of the pacific symposium*. Singapore: World Scientific.

Anderson, P. W. (1972). More is different: Broken symmetry and the nature of the hierarchical structure of sciences. *Science, 177*(4047), 393–396.

Anderson, N. G., & Bhanja, S. (2014). *Field-coupled nanocomputing: Paradigms, progress, and perspectives (lecture notes in computer science/theoretical computer science and general issues)*. New York: Springer.

Aoki, I. (1982). A simulation study on the schooling mechanism in fish. *Bulletin of the Japanese Society of Scientific Fisheries, 48*, 1081–1088.

Ballarini, M., Cabibbo, N., Candelier, R., Cavagna, A., Cisbani, E., Giardina, I., et al. (2008). Interaction ruling animal collective behaviour depends on topological rather than metric distance: Evidence from a field study. *Proceedings of the National Academy of Science, 105*(4), 1232–1237.

Barabási, A. L. (2011). *Bursts: The hidden patterns behind everything we do, from your e-mail to bloody crusades*. London: Plume.

Batterman, R. W. (2011). Emergence, singularities, and symmetry breaking. *Foundations of Physics, 41*(6), 1031–1050.

Bedau, M. (2011). Weak emergence and computer simulation. In P. Humphreys & C. Imbert (Eds.), *Models, simulations, and representations* (pp. 91–114). New York: Routledge.

Bell, J. S. (1987). *Speakable and unspeakable in quantum mechanics* (pp. 52–62). Cambridge: Cambridge University Press.

Bianconi, G., & Barabási, A. (2001). Bose–Einstein condensation in complex networks. *Physical Review Letters, 86*(24), 5632–5635.

Blasone, M., Jizba, P., & Vitiello, G. (2001). Dissipation and quantization. *Physics Letters A, 287*(3), 205–210.

Brunner, K. A. (2002). What's emergent in emergent computing? In R. Trappl (Ed.), *Cybernetics and systems 2002: Proceedings of the 16th European meeting on cybernetics and systems research* (pp. 189–192). Vienna: Austrian Society for Cybernetics Study.

Buchanan, M. (2000). *Ubiquity*. London: Wiedenfield & Nicholson.

Burks, A. W. (Ed.). (1970). *Essays on cellular automata*. Urbana (IL): Illinois University Press.

Butterfield, J. (2011). Emergence, reduction and supervenience: A varied landscape. *Foundations of Physics, 41*(6), 920–959.

Cavagna, A., Cimarelli, A., Giardina, I., Parisi, G., Santagati, R., Stefanini, F., et al. (2010). Scale-free correlations in starling flocks. *Proceeding of the National Academy of Sciences of the United States of America, 107*(26), 11865–11870.

Chalmers, D. J. (2006). Strong and weak emergence. In P. Davies & P. Clayton (Eds.), *The re-emergence of emergence* (pp. 244–256). Oxford: Oxford University Press.

Claude, C., & Longo, G. (2016). The deluge of spurious correlations in big data, *Found. of Sc.*, First online http://www.di.ens.fr/users/longo/files/BigData-Calude-LongoAug21.pdf. 07 Mar.

Crutchfield, J. P. (1994). The calculi of emergence: Computation, dynamics and induction. *Physica D, 75*, 11–54.

Crutchfield, J. P. (1999). Is anything ever new? Considering emergence. In G. A. Cowan, D. Pines, & D. Meltzer (Eds.), *Complexity: Metaphors, models, and reality* (pp. 515–537). Cambridge (MA): Perseus Books.

De Finetti, B. (2008). *Philosophical lectures on probability english translation of B. de Finetti's: Filosofia della probabilitá, synthese library*. Vol. 340, New York: Springer.

Erl, T., Puttini, R., & Mahmood, Z. (2013). *Cloud computing: Concepts, technology and architecture*. New York: Prentice Hall.

Faloutsos, C., & Megalooikonomoum, V. (2007). On data mining, compression, and Kolmogorov complexity. *Data Mining and Knowledge Discovery, 15*(1), 3–20.

Fokkink, W. (2014). *Distributed algorithms: An intuitive approach*. Cambridge (MA): MIT Press.

Forrest, S. (1990). *Emergent computation*. Cambridge (MA): MIT Press.

Gardner, M. (1970). Mathematical games—The fantastic combinations of John Conway's new solitaire game "life". *Scientific American, 223*, 120–123.

Goldstein, J. (1999). Emergence as a construct: History and issues. *Emergence, 1*(1), 49–72.

Gorban, A. N., Smirnova, E. V., & Tyukina, T. A. (2009). General laws of adaptation to environmental factors: From ecological stress to financial crisis. *Mathematical Modelling of Natural Phenomena, 4*(6), 1–53.

Gorban, A. N., Smirnova, E. V., & Tyukina, T. A. (2010). Correlations, risk and crisis: From physiology to finance. *Physica A, 389*(16), 3193–3217.

Haken, H. (1987). Synergetics: An approach to self-organization. In F. E. Yates (Ed.), *Self-organizing systems: The emergence of order* (pp. 417–434). New York: Plenum.

Haken, H. (1988). *Information and self-organization. A macroscopic approach to complex systems*. Berlin: Springer.

Hoekstra, A. G., Kroc, J., & Sloot, P. M. A. (2010). *Simulating complex systems by cellular automata*. Berlin: Springer.

Hosni, H., Fedel, M., & Montagna, F. (2011). A logical characterization of coherence for imprecise probabilities. *International Journal of Approximate Reasoning, 52*(8), 1147–1170.

Ishii, H., & Morishita, S. (2010). A learning algorithm for the simulation of pedestrian flow by cellular automata. In S. Bandini & S. Manzoni (Eds.), *Cellular automata, lecture notes in computer science* (pp. 465–473). Berlin: Springer.

Kitto, K. (2014). A contextualised general systems theory. *Systems, 2*(4), 541–565.

Korotkikh, V. (2014). *A mathematical structure for emergent computation*. Dordrecht: Springer.

Kroger, B. (2014). *Hermann Haken: From the laser to synergetics: A scientific biography of the early years*. New York: Springer.

Langton, C. G. (1990). Computation at the edge of chaos: Phase transitions and emergent computation. In S. Forrest (Ed.), *Emergent computation*. Amsterdam: North-Holland.

Laughlin, R. B., Pines, D., Schmalian, J., Stojkovic, B. P., & Wolynes, P. (2000). The middle way. *Proceedings of the National Academy of Sciences*, *97*(1), 32–37. http://www.pnas.org/content/97/1/32.full.pdf.

Li, M., & Vitányi, P. M. B. (2009). *An introduction to Kolmogorov complexity and its applications*. New York: Springer.

Licata, I. (2006). General system theory, link-quantum semantics and fuzzy sets. In G. Minati, E. Pessa, & M. Abram (Eds.), *Systemics of emergence: Research and development* (pp. 723–734). New York: Springer.

Licata, I. (2008a). *La logica aperta della mente*. Torino: Codice Edizioni.

Licata, I. (2008b). Emergence and computation to the edge of classical and quantum systems. In I. Licata & A. Sakaji (Eds.), *Physics of emergence and organization* (pp. 1–25). Singapore: World Scientific.

Licata, I. (2010). Living with radical uncertainty: The exemplary case of folding protein. In I. Licata & A. Sakaji (Eds.), *Crossing in complexity. Interdisciplinary application of physics in biological and social systems* (pp. 1–10). New York, NY: Nova Publishers.

Licata, I., & Minati, G. (2010). Creativity as cognitive design—The case of mesoscopic variables in meta-structures. In Alessandra M. Corrigan (Ed.), *Creativity: Fostering, measuring and contexts* (pp. 95–107). New York: Nova Publishers.

Liu, X. F., & Sun, C. P. (2001). Consequences of 't Hooft's equivalence class theory and symmetry by large coarse graining. *Journal of Mathematical Physics, 42*(8), 3665–3672.

MacLennan, B. J. (2004). Natural computation and non-Turing models of computation. *Theoretical Computer Science, 317*(1–3), 115–145.

MacLennan, B. (2012). Molecular coordination of hierarchical self-assembly. *Nano Communication Networks, 3*(2), 116–128.

Minati, G., & Licata, I. (2012). Meta-structural properties in collective behaviours. *The International Journal of General Systems, 41*(3), 289–311.

Minati, G., & Licata, I. (2013). Emergence as mesoscopic coherence. *Systems, 1*(4), 50–65.

Minati, G., & Pessa, E. (2006). *Collective beings*. New York: Springer.

Minati, G., Penna, M. P., & Pessa, E. (1998). Thermodynamic and logical openness in general systems. *Systems Research and Behavioral Science, 15*(3), 131–145.

Minati, G., Licata, I., & Pessa, E. (2013). Meta-structures: The search of coherence in collective behaviours (without physics). In A. Graudenzi, G. Caravagna, G. Mauri, & M. Antoniotti (Eds.) *Wivace 2013—Proceedings of the Italian workshop on artificial life and evolutionary computation* (pp. 35–42). Electronic proceedings in theoretical computer science. http://rvg.web.cse.unsw.edu.au/eptcs/paper.cgi?Wivace2013.6. Accessed Jan 2016.

Nagatani, T. (2012). Four species CA model for facing pedestrian traffic at rush hour. *Applied Mathematical Modelling, 36*(2), 702–711.

Pacheco, P. (2011). *An introduction to parallel programming*. Burlington (MA): Morgan Kaufmann.

Pastor-Satorras, R., & Vespignani, A. (2004). *Evolution and structure of the internet: A statistical physics approach*. Cambridge: Cambridge University Press.

Pavlov, Y. P., & Andreev, R. D. (2013). *Decision control, management, and support in adaptive and complex systems: Quantitative models*. Hershey (PA): IGI global.

Pessa, E. (2008). Phase transitions in biological matter. In I. Licata & A. Sakaji (Eds.), *Physics of emergence and organization* (pp. 165–228). Singapore: World Scientific.

Pinto, S. E., Lopes, S. R., & Viana, R. L. (2002). Collective behavior in a chain of van der Pol oscillators with power-law coupling. *Physica A, 303*(3), 339–356.

Reynolds, C. W. (1987). Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics, 21*(4), 25–34.

Ronald, E. M. A., Sipper, M., & Capcarrère, M. S. (1999). Design, observation, surprise! A test for emergence. *Artificial Life, 5*(3), 225–239.

Ryan, A. J. (2006). Emergence is coupled to scope, not level. *Complexity, 67*(2), 67–77.

Scheffer, M., Carpenter, S. R., Lenton, T. M., Bascompte, J., Brock, W., Dakos, V., et al. (2012). Anticipating critical transitions. *Science, 338*(6105), 344–348.

Schmidt, D., Stal, M., Rohnert, H., & Buschmann, F. (2000). *Pattern-oriented software architecture volume 2: Patterns for concurrent and networked objects*. New York: Wiley.

Sethna, J. P. (2006). *Entropy, order parameters and complexity*. Oxford: Oxford University Press.

Shafee, F. (2010). Organization and complexity in a nested hierarchical spin-glass like social space. *Electronic Journal of Theoretical Physics (EJTP), 7*(24), 93–130.

Simon, M. (2005). *Emergent computation: Emphasizing bioinformatics*. New York: Springer.

Soare, R. I. (2009). Turing oracle machines, online computing, and three displacements in computability theory. *Annals of Pure and Applied Logic, 160*(3), 368–399.

Sornette, D. (2006). *Critical phenomena in natural sciences: Chaos, fractals, self-organization and disorder: Concepts and tools*. Heidelberg: Springer.

Syropoulos, A. (2008). *Hypercomputation. Computing beyond the Church–Turing barrier*. New York: Springer.

‘t Hooft, G. (1993). Dimensional reduction in quantum gravity. In A. Ali, J. Ellis, & S. Randjbar-Daemi (Eds.) *Salamfestschrift: A collection of talks. Series in 20th century physics, Vol. 4* (pp. 284–296). Singapore: World Scientific.

‘t Hooft, G. (2015). *The cellular automaton interpretation of quantum mechanics*. https://arxiv.org/pdf/1405.1548v3.pdf [quant-ph].

Takagi, T., Moritomi, Y., Iwata, J., Nakamine, H., & Sannomiya, N. (2004). Mathematical model of fish schooling behaviour in a set-net. *ICES Journal of Marine Science, 61*(7), 1214–1223.

Toby, O. (2006). Hypercomputation: Computing more than the Turing machine. *Applied Mathematics and Computation, 178*, 143–153.

Vicsek, T., & Zafeiris, A. (2012). Collective motion. *Physics Reports, 517*(3–4), 71–140.

Vitiello, G. (2001). *My double unveiled*. Amsterdam: Benjamins.

Von Foerster, H. (1984). *Observing systems*. Seaside (CA): Intersystems Publications.

Waldner, J. B. (2010). *Nanocomputers and swarm intelligence*. Hoboken (NJ): Wiley.

Wolfram, S. (2002). *A new kind of science*. Champaign (IL): Wolfram Media Inc.

Zhang, W.-B. (1991). The Haken slaving principle and time scale in economic analysis. *Springer Series in Synergetics, 53*, 193–212.

Ignazio Licata Full Prof. Theoretical Physicist, ISEM, Inst. Scient. Methodoloy, Palermo, and IIAMIS, International Institute for Applicable Mathematics and Information Sciences, B.M. Birla Science Centre, India. EJTP Editor in Chief, Edit. Board. In Frontiers in Physics, Entropy. Member of: CiE, Association Computability in Europe, AIEMS, Italian Association of Epistemology and System Methodology. Member of Italian Society of Neuroestetics and New York Science Academy. Research areas: Foundation of Quantum Mechanics, Quantum Dissipative Field Theories, Quantum Cosmology, Neural Nets, Computational Models for Physical, Biological and Cognitive Systems, Theories and Physics of Emergence, EconoPhysics and Management Models.

Gianfranco Minati Mathematics, Doctoral lecturer at the Polytechnic of Milan/Department ‘Building Environment Sciences and Technology; Founder and president of the Italian Systems Society (AIRS), Milan, Italy; Italian Association of Epistemology and System Methodology; President of the Union Européenne de Systémique (UES, association of some European Systems Societies); Member of the scientific committee of Conferences and Systems Societies. Research Areas: Theoretical issues on Systems Science, such as: Logical Openness, Collective Behaviour, Emergence, Ergodicity, Dynamic Usage of Models, Multiple-Systems and Collective Beings, meta-structures. Author of 30 articles, 15 books, editor of seven books, and 34 chapters in edited books http://www.gianfrancominati.net/.