Emergent Open-Endedness from Contagion of the Fittest

Felipe S. Abrahão *
Klaus Wehmuth †
Artur Ziviani ‡
National Laboratory for Scientific Computing (LNCC)
25651-075 – Petropolis, RJ – Brazil
**fsa@lncc.br*
†*klaus@lncc.br*
‡*ziviani@lncc.br*

This paper presents a theoretical investigation of the general problem of emergent irreducible information in networked populations of computable systems. In particular, we narrow our scope to study this problem in algorithmic networks composed of randomly generated Turing machines that follow a susceptible-infected-susceptible contagion model of imitation of the fittest neighbor. We show that there is a lower bound for the stationary prevalence (i.e., the average density of infected nodes by the fittest nodes) that triggers expected (local) emergent open-endedness, that is, that triggers an unlimited increase of the expected local emergent algorithmic complexity (or information) of a node as the population size grows. In addition, we show that static networks with a power-law degree distribution following the Barabási–Albert model satisfy this lower bound and thus display expected (local) emergent open-endedness.

Keywords: emergence; algorithmic information; spreading; complex networks; distributed systems

1. Introduction

The general scope of this paper encompasses complex systems, complex networks, information theory and computability theory. In particular, we study the general problem of emergence of complexity or information when complex systems are networked compared with when they are isolated. This issue has a pervasive importance in the literature about complex systems, with applications to investigating systemic properties of biological, economical or social systems. As discussed in [1], it may be a subject connected to questions ranging from the problem of symbiosis [2], cooperation [3] and integration [4] to biological [5], economic [6] and social [7] networks. To this end, we tackle this general problem with a theoretical approach. We present a

https://doi.org/10.25088/ComplexSystems.27.4.369

mathematical study on the emergence of irreducible information in networked computable systems that follow an information-sharing (or communication) protocol based on contagion or infection models, as described in [8–10]. As supported by these references, such models of spreading using the approach from complex networks have been shown to be relevant in order to study epidemic and disease spreading, computer virus infections or the spreading of polluting agents. Consequently, it has helped, for instance, with immunization strategies, epidemiology or pollution control [8–10].

However, instead of focusing on the pathological properties of these complex networks' contagion dynamics, we show that this dynamics may instead trigger an unlimited potential of optimization through diffusion. That is, diffusing the best solution (or the largest integer when one uses the busy beaver game [1] as a toy model) through the network may trigger an unlimited increase of expected emergent algorithmic information of the nodes as the randomly generated population of computable systems (i.e., nodes) grows. Thus, this paper's objective is to present theorems and corollaries in order to mathematically investigate under which conditions this phenomenon is expected to happen.

For this purpose, we use the theoretical framework for networked computable systems developed in [1, 11, 12] and a susceptible-infected-susceptible (SIS) [13] epidemiological model, which was also studied in [8–10]. The definitions and main lemmas and theorems are based on [1, 11] and, from preliminary results in [12] only for Barabási–Albert static networks, we extend our previous results to general dynamic topologies with a stationary prevalence. The phenomenon of triggering an unlimited increase of the expected local emergent algorithmic complexity (or information) of a node as the population size grows indefinitely was previously studied in [1], however, under different topological conditions and under different communication protocols. We have called this phenomenon expected (local) emergent open-endedness.

From an information-theoretic perspective, this paper is related to emergence in complex systems as in [14–16]. In addition, it is related to an information-theoretic study (statistical and/or algorithmic) on complex networks or graphs [17–20]. Besides complex systems and complex networks, our work is also related to and inspired by fundamental concepts in distributed computing, multi-agent systems and evolutionary game theory. See [1] for more discussion.

In this paper, we show that there is a lower bound for the stationary prevalence that triggers expected (local) emergent open-endedness. As a direct consequence, we show that static networks with a power-law degree distribution following the Barabási–Albert model display expected (local) emergent open-endedness.

First, we base our model and definitions on a mathematical representation for randomly generated computable systems (i.e., systems that can be fully simulated in a Turing machine) that are networked in a time-varying topology (i.e., a dynamic network). In our model, nodes are randomly generated Turing machines that can send and receive information (i.e., partial outputs) as each node runs its computations until returning a final output. We have defined this networked population of randomly generated Turing machines (and a more general mathematical model for networked computable systems) in [1, 11]. We have called it *algorithmic networks*.

Second, we describe and define an algorithmic network that plays a modified version of the busy beaver imitation game (BBIG), in which each node always imitates the fittest neighbor only. We present here a variation on the information-sharing (or communication) protocol that is different from the model in [1]. The major difference with respect to this previous work comes from allowing nodes to become "cured" with rate δ and get "infected" with rate ν. That is, although still playing a BBIG, now susceptible nodes follow a rule of imitating the neighbor that had output the largest integer (which corresponds to the fittest individual outcome in the population). However, they follow this rule with probability ν. In addition, infected nodes come back—become cured—to an "uninfected" state with probability δ. Thus, the key idea of this model is that the infection scheme of the best output returned by a randomly generated node is ruled by the SIS epidemic model, in which there is a constant probability ν of being infected by a previously fitter neighbor and a constant probability δ of returning to a non-fittest state.

Here we also assume, as in [8–10], that the prevalence of infected nodes (i.e., the average density of infected nodes) becomes stationary after sufficient time. In particular, our results hold if this amount of time is upper bounded by a computable function. Therefore, the effective spreading rate $\lambda = \nu / \delta$ defined in [8–10] assumes a direct interpretation of the rate in which the *imitation-of-the-fittest protocol* [1] was applied on a node—and this is the reason why we are using the words "infection" and "cure" between quotation marks.

Then we show that, for big enough values of m compared to λ, if the time for achieving a stationary prevalence of infected nodes $\rho \sim \exp\left(-1 / m\lambda\right)$ is upper bounded by a value given by a computable function, then the expected emergent algorithmic complexity/information of a node (i.e., the expected local emergent algorithmic complexity/information) goes to infinity as the network/population size N goes to infinity. In other words, the average local irreducible information that emerges when nodes are networked compared with when they are isolated is expected to always increase for large enough

populations of randomly generated Turing machines. As a direct consequence of [8–10], our results also imply that the same emergent phenomenon occurs if the network is static and has a scale-free degree distribution in the form of a power law $P(k) \sim 2m^2 / k^3$. This topology and construction of the networks are defined by a random process connecting new nodes under a probability distribution given by a preferential attachment as in [21]. That is, new nodes are more likely to have connections to higher-degree previous nodes. Thus, it is a corollary of our main result that such scale-free static algorithmic networks also display expected local emergent open-endedness.

Our proofs stem from the main idea of combining an estimation of a lower bound for the average algorithmic complexity/information of a networked node and an estimation of an upper bound for the expected algorithmic complexity/information of an isolated node. Additionally, as in [1], the estimation of the latter still comes from the law of large numbers, Gibb's inequality and algorithmic information theory applied on the randomly generated population. However, now the estimation of the former comes from the SIS model with a stationary prevalence (i.e., a stationary average density of infected nodes). The central idea is that the prevalence $\rho \sim \exp(-1 / m\lambda)$, as in [8–10], becomes equal to the average diffusion density τ_{E} in [1].

2. Background

In this section, we provide preliminary definitions and concepts along with previous results on which this paper is based. In particular, these are based on [1]. A more extensive description of these models can also be found in [22].

2.1 Algorithmic Networks

We remember here the general definition of *algorithmic networks* $\mathfrak{N} = (G, \mathfrak{P}, b)$ in [1]. It is a triple $(G, \mathfrak{P}, b)$ defined upon a population of theoretical machines $\mathfrak{P}$, a generalized graph $G = (\mathcal{A}, \mathcal{E})$ and a function b that makes aspects of G correspond to properties of $\mathfrak{P}$, so that a node in $\mathrm{V}(G)$ is mapped one-to-one to an element of $\mathfrak{P}$.

First, (generalized) graphs G, which are multi-aspect graphs (MAGs), are generalized representations for different types of graphs [23, 24]. In particular, an MAG represents dyadic (or two-place) relations between arbitrary n-tuples. Since we aim at a wider range of different network configurations, MAGs allow you to mathematically represent abstract aspects that may appear in complex high-order networks [25]. For example, these may be dynamic (or time-varying)

networks, multicolored nodes (or edges) or multilayer networks, among others. Moreover, this representation facilitates network analysis by showing that their aspects can be isomorphically mapped into a classical directed graph [24]. Thus, the MAG abstraction has proved to be crucial in [1] to establish connections between the characteristics of the network and the properties of the population composed of theoretical machines. Formally:

Definition 1. Let $G = (\mathcal{A}, \mathcal{E})$ be an MAG, where $\mathcal{E}$ is the set of existing composite edges of the MAG and $\mathcal{A}$ is a class of sets, each of which is an aspect. Each aspect $\sigma \in \mathcal{A}$ is a finite set, and the number of aspects $p = |\mathcal{A}|$ is called the order of G. By an immediate convention, we call an MAG with only one aspect a first-order MAG, an MAG with two aspects a second-order MAG and so on. Each composite edge (or arrow) $e \in \mathcal{E}$ may be denoted by an ordered $2p$-tuple $(a_1, \ldots, a_p, b_1, \ldots, b_p)$, where a_i, b_i are elements of the i^{th} aspect with $1 \le i \le p = |\mathcal{A}|$.

$\mathcal{A}(G)$ denotes the class of aspects of G and $\mathcal{E}(G)$ denotes the *composite edge set* of G. We denote the i^{th} aspect of G as $\mathcal{A}(G)[i]$. So, $|\mathcal{A}(G)[i]|$ denotes the number of elements in $\mathcal{A}(G)[i]$. In order to match the classical graph case, we adopt the convention of calling the elements of the first aspect of an MAG *vertices*. Therefore, we denote the set $\mathcal{A}(G)[1]$ of elements of the first aspect of an MAG G as $\mathrm{V}(G)$. Thus, a vertex should not be confused with a composite vertex.

Note that the terms *vertex* and *node* may be employed interchangeably in this paper. However, we choose to use the term *node* preferentially in the context of networks, where nodes may realize operations or computations or would have some kind of agency, like in real networks or algorithmic networks. Thus, we choose to use the term *vertex* preferentially in the mathematical context of graph theory.

Second, we define a *population* $\mathfrak{P}$ as an ordered sequence (in which repetitions are allowed) $(o_1, \ldots, o_i, \ldots, o_{|\mathfrak{P}|})$, where X is the support set of the population and f_o is a labeling surjective function

$$\begin{aligned} f_o : \ \mathfrak{P} = (o_1, \ldots, o_i, \ldots, o_{|\mathfrak{P}|}) &\to X \subseteq L \\ o_i &\mapsto f_o(o_i) = w \end{aligned},$$

where L is the language on which the chosen theoretical machine U is running. Each member of this population may receive inputs and return outputs through communication channels. A *communication channel* between a pair of elements from $\mathfrak{P}$ is defined in $\mathcal{E}$ by an edge (whether directed or undirected) linking this pair of nodes/programs.

Third, we define function b as follows:

Definition 2. Let

$$b: \; Y \subseteq \mathcal{A}(G) \to X \subseteq Pr(\mathfrak{P})$$
$$\overline{\mathbf{a}} \quad \mapsto b(\overline{\mathbf{a}}) = \overline{\mathbf{p_r}}$$

be a function that maps a subspace of aspects Y in $\mathcal{A}$ into a subspace of properties X in the set of properties $Pr(\mathfrak{P})$ of the respective population. In addition, there is a bijective function $f_{V\mathfrak{P}}$ such that, for every $(v, \overline{\mathbf{x}}) \in Y \subseteq \mathcal{A}(G)$ with $b(v, \overline{\mathbf{x}}) = (o_i, b_{|Y|-1}(\overline{\mathbf{x}})) \in X$,

$$f_{V\mathfrak{P}}: \; V(G) \to \mathfrak{P} = \{o_i \,|\, f_o(o_i) = w \in L\}$$
$$v \quad \mapsto f_{V\mathfrak{P}}(v) = o_i$$

where v is a vertex (or node) and o_i is an element of sequence $\mathfrak{P}$.

2.2 Busy Beaver Imitation Game

In [1], we have narrowed our theoretical approach by defining a class of algorithmic networks $\mathfrak{N}_{BB}(N, f, t, \tau, j)$—also denoted by a triplet as $(G_t, \mathfrak{P}_{BB}(N), b_j)$—in which their populations $\mathfrak{P}_{BB}(N)$ and graphs $G_t \in \mathbb{G}(f, t, \tau)$ have determined properties.

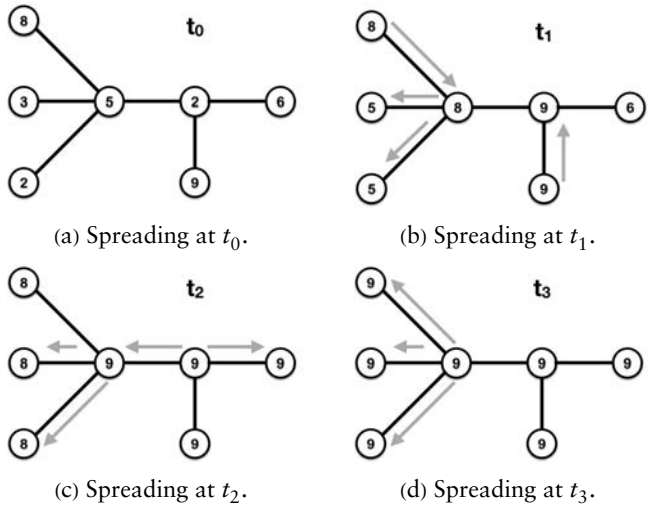
As defined in Section 2.1, each element of the population corresponds one-to-one to a node/vertex in G_t, and each time instant in G_t is mapped to a cycle (or communication round). These mappings are defined by the function b_j.

The population $\mathfrak{P}_{BB}(N)$ is composed of randomly generated Turing machines (or randomly generated self-delimiting programs), which are represented in a self-delimiting universal programming language $\mathbf{L_U}$. This population is synchronous with respect to halting cycles. That is, at the end of a cycle (or communication round, as in distributed computing), every node returns its outputs at the same time. In addition, nodes that do not halt in a cycle always return as final output the lowest fitness, that is, the integer value 0. Here, a straightforward interpretation is that nodes that eventually do not halt in a cycle are "killed," so that their final output has the "worst" fitness. Thus, these nodes are programs that ultimately run on an oracle Turing machine (or a hypercomputable system) $\mathbf{U}'$—this requirement is also analogous to the one in [14, 26, 27], which deals with a sole program at a time and not with a population of them. However, the oracle is only necessary to deal with the non-halting computations. That is, $\mathbf{U}'$ behaves like a universal Turing machine $\mathbf{U}$, except that it returns zero whenever a nonhalting computation occurs.

In addition, the networked population $\mathfrak{P}_{BB}(N)$ follows an imitation-of-the-fittest protocol (IFP), diffusing the information of the fittest

randomly generated node (i.e., the randomly generated node that partially outputs the largest integer in cycle 1). As in [11, 14, 26, 28], note that we still use the busy beaver function as our fitness function. Therefore, the largest integer directly represents the fittest final output of a node. Every node in $\mathfrak{N}_{\mathrm{BB}}(N, f, t, \tau, j)$ obeys the IFP, in which after the first cycle (i.e., after the first round of partial outputs) every node only imitates the neighbor that has partially output the largest integer, repeating this value as its own partial output in the next cycle. Thus, the main idea defining the IFP is a procedure in which each node o_i compares its neighbors' partial output (i.e., the integer they have calculated in the respective cycle) and runs the program of the neighbor that has output the largest integer if and only if this integer is larger than the one that the very node o_i has output.

Since $\mathfrak{N}_{\mathrm{BB}}(N, f, t, \tau, j)$ is playing the busy beaver game [1] on a network while limited to simple imitation performed by a randomly generated population of programs, we say it is playing a BBIG. A (network) busy beaver game is a game in which each player is trying to calculate the largest integer—as established as our measure of fitness or payoff (see [1, 14, 27] for more discussion)—it can by using the information shared by its neighbors. Thus, the BBIG is a special case of the busy beaver game. In Figure 1, we give an illustrative simple example of the IFP being applied to a static network with seven nodes in four time instants. Note that static networks are a special case of a TVG. See also Section 3.



(a) Spreading at t_0. (b) Spreading at t_1.

(c) Spreading at t_2. (d) Spreading at t_3.

Figure 1. The spreading of the largest integer.

Graphs $G_t = (V, \mathcal{E}, T)$ are time-varying graphs (TVGs) as defined in [24, 29, 30]. These are a special case of second-order MAGs that have only one additional aspect relative to variation over time with respect to the set of nodes/vertices. Therefore, $V(G_t)$ is the set of nodes, $T(G_t)$ is the set of time instants, and $\mathcal{E} \subseteq V(G_t) \times T(G_t) \times V(G_t) \times T(G_t)$ is the set of edges. Formally, we define a family of graphs G_t that share a common property:

Definition 3. Let

$$\mathbb{G}(f, t, \tau)\, j = \left\{ G_t \;\middle|\; \begin{array}{l} i = |V(G_t)| \text{ and this size is unique in } \mathbb{G}(f, t, \tau) \\ f(i, t, \tau) \text{ is well defined} \end{array} \right\}$$

where

$$\begin{array}{rccl} f\colon & \mathbb{N}^* \times X \subseteq T(G_t) \times Y \subseteq &]0, 1] & \to \mathbb{N} \\ & (x, t, \tau) & & \mapsto y \end{array}$$

be a family of unique-sized time-varying graphs that share function $f(i, t, \tau)$, where i is the number of vertices, as a common property.

In summary, $\mathfrak{N}_{BB}(N, f, t, \tau, j)$ is a synchronous algorithmic network populated by N randomly generated nodes such that, after the first cycle (or arbitrary c_0 cycles), it starts a diffusion process of the biggest partial output (given at the end of the first cycle) determined by network G_t: at the first time instant, each node may receive a network input w, which is given to every node in the network, and runs separately (i.e., not networked), returning its respective first partial output; then, the plain diffusion of large integers starts as determined by the IFP through the respective dynamical network G_t. At the last time instant, contagion stops and one cycle (or more) is spent in order to make each node return a final output. Formally:

Definition 4. Let

$$\mathfrak{N}_{BB}(N, f, t, \tau, j) = (G_t, \mathfrak{P}_{BB}(N), b_j)$$

be an algorithmic network where f is an arbitrary well-defined function such that

$$\begin{array}{rccl} f\colon & \mathbb{N}^* \times X \subseteq T(G_t) \times Y \subseteq &]0, 1] & \to \mathbb{N} \\ & (x, t, \tau) & & \mapsto y \end{array}$$

and $G_t \in \mathbb{G}(f, t, \tau)$, $|V(G_t)| = N$, $|T(G_t)| > 0$, and there are arbitrarily chosen $c_0, n \in \mathbb{N}$ where $c_0 + |T(G_t)| + 1 \leq n \in \mathbb{N}$ such that b_j is an

injective function, where

$$b_j : \mathrm{V}(G_t) \times \mathrm{T}(G_t) \to \mathfrak{P}_{\mathrm{BB}}(N) \times \mathbb{N} \Big|_1^n$$
$$(v, t_{c-1}) \mapsto b_j(v, t_{c-1}) = (o_i, c_0 + c).$$

Since c_0 and n are arbitrarily chosen, we can choose to make them as small as possible in order to minimize the number of cycles, for example, $c_0 = 0$ and $n = \left|\mathrm{T}(G_t)\right| + 1$.

2.3 Evolutionary Open-Endedness versus Emergent Open-Endedness

Although investigating evolutionary open-endedness (EOE) is not in the scope of the present paper, we mention in this section some general concepts from the literature in order to differentiate and emphasize its relation to *emergent open-endedness*, as introduced in [1]. EOE is also known as open-ended evolution (OEE), and although the term "open-ended evolution" is more frequently used in our references, we choose to employ the term "evolutionary open-endedness" with the purpose of drawing a comparison with emergent open-endedness.

In a general sense, as pointed out in [14, 31], the concept of EOE is commonly understood in evolutionary computation, evolutionary biology or complex systems as the inherent potential of an evolutionary process to trigger an endless increase of distinct systemic behavior capabilities. For example, within the context of dynamical systems, EOE is shown to be strictly related to innovation and unbounded evolution in state trajectories or rule trajectories [31]. However, since we are studying networked computable systems in this paper, we choose to follow an algorithmic and universal approach to EOE in which (absolute or relative) undecidability and irreducibility play a central role [14, 27, 32–34]. In particular, it is shown in [14] and experimentally supported in [34] that the model introduced in [32, 33] satisfies the requirements for *strong* EOE. In [27], it is shown that this approach can be relativized, so that it also holds for subrecursive classes of computable systems. Thus, in the specific case of evolutionary computation and strong EOE in general computable systems, *open-endedness* is strictly related to an endless increase of complexity or irreducible information.

Following this algorithmic approach to EOE, we have found in [1]—and we also show in this paper—that open-endedness may also emerge as a phenomenon that is related to, but different from EOE: instead of achieving an unbounded quantity of algorithmic complexity over time (e.g., after successive mutations), an unbounded quantity of emergent algorithmic complexity is achieved as the

population/network size increases indefinitely. And since it is a property that emerges depending on the number of parts of a system—only when these nodes are interacting somehow (e.g., exchanging information)—this additional irreducible information becomes by definition an emergent systemic property [15, 16, 35].

We follow a consensual abstract notion of *emergence* [15, 16, 35–37] as a systemic feature or property that appears only when the system is analyzed (theoretically or empirically) as a "whole." Thus, the algorithmic complexity (i.e., an irreducible number of bits of information) of a node/program's final output when networked minus the algorithmic complexity of a node/program's final output when isolated formally defines an irreducible quantity of information that *emerges* with respect to a node/program that belongs to an algorithmic network. We call it the *emergent algorithmic complexity* (EAC) of a node/program. The reader may also find more discussion on emergence and open-endedness in [1].

Formally, we have defined *average emergent open-endedness* in the context of general algorithmic networks as follows:

Definition 5. We say an algorithmic network $\mathfrak{N}$ with a population of N nodes has the property of average (local) emergent open-endedness (AEOE) for a given network input w in c cycles if and only if

$$\lim_{N\to\infty} \mathbf{E}_{\mathfrak{N}}\left(\overset{\text{net}}{\underset{\text{iso}}{\Delta}} A(o_i, c)\right) = \infty.$$

And in the case of an algorithmic network $\mathfrak{N} = (G, \mathfrak{P}, b)$ with randomly generated nodes, we call this property *expected (local) emergent open-endedness*. We have that

$$\mathbf{E}_{\mathfrak{N}}\left(\overset{\text{net}}{\underset{\text{iso}}{\Delta}} A(o_i, c)\right) = \sum_b \frac{\frac{\sum_{o_i\in\mathfrak{P}} \overset{\text{net}(b)}{\underset{\text{iso}}{\Delta}} A(o_i,c)}{N}}{|\{b\}|}$$

denotes the average emergent algorithmic complexity of a node/program (AEAC) in an algorithmic network $\mathfrak{N} = (G, \mathfrak{P}, b)$ with network input w. In addition:

Definition 6. The emergent algorithmic complexity (EAC) of a node/program o_i in c cycles is given in an algorithmic network that always produces partial and final outputs by

$$\overset{\text{net}(b)}{\underset{\text{iso}}{\Delta}} A(o_i, c) = A\left(\mathbf{U}\left(p^b_{\text{net}}(o_i, c)\right)\right) - A(\mathbf{U}(p_{\text{iso}}(o_i, c))$$

where:

1. $f_o(o_i) \in L$.

2. p^b_{net} is the program that computes cycle per cycle the partial outputs of o_i when networked assuming the position v, where $b(v, \overline{\mathbf{x}}) = (o_i, b(\overline{\mathbf{x}}))$, in the graph G in the specified number of cycles c with network input w. Thus, $p^b_{net}(o_i, c)$ represents the program that returns the final output of o_i when networked assuming the position v, where $b(v, \overline{\mathbf{x}}) = (o_i, b(\overline{\mathbf{x}}))$, in the graph G in the specified number of cycles c with network input w.
3. p_{iso} is the program that computes cycle per cycle the partial outputs of o_i when isolated in the specified number of cycles c with network input w. Thus, $p_{\text{iso}}(o_i, c)$ represents the program that returns the final output of o_i when isolated in the specified number of cycles c with network input w.

The (prefix) algorithmic complexity (Kolmogorov complexity, program-size complexity or Solomonoff–Komogorov–Chaitin complexity) of a string $w \in \mathbf{L_U}$, denoted here by $A(w)$, is the length of the shortest program $w^* \in \mathbf{L_U}$ such that $\mathbf{U}(w^*) = w$. $\mathbf{L_U}$ is an arbitrarily chosen binary self-delimiting universal programming language for a universal Turing machine $\mathbf{U}$. See [1] for more discussion on this notation.

2.4 Background Results

In [1], we presented our main theorem proving that there is a lower bound for the expected emergent algorithmic complexity in algorithmic networks $\mathfrak{N}_{\text{BB}}$ such that it depends on how much larger the average diffusion density (in a given time interval) $\tau_{\mathbf{E}(\max)}(N, f, t, \tau)\big|_t^{t'}$ is, compared to the cycle-bounded conditional halting probability $\Omega(w, c(x))$. Formally:

Theorem 1. Let $w \in \mathbf{L_U}$ be a network input. Let $0 < N \in \mathbb{N}$. Let $\mathfrak{N}_{\text{BB}}(N, f, t, \tau, j) = \left(G_{\mathfrak{t}}, \mathfrak{P}_{\text{BB}}(N), b_j\right)$ be well defined. Let $t_0 \le t \le t' \le t_{|\mathrm{T}(G_{\mathfrak{t}})|-1}$. Let

$$c: \mathbb{N} \to \mathfrak{C}_{\text{BB}}$$
$$x \mapsto c(x) = y$$

be a total computable function where $c(x) \ge c_0 + t' + 1$. Then, we will have that:

$$\lim_{N\to\infty} \mathbf{E}_{\mathfrak{N}_{\text{BB}}(N,f,t,\tau)}\left(\overset{\text{net}}{\underset{\text{iso}}{\Delta}} A(o_i, c(x))\right) \ge$$
$$\lim_{N\to\infty} \left(\tau_{\mathbf{E}(\max)}(N, f, t, \tau)\big|_t^{t'} - \Omega(w, c(x))\right) \lg(N) -$$
$$\Omega(w, c(x)) \lg(x) - 2\Omega(w, c(x)) \lg\big(\lg(x)\big) - A(w) - \mathrm{C}_5,$$

where $\lg(x)$ denotes the binary logarithm $\log_2(x)$.

https://doi.org/10.25088/ComplexSystems.27.4.369

This lower bound also depends on the parameter for which one is calculating the number of node cycles. In fact, we have proved that our results hold even in the case of spending a computably larger number of node cycles compared to x. Furthermore, we have proved that there are asymptotic conditions such that they ensure that there is a central time $t_{\mathrm{cen}_1}(c)$ to trigger expected emergent open-endedness. Formally:

Theorem 2. Let $w \in \mathbf{L}_{\mathrm{U}}$ be a network input. Let $0 < N \in \mathbb{N}$. If there is $0 \leq z_0 \leq \left|\mathrm{T}(G_{\mathrm{t}})\right| - 1$ and $\epsilon, \epsilon_2 > 0$ such that

$$z_0 + f\left(N, t_{z_0}, \tau\right) + 2 = \mathbf{O}\left(\frac{N^C}{\lg(N)}\right)$$

where

$$0 < C = \left(\tau_{\mathrm{E(max)}}\left(N, f, t_{z_0}, \tau\right)\Big|_{t_{z_0}}^{t_{z_0+f\left(N, t_{z_0}, \tau\right)}} - \Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}, \tau\right) + 2\right) - \epsilon\right) \Big/ \left(\Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}, \tau\right) + 2\right)\right) \leq \frac{1}{\epsilon_2},$$

then, for every nondecreasing total computable function

$$c\colon \begin{array}{l} \mathbb{N} \to \mathrm{C}_{\mathrm{BB}} \\ \quad x \mapsto c(x) = y \end{array},$$

where $t_{z_0}, t_{z_0+f\left(N, t_{z_0}, \tau\right)} \in \mathrm{T}\left(G_{\mathrm{t}}\right)$ and

$$c\left(z_0 + f\left(N, t_{z_0}, \tau\right) + 2\right) \geq c_0 + z_0 + f\left(N, t_{z_0}, \tau\right) + 2,$$

such that

$$\mathfrak{N}_{\mathrm{BB}}\left(N, f, t_{z_0}, \tau, j\right) = \left(G_{\mathrm{t}}, \mathrm{P}_{\mathrm{BB}}(N), b_j\right)$$

is well defined, we will have that there is $t_{\mathrm{cen}_1}(c)$ such that

$$t_{\mathrm{cen}_1}(c) \leq t_{z_0}.$$

In particular, we will prove a modified version of Theorem 2 in Section 4 for our new model of information-sharing protocol. See Section 3.

Our proofs follow mainly from information theory, computability theory and graph theory. Therefore, we have shown that there are topological conditions (e.g., the small-diameter phenomenon) that trigger a phase transition in which eventually the algorithmic network $\mathfrak{N}_{\mathrm{BB}}$ begins to produce an unlimited number of bits of average local emergent algorithmic complexity/information. These conditions come

from a positive tradeoff between the average diffusion density and the number of cycles (i.e., communication rounds). In particular, the diffusion power of a dynamic (or static) network has proved to be paramount to the purpose of optimizing the average fitness/payoff of an algorithmic network that plays the BBIG in a randomly generated population of Turing machines.

3. Model

In this section, we present the model of algorithmic networks on which we prove lemmas and theorems. The main idea that defines these algorithmic networks $\mathfrak{N}'_{\mathrm{BB}}$ is to formalize an SIS contagion scheme. This is a modification of the algorithmic networks $\mathfrak{N}_{\mathrm{BB}}$ that only follow a plain IFP with a plain spreading of the largest integer, as described in Section 2.2. Thus, in this section, we focus on a description of the new model. For extended formal definitions and extensive discussions, see [22].

Now we will define a class of algorithmic networks $\mathfrak{N}'_{\mathrm{BB}}(N, f, t, j)$—which can also be denoted as the triplet $(G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j)$—in which their populations $\mathfrak{P}'_{\mathrm{BB}}(N)$ and graphs $G_{\mathrm{t}} \in \mathbb{G}_{\mathrm{SIS}}(f, t)$ have determined properties. The terms in parentheses determine the full characterization of the algorithmic network. N is the network/population size, that is, the number of nodes, and j is the index of the arbitrarily chosen function b_j. Terms f and t are intrinsically defined by the family of graphs $\mathbb{G}_{\mathrm{SIS}}(f, t)$, as we will explain later. Each element of the population corresponds one-to-one to a node/vertex in G_{t}, and each time instant in G_{t} is mapped one-to-one to a cyle of the population. These mappings are also defined by the function b_j.

The population $\mathfrak{P}'_{\mathrm{BB}}(N)$ is composed of randomly generated Turing machines (or randomly generated self-delimiting programs) that are represented in a self-delimiting universal programming language $\mathbf{L}_{\mathbf{U}}$. The population is also synchronous with respect to halting cycles; that is, at the end of a cycle (or communication round, as in distributed computing), every node returns its partial and final outputs at the same time. Nodes that do not halt in a cycle always return as final output the lowest fitness/payoff, that is, the integer value 0. Here, a straightforward interpretation is that nodes that eventually do not halt in a cycle are "killed," so that their final output has the "worst" fitness/payoff.

Thus, as presented in [1], these nodes are programs that ultimately run on an oracle Turing machine (or hypercomputer) $\mathbf{U}'$—this requirement is also analogous to the one presented in [26, 28, 32], which

https://doi.org/10.25088/ComplexSystems.27.4.369

deals with a sole program at a time and not with a population of them. The difference in the present paper with respect to [1] is that now the oracle Turing machine also needs access to a randomly generated number in order to deal with the probabilities ν and δ in the SIS contagion scheme explained in the following. See also [1, 32, 33] for a complete evolutionary formalization of this property. Note that now there is a population of software, while in [32, 33] there is only one single organism at a time.

In addition, unlike the networked population $\mathbb{P}_{\mathrm{BB}}(N)$ described in Section 2.2, the networked population $\mathbb{P}'_{\mathrm{BB}}(N)$ follows an IFP by an SIS contagion scheme (IFPSIS) on the fittest randomly generated node (i.e., the randomly generated node that partially outputs the largest integer in cycle 1). As in [11, 14, 26, 28], note that we still use the busy beaver function as our fitness function. Therefore, the largest integer directly represents the fittest final output of a node. Thus, every node still obeys the IFP as in [1], in which after the first cycle (i.e., after the first round of partial outputs), every node only imitates the neighbor that has partially output the largest integer, repeating this value as its own partial output in the next cycle. However, the difference now is that, if a node has not been infected by the fittest randomly generated node and one of its neighbors sends the largest integer, then the node obeys the IFP with probability ν. Otherwise, the node just continues to be susceptible with probability $1-\nu$. Another difference is that, if a node got infected by the largest integer, then it may be cured, returning 0 as partial output, with probability δ. Otherwise, it remains infected with probability $1-\delta$.

In Figure 2, we give an illustrative simplified example of the IFP under an SIS contagion model. This is a static network with six time instants and seven nodes. Note that the initial stage Figure 2(a) at time instant t_0 is exactly the same as in Figure 1. At the first time interval Figure 2(b), almost the same spreading occurs as in Figure 1, but only from the fittest node and under probability ν. After time instant t_1, the infection dynamics occurs totally differently from that described in Figure 1. It is now a contagion of the fittest under an SIS model. In particular, taking into account only these six time instants, one may see that the prevalence becomes stationary at time instant t_4.

Graphs $G_{\mathrm{t}} = (\mathrm{V}, \mathcal{E}, \mathrm{T})$ are TVGs as defined in [1, 24, 29, 30]. These are a special case of MAGs that have only one additional aspect relative to variation over time with respect to the set of nodes/vertices. Therefore, $\mathrm{V}(G_{\mathrm{t}})$ is the set of nodes, $\mathrm{T}(G_{\mathrm{t}})$ is the set of time instants and

$$\mathcal{E} \subseteq \mathrm{V}(G_{\mathrm{t}}) \times \mathrm{T}(G_{\mathrm{t}}) \times \mathrm{V}(G_{\mathrm{t}}) \times \mathrm{T}(G_{\mathrm{t}})$$

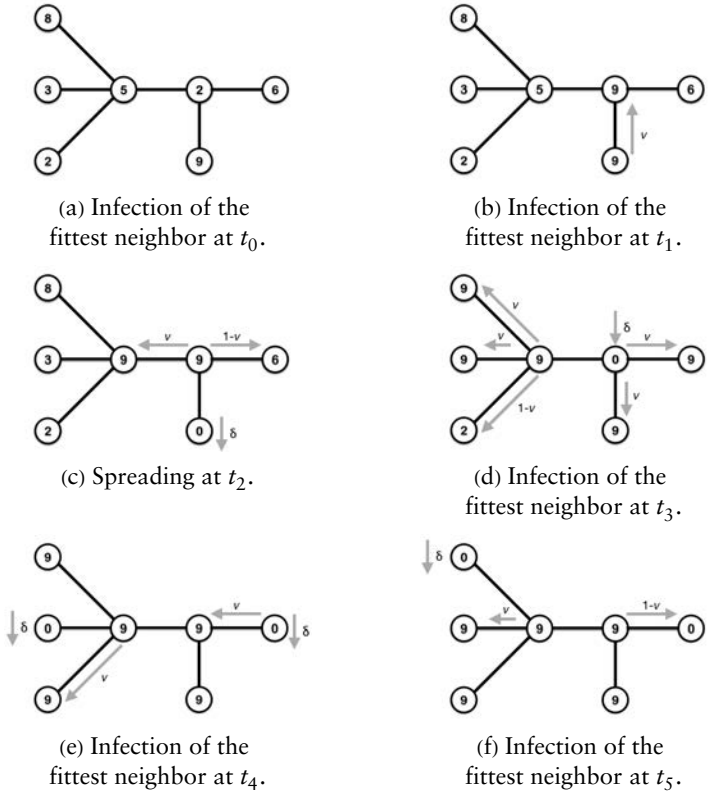


Figure 2. Example of the IFP under an SIS contagion model.

is the set of edges regarding G_t. We assume that an undirected graph (or MAG) is a special case of a directed graph (or MAG) in which each edge represents two opposing arrows. In addition, a static network G_s is also a special case of MAGs, which is obtained from collapsing all the aspects in $\mathcal{A}$ into just one aspect (i.e., into the set of vertices/nodes V) where the set of edges of this MAG is invariant under any relation other than the set of vertices/nodes—see also subdetermination in [24]. Thus, a static network G_s is a traditional (directed or undirected) graph $G = (V, E)$ with all relations (e.g., with respect to time instants or layers) depending only on its set of edges E. However, for present purposes, a TVG is sufficient to deal with the SIS model and hence, there is only one aspect we are collapsing. Therefore, we define a *static network* $G_s = (V, \mathcal{E}, T)$ as a TVG in which, for every fixed value of $t_i, t_j, t_k, t_h \in T$,

$$\{(v_i, v_j) \,|\, (v_i, t_i, v_j, t_j) \in \mathcal{E}\} = \{(v_i, v_j) \,|\, (v_i, t_k, v_j, t_h) \in \mathcal{E}\}.$$

Inspired by the networks in [8–10], let $\mathbb{G}_{\mathrm{SIS}}(f, t)$ be a family of TVGs in which every $G_{\mathrm{t}} \in \mathbb{G}_{\mathrm{SIS}}(f, t)$ achieves stationary prevalence ρ in a number of time intervals Δ_t^* (after an arbitrary time instant $t \in \mathrm{T}(G_{\mathrm{t}})$ from which contagion may have been started in the first place) following the SIS scheme. Formally:

Definition 7. Let

$$\mathbb{G}_{\mathrm{SIS}}(f, t) = \{G_{\mathrm{t}} \mid i = |\mathrm{V}(G_{\mathrm{t}})| \text{ and this size is unique in } \mathbb{G}_{\mathrm{SIS}}(f, t)\}$$

where

$$f : \begin{array}{c} \mathbb{N}^* \times X \subseteq \mathrm{T}(G_{\mathrm{t}}) \to \mathbb{N} \\ (x, t) \qquad\quad \mapsto y \end{array}$$

be a family of unique-sized TVGs that depends on the choice of function f, the time instant t and on the fact that every $G_{\mathrm{t}} \in \mathbb{G}_{\mathrm{SIS}}$ achieves stationary prevalence ρ in a number of time intervals Δ_t^* (after an arbitrary time instant $t \in \mathrm{T}(G_{\mathrm{t}})$) following a susceptible-infected-susceptible (SIS) contagion scheme.

Thus, $\mathbb{G}_{\mathrm{SIS}}(f, t)$ defines a family of dynamic networks [29, 30, 38] that follows the SIS model. Since we have defined static networks as a special case of dynamic networks, family $\mathbb{G}_{\mathrm{SIS}}(f, t)$ can be seen as a generalization of the model presented in [8–10] (see Definition 8) in order to encompass a broader class of dynamic networks. Since function f and time instant t are not specified in the condition of the set $\mathbb{G}_{\mathrm{SIS}}(f, t)$, then this family is independent of the choice of (f, t). However, the reader will see that this pair (f, t) is crucial for extending the results in [1] in order to build the proof of Theorem 3 and Corollary 1.

In addition, we define a family $\mathbb{G}_{\mathrm{BA}}(f, t)$ of TVGs in $\mathbb{G}_{\mathrm{SIS}}$ that are static networks following a classical Barabási–Albert model [21, 39]. They have a scale-free distribution of connectivities as a consequence of an application of preferential attachment at the addition of each new node, which results in a degree distribution in the form of a power law

$$P(k) \sim \frac{2m^2}{k^3}$$

as the number of nodes goes to infinity. The finite number of nodes of each graph in this family may vary from 1 to ∞ as each new node is added with m edges linked to previous nodes i under probability

distribution

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}.$$

Note that, as shown in [8–10], these networks in $\mathbb{G}_{\mathrm{BA}}(f, t)$ are expected to display a stationary prevalence

$$\rho \sim \exp\left(-\frac{1}{m\lambda}\right)$$

for a large enough network size and for a small enough spreading rate λ. If these two conditions are met, then $\mathbb{G}_{\mathrm{BA}}(f, t) \subseteq \mathbb{G}_{\mathrm{SIS}}(f, t)$. Therefore, family $\mathbb{G}_{\mathrm{BA}}(f, t)$ is defined to directly correspond to the networks presented in [8–10]. Formally:

Definition 8. We define a family of static networks analogous to the ones presented in [8–10] as

$$\mathbb{G}_{\mathrm{BA}}(f, t) = \{G_{\mathrm{s}} \mid i = |\mathrm{V}(G_{\mathrm{s}})| \text{ and the size is unique in } \mathbb{G}_{\mathrm{BA}}(f, t)\}$$

where

$$f\colon \begin{array}{ccc} \mathbb{N}^* \times X \subseteq \mathrm{T}(G_{\mathrm{t}}) & \to & \mathbb{N} \\ (x, t) & \mapsto & y \end{array}.$$

Thus, $\mathbb{G}_{\mathrm{BA}}(f, t)$ is a family of unique-sized TVGs that are static networks following a classical Barabási–Albert model [21, 39] such that every $G_{\mathrm{s}} \in \mathbb{G}_{\mathrm{BA}}$ achieves stationary prevalence ρ in a number of time intervals Δ_t^* (after an arbitrary time instant $t \in \mathrm{T}(G_{\mathrm{t}})$) following an SIS contagion scheme.

In summary, $\mathfrak{N}'_{\mathrm{BB}}(N, f, t, j)$ is a synchronous algorithmic network populated by N randomly generated nodes such that, after the first cycle (or arbitrary c_0 cycles), it starts a diffusion process of the biggest partial output (given at the end of the first cycle) determined by network G_{t} that belongs to a family of graphs $\mathbb{G}_{\mathrm{SIS}}(f, t)$—remember that each network in $\mathbb{G}_{\mathrm{SIS}}(f, t)$ follows an SIS contagion scheme. Note that in our model, once the first cycle is started, the population remains fixed. Thus, during the cycles (i.e., when the algorithmic network is running its computations) no new node is created and no node is killed. At the beginning of the first cycle, each node receives a network input w, which is given to every node in the network, and runs separately (i.e., not networked), returning its respective first partial output. At the last time instant, contagion stops and one cycle (or more) is spent in order to make each node return a final output. Formally:

Definition 9. Let

$$\mathfrak{N}'_{\mathrm{BB}}(N, f, t, j) = \left(G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j\right)$$

be an algorithmic network where f is an arbitrary well-defined function such that

$$\begin{aligned} f\colon\ & \mathrm{N}^* \times X \subseteq \mathrm{T}\left(G_{\mathrm{t}}\right) \to \mathbb{N} \\ & (x, t) \qquad \mapsto y \geq x \end{aligned}$$

and $G_{\mathrm{t}} \in \mathbb{G}_{\mathrm{SIS}}(f, t)$, $\left|\mathrm{V}\left(G_{\mathrm{t}}\right)\right| = N$, $\left|\mathrm{T}\left(G_{\mathrm{t}}\right)\right| > 0$ and there are arbitrarily chosen $c_0, n \in \mathbb{N}$ where $c_0 + \left|\mathrm{T}\left(G_{\mathrm{t}}\right)\right| + 1 \leq n \in \mathbb{N}$ such that b_j is an injective function

$$\begin{aligned} b_j\colon\ & \mathrm{V}\left(G_{\mathrm{t}}\right) \times \mathrm{T}\left(G_{\mathrm{t}}\right) \to \mathfrak{P}'_{\mathrm{BB}}(N) \times \mathbb{N}\,|_1^N \\ & (v, t_{c-1}) \quad \mapsto b_j(v, t_{c-1}) = (o_i, c_0 + c) \end{aligned}$$

where population $\mathfrak{P}'_{\mathrm{BB}}(N)$ is an ordered set of labels from a sequence as in Section 2.1. Since c_0 and n are arbitrarily chosen, we can make them as small as possible in order to minimize the number of cycles, for example. That is, $c_0 = 0$ and $n = \left|\mathrm{T}\left(G_{\mathrm{t}}\right)\right| + 1$.

4. Expected Local Emergent Open-Endedness from a Susceptible-Infected-Susceptible Model

In this section, we present the central theorem and its two corollaries with the purpose of showing that $\mathfrak{N}'_{\mathrm{BB}}(N, f, t, j) = \left(G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j\right)$ is an algorithmic network capable of exhibiting expected (local) emergent open-endedness. We show that it occurs under certain topological conditions of the graph G_{t} in which the prevalence (i.e., the average density of infected nodes) becomes stationary within a computably bigger time interval. During these time intervals, the algorithmic network is running under the IFP with an SIS contagion scheme.

Moreover, once these topological properties are met, the concept of central time (denoted as t_{cen_1}) to trigger expected emergent open-endedness within the minimum number of cycles becomes well defined. We define the central time t_{cen_1} in generating *unlimited* expected emergent algorithmic complexity of a node (i.e., expected *local* emergent algorithmic complexity) in a network $\mathfrak{N}'_{\mathrm{BB}}(N, f, t_{z_0}, j)$ during $c\left(t_{\mathrm{cen}_1}(c) + f\left(N, t_{\mathrm{cen}_1}(c)\right) + 1\right)$ cycles, where $c(x)$ is a nondecreasing total computable function and f is an arbitrary function, as the minimum time instant t in which the expected local emergent algorith-

mic complexity goes to infinity as $N \to \infty$ after $c\big(t + f(N, t) + 1\big)$ cycles. Formally:

Definition 10. Let $w \in \mathbf{L}_{\mathrm{U}}$ be a network input. Let $0 < N \in \mathbb{N}$. Let $c(x)$ be a nondecreasing total computable function where

$$c\colon \begin{array}{l} \mathbb{N} \to \mathbb{N}^* \\ x \mapsto c(x) = y \end{array}.$$

Let $\mathfrak{N}'_{\mathrm{BB}}\big(N, f, t_z, j\big) = \big(G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j\big)$, where $0 \le z \le \big|\mathrm{T}\big(G_{\mathrm{t}}\big)\big| - 1$, be well defined, where there is $t_{z_0} \in \mathrm{T}\big(G_{\mathrm{t}}\big)$ such that

$$\lim_{N\to\infty} \mathbf{E}_{\mathfrak{N}'_{\mathrm{BB}}(N,f,t_{z_0})}\Big(\overset{\text{net}}{\underset{\text{iso}}{\Delta}} A\big(o_i, c\big(z_0 + f\big(N, t_{z_0}\big) + 2\big)\big)\Big) = \infty.$$

We define the central time t_{cen_1} in generating *unlimited* expected emergent algorithmic complexity of a node in a network $N'_{\mathrm{BB}}\big(N, f, t_{z_0}, j\big)$ during $c\big(t_{\mathrm{cen}_1}(c) + f\big(N, t_{\mathrm{cen}_1}(c)\big) + 1\big)$ cycles as

$$t_{\mathrm{cen}_1}(c) = \min\Big\{t_i \,\Big|\, \lim_{N\to\infty} \mathbf{E}_{\mathfrak{N}'_{\mathrm{BB}}(N,f,t_i)}\Big(\overset{\text{net}}{\underset{\text{iso}}{\Delta}} A\big(o_i, c\big(i + f(N, t_i) + 2\big)\big)\Big) = \infty\Big\}.$$

Note that the arbitrarily chosen function f may not behave monotonically with t in general.

The expected local emergent algorithmic complexity is defined in [1, 11] (see also Section 2.3) as the number of extra bits of algorithmic complexity (or information) that emerges from a direct comparison of the algorithmic complexity of the final output of a networked node with the algorithmic complexity of the final output of the isolated (same) node. The term "local" here refers to the emergent algorithmic complexity of a node. The investigation of the emergent algorithmic complexity of the population as a whole, as also mentioned in [1], is out of our current scope. We can call this latter a *global* emergent algorithmic complexity. Note that it may behave differently from the local one. Thus, we leave the investigation of expected global emergent open-endedness for future research.

As in [1], the main idea behind the construction of the proof of Theorem 3 comes from combining an estimation of a lower bound for the average algorithmic complexity of a *networked* node and an estimation of an upper bound for the expected algorithmic complexity of an *isolated* node. The estimation of the latter comes from the law of large numbers, Gibb's inequality and algorithmic information theory applied on the randomly generated population $\mathfrak{P}'_{\mathrm{BB}}(N)$, which is analogously the same as $\mathfrak{P}_{\mathrm{BB}}(N)$ in [1] for the isolated case. However, now the estimation of the former comes from the very BBIG dynamics in an SIS contagion scheme. Thus, as presented in

Section 2.3, calculating the former estimation minus the latter gives directly a lower bound for the expected local emergent algorithmic complexity of a node.

In this section, we present short proofs of Theorem 3 and Corollary 1. Most of these proof steps are based on a direct analogy to the proof steps developed in [1], so only in Corollaries 1 and 2 would our new model introduce new conceptual substantial differences in the mathematical formal text. For complete and self-contained definitions, lemmas, theorems and corollaries, see [22].

4.1 Central Time to Trigger Expected Emergent Open-Endedness

Theorem 3. Let $w \in \mathbf{L}_{\mathrm{U}}$ be a network input. Let $0 < N \in \mathbb{N}$. Let f be an arbitrary function where

$$\begin{array}{rccc} f\colon & \mathbb{N}^{*}\times X \subseteq \mathrm{T}\left(G_{\mathrm{t}}\right) & \to & \mathbb{N} \\ & (x,\, t) & \mapsto & y \end{array}.$$

Let

$$\begin{array}{l} c\colon \mathbb{N} \to \mathfrak{C}_{\mathrm{BB}} \\ \quad x \mapsto c(x) = y \end{array},$$

where $\mathfrak{C}_{\mathrm{BB}}$ is the set of cycles of the population $\mathfrak{P}'_{\mathrm{BB}}(N)$, be a total computable nondecreasing function where

$$c\left(z + f(N,\, t_z) + 2\right) \geq c_0 + z + f(N,\, t_z) + 2$$

and

$$c\left(z + f(N,\, t_z) + 2\right) - c_0 - 1 \leq t_{|\mathrm{T}(G_{\mathrm{t}})|-1}.$$

If there are $0 \leq z_0 \leq \left|\mathrm{T}\left(G_{\mathrm{t}}\right)\right| - 1$ and $\epsilon, \epsilon_2 > 0$ such that:

$$z_0 + f\left(N,\, t_{z_0}\right) + 2 = \mathbf{O}\left(\frac{N^{C}}{\lg(N)}\right)$$

where

$$\begin{aligned} 0 \leq C = \Big(\tau_{\mathrm{E}(\rho)}\left(N,\, f,\, t_{z_0}\right)\Big|_{t_{z_0}}^{c\left(z_0 + f\left(N, t_{z_0}\right)+2\right)-c_0-1} \\ -\Omega\left(w,\, c_0 + z_0 + f\left(N,\, t_{z_0}\right) + 2\right) - \epsilon\Big) \Big/ \\ \left(\Omega\left(w,\, c_0 + z_0 + f\left(N,\, t_{z_0}\right) + 2\right)\right) \leq \frac{1}{\epsilon_2} \end{aligned}$$

and $\mathfrak{N}'_{\mathrm{BB}}(N, f, t_{z_0}, j) = (G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j)$ is well defined, then there is $t_{\mathrm{cen}_1}(c)$ such that

$$t_{\mathrm{cen}_1}(c) \leq t_{z_0}.$$

Proof. This proof follows from the six lemmas, Theorem 8.1 and Corollary 8.1.1 in [1]. First, replace algorithmic network $\mathfrak{N}_{\mathrm{BB}}(N, f, t_{z_0}, \tau, j) = (G_{\mathrm{t}}, \mathfrak{P}_{\mathrm{BB}}(N), b_j)$ and its respective characteristics, for example, population $\mathfrak{P}_{\mathrm{BB}}(N)$ and family of graphs $\mathbb{G}(f, t, \tau)$, with $\mathfrak{N}'_{\mathrm{BB}}(N, f, t_{z_0}, j) = (G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j)$, $\mathfrak{P}'_{\mathrm{BB}}(N)$, $\mathbb{G}_{\mathrm{SIS}}(f, t)$, and others in the six lemmas, Theorem 8.1 and Corollary 8.1.1 in [1]. Note that in the proof of the sixth lemma, the average (singleton) diffusion density $\tau_{\mathbf{E}(\max)}$ is replaced with the prevalence $\tau_{\mathbf{E}(\rho)}$. Also note that in Corollary 8.1.1 in [1], the last time instant $t_{z+f(N,t_z,\tau)}$ is replaced with $c(z_0 + f(N, t_{z_0}) + 2) - c_0 - 1$. Then, the proof of Theorem 3 follows directly analogously to Theorem 8.2 in [1] (see also Theorem 2 in Section 2.4). □

4.2 Expected Emergent Open-Endedness from a Stationary Prevalence

Corollary 1. Let $w \in \mathbf{L}_{\mathbf{U}}$ be a network input. Let $0 < N \in \mathbb{N}$. Let $\mathfrak{N}'_{\mathrm{BB}}(N, f, t_{z_0}, j) = (G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j)$ be well defined. Let

$$\begin{aligned} c\colon\ & \mathbb{N} \to C_{\mathrm{BB}} \\ & x \mapsto c(x) = y \end{aligned}$$

be a total computable nondecreasing function where

$$c(z_0 + f(N, t_{z_0}) + 2) \geq c_0 + z_0 + f(N, t_{z_0}) + 2$$

and

$$c(z_0 + f(N, t_{z_0}) + 2) - c_0 - 1 \leq t_{|\mathrm{T}(G_{\mathrm{t}})|-1}.$$

If

$$f(N, t_{z_0}) = \mathbf{O}(\lg(N))$$

where every $G_{\mathrm{s}} \in \mathbb{G}_{\mathrm{BA}}(f, t_{z_0})$ achieves stationary prevalence ρ in a number of time intervals

$$\Delta^*_{t_{z_0}} \leq c(z_0 + f(N, t_{z_0}) + 2) - c_0 - 1$$

https://doi.org/10.25088/ComplexSystems.27.4.369

after time instant t_{z_0} and

$$\rho \sim \exp\left(-\frac{1}{m\lambda}\right) > \Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}\right) + 2\right),$$

then there is $t_{\mathrm{cen}_1}(c)$ such that

$$t_{\mathrm{cen}_1}(c) \leq t_{z_0}.$$

Proof. The proof follows directly from Theorem 3 and the definition of the algorithmic network $\mathfrak{N}'_{\mathrm{BB}}\left(N, f, t_{z_0}, j\right) = \left(G_{\mathrm{t}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j\right)$ by noting that:

$$z_0 + f\left(N, t_{z_0}\right) + 2 = z_0 + \mathbf{O}\left(\lg(N)\right) + 2 = \mathbf{O}\left(\lg(N)\right)$$

and that there is $\epsilon > 0$ such that

$$\frac{-1-\epsilon}{\epsilon_2} < 0 < C = \frac{\frac{1}{e^{(1/m\lambda)}} - \Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}\right) + 2\right) - \epsilon}{\Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}\right) + 2\right)} =$$
$$\left(\tau_{\mathrm{E}(\rho)}\left(N, f, t_{z_0}\right)\Big|_{t_{z_0}}^{c\left(z_0 + f\left(N, t_{z_0}\right) + 2\right) - c_0 - 1}\right.$$
$$\left. - \Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}\right) + 2\right) - \epsilon\right) \Big/$$
$$\left(\Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}\right) + 2\right)\right) \leq$$
$$\frac{1 - \Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}\right) + 2\right) - \epsilon}{\Omega\left(w, c_0 + z_0 + f\left(N, t_{z_0}\right) + 2\right)} \leq \frac{1}{\epsilon_2}. \square$$

4.3 Expected Emergent Open-Endedness from a Scale-Free Algorithmic Network

Corollary 2. Let $w \in \mathbf{L}_{\mathrm{U}}$ be a network input. Let $0 < N \in \mathbb{N}$. Let $\mathfrak{N}'_{\mathrm{BB}}\left(N, f, t_{z_0}, j\right) = \left(G_{\mathrm{s}}, \mathfrak{P}'_{\mathrm{BB}}(N), b_j\right)$ be well defined for every $G_{\mathrm{s}} \in \mathfrak{G}_{\mathrm{BA}}\left(f, t\right)$. Let

$$c\colon \mathbb{N} \to C_{\mathrm{BB}}$$
$$x \mapsto c(x) = y$$

be a total computable nondecreasing function where

$$c\left(z_0 + f\left(N, t_{z_0}\right) + 2\right) \geq c_0 + z_0 + f\left(N, t_{z_0}\right) + 2$$

and

$$c\left(z_0 + f\left(N, t_{z_0}\right) + 2\right) - c_0 - 1 \leq t_{|\mathrm{T}(G_{\mathrm{t}})|-1}.$$

If

$$f(N, t_{z_0}) = \mathbf{O}(\lg(N))$$

where every $G_t \in \mathbb{G}_{\text{SIS}}(f, t_{z_0})$ achieves stationary prevalence ρ in a number of time intervals

$$\Delta^*_{t_{z_0}} \leq c(z_0 + f(N, t_{z_0}) + 2) - c_0 - 1$$

after time instant t_{z_0}, then for a small enough value of $\lambda = \frac{\nu}{\delta}$, there are $t_{\text{cen}_1}(c)$ and a big enough value of m such that

$$t_0 = t_{\text{cen}_1}(c) \leq t_{z_0}.$$

Proof. Since, by supposition, $\mathfrak{N}'_{\text{BB}}(N, f, t_{z_0}, j) = (G_s, \mathfrak{P}'_{\text{BB}}(N), b_j)$ and $G_s \in \mathbb{G}_{\text{BA}}(f, t)$, then we will have from [8–10] that

$$\mathbb{G}_{\text{BA}} \subseteq \mathbb{G}_{\text{SIS}}$$

and

$$\rho \sim \exp\left(-\frac{1}{m\lambda}\right)$$

for sufficiently large populations and for a small enough value of λ. Thus, as Theorem 3 and Corollary 1 hold where the population size tends to ∞, we will have that condition

$$\rho \sim \exp\left(-\frac{1}{m\lambda}\right) > \Omega(w, c_0 + z_0 + f(N, t_{z_0}) + 2)$$

in Corollary 1 holds for a big enough value of m given a small enough value of λ. Thus, from Corollary 1, we will have that there is $t_{\text{cen}_1}(c)$ such that

$$t_{\text{cen}_1}(c) \leq t_{z_0}.$$

And, since every G_s is a static network, then

$$t_0 = t_{\text{cen}_1}(c) \leq t_{z_0}. \square$$

5. Conclusion

In this paper, we have presented a model for networked computable systems in order to investigate the problem of emergence of algorithmic complexity. In particular, we have mathematically investigated conditions that enable the triggering of emergent open-endedness, that is, the conditions that trigger an unlimited increase of emergent

complexity as the population size grows toward infinity. We have shown that these conditions are met by dynamic networks (or static networks) that exhibit a stationary prevalence of infected nodes under a susceptible-infected-susceptible (SIS) model for contagion of the fittest randomly generated node. As shown in [1], such research may be crucial for optimizing communication protocols in artificial networks of randomly generated systems that seek a better solution to a problem.

Our model for networked computable systems is based on that previously established in [1]. Nodes are randomly generated Turing machines that can send and receive information (partial outputs) as each node runs its computations until returning a final output, and edges (or arrows) are communication channels. Thus, as defined in [1], these algorithmic networks are composed of a synchronous population that follows a protocol of imitation of the "best information" shared by a neighbor. However, the present paper introduced a variation on this previous model such that this protocol is now followed under an SIS model [8–10].

In this paper, we have shown that, for big enough arbitrary values of $m \in \mathbb{N}$ compared to the effective spreading rate λ, if the time for achieving a stationary prevalence of infected nodes $\rho \sim \exp(-1/m\lambda)$ is upper bounded by a computably big enough function of $\mathbf{O}(\log(N))$ (e.g., as a function of the expected diameter or average shortest path length in scale-free networks or in classical random networks [40, 41]), then a lower bound for the expected emergent algorithmic complexity/information of a node goes to infinity as the network/population size N goes to infinity. That is, the average local irreducible information that emerges when nodes are networked (from a comparison with the isolated case) is expected to always increase for large enough populations of randomly generated Turing machines. Thus, these dynamic (or static) algorithmic networks with stationary prevalence cross the phase that we call expected local emergent open-endedness [1] for sufficiently large randomly generated populations.

In addition, since our main result only depends on assuming a stationary prevalence in the form of $\rho \sim \exp(-1/m\lambda)$, we have shown as a corollary from our theorems and from [8–10] that under the same conditions on m and λ, the same lower bound holds for static algorithmic networks with a scale-free degree distribution in the form of a power law $P(k) \sim 2m^2/k^3$ [21]. Therefore, synchronous algorithmic networks with a randomly generated population of computable systems and with a topology and a contagion-of-the-fittest model sufficiently close to the ones studied in [8–10] are also expected to display expected local emergent open-endedness.

These results suggest that contagion schemes like the SIS model, which have been shown to be important for studying epidemic and disease spreading and computer virus infections, may be also related to the emergence of complexity or irreducible information [1, 11] in networked systems. In this way, the present work and the investigation on networked computable systems using algorithmic networks in [1, 11, 12] are able to formally define sound and crucial properties and to prove fruitful theorems in order to grasp fundamental aspects and limitations of the theories. Such a theoretical approach to studying emergence of complexity or information in networked computable systems may help to understand and establish foundational properties on why an information dynamics within a system displaying synergistic or emergent behavior might be advantageous from a computational, evolutionary or game-theoretical point of view. Thus, as it is our goal to suggest in the present paper, these phenomena may be also related to infection dynamics [8–10] by considering an opposed but analogous perspective: contagion of the fittest (or the best solution for a problem) element in a population instead of contagion of pathological or undesirable elements.

Regarding only the lower bound for the expected emergent algorithmic complexity of a node, our main results show that a version of the halting probability for synchronous algorithmic networks may work like an asymptotic threshold for triggering expected local emergent open-endedness through a stationary prevalence. For example, in the case of the static algorithmic network with a Barabási–Albert scale-free degree distribution [21], we have shown that arbitrarily small values of the spreading rate can be overcome by big enough values of m (i.e., the number of new edges per node addition) in order to surpass this "threshold," triggering the expected local emergent open-endedness. However, since we have only investigated a lower bound, this halting probability may not actually be the threshold for the actual expected emergent algorithmic complexity of a node. Thus, in order to study the existence of such a threshold, we suggest for future research the investigation of an upper bound and an asymptotically tight bound for the expected local emergent algorithmic complexity.

Acknowledgments

In [12], some preliminary results of this paper are presented as an extended abstract. In [22], an extended version of this paper is presented as research report no. 5/2018 at the National Laboratory for Scientific Computing (LNCC). The authors acknowledge partial support from CNPq through their individual grants: F. S. Abrahão (313.043/2016-7), K. Wehmuth (312599/2016-1) and A. Ziviani

(308.729/2015-3). The authors also acknowledge the INCT in Data Science – INCT-CiD (CNPq 465.560/2014-8) and partial support from CAPES, FAPESP and FAPERJ.

References

[1] F. S. Abrahão, K. Wehmuth and A. Ziviani, *Algorithmic Networks: Central Time to Trigger Expected Emergent Open-Endedness*, Research Report no. 4/2018 at the National Laboratory for Scientific Computing (LNCC), 2017. arxiv.org/abs/1708.09149.

[2] L. Margulis, *Symbiosis in Cell Evolution: Life and Its Environment on the Early Earth*, San Francisco: W. H. Freeman, 1981 p. 419.

[3] R. Axelrod, *The Evolution of Cooperation*, rev. ed., New York: Basic Books, 2006.

[4] M. Oizumi, L. Albantakis and G. Tononi, "From the Phenomenology to the Mechanisms of Consciousness: Integrated Information Theory 3.0," *PLoS Computational Biology*, **10**(5), 2014 e1003588. doi:10.1371/journal.pcbi.1003588.

[5] H. Kim, P. Davies and S. I. Walker, "New Scaling Relation for Information Transfer in Biological Networks," *Journal of The Royal Society Interface*, **12**(113), 2015 20150944. doi:10.1098/rsif.2015.0944.

[6] F. Schweitzer, G. Fagiolo, D. Sornette, F. Vega-Redondo and D. R. White, "Economic Networks: What Do We Know and What Do We Need to Know?," *Advances in Complex Systems*, **12**(04n05), 2009 pp. 407–422. doi:10.1142/S0219525909002337.

[7] J. H. Miller and S. E. Page, *Complex Adaptive Systems: An Introduction to Computational Models of Social Life*, Princeton, NJ: Princeton University Press, 2007.

[8] R. Pastor-Satorras and A. Vespignani, "Epidemic Dynamics and Endemic States in Complex Networks," *Physical Review E*, **63**(6), 2001 066117. doi:10.1103/PhysRevE.63.066117.

[9] R. Pastor-Satorras and A. Vespignani, "Epidemic Spreading in Scale-Free Networks," *Physical Review Letters*, **86**(14), 2001 pp. 3200–3203. doi:10.1103/PhysRevLett.86.3200.

[10] R. Pastor-Satorras and A. Vespignani, "Immunization of Complex Networks," *Physical Review E*, **65**(3), 2002 036104. doi:10.1103/PhysRevE.65.036104.

[11] F. S. Abrahão, "Emergent Algorithmic Creativity on Networked Turing Machines," in *The 8th International Workshop on Guided Self-Organization at the 15th International Conference on the Synthesis and Simulation of Living Systems (ALIFE)*, Cancún, 2016. Extended abstract available at: www.guided-self.org/uploads/2/1/7/6/21762362/abrahao.pdf.

[12] F. S. Abrahão, K. Wehmuth and A. Ziviani, "Expected Emergence of Algorithmic Information from a Lower Bound for Stationary Prevalence," in *Brazilian Computer Society Congress 2018 (CSBC 2018)*, Natal, 2018. Brazilian Computer Society (SBC). doi:10.5281/zenodo.1241237.

[13] N. T. J. Bailey, *The Mathematical Theory of Infectious Diseases and Its Applications*, London: Griffin, 1975.

[14] S. Hernández-Orozco, F. Hernández-Quiroz and H. Zenil, "Undecidability and Irreducibility Conditions for Open-Ended Evolution and Emergence," *Artificial Life*, **24**(1), 2018 pp. 56–70. doi:10.1162/ARTL_a_ 00254.

[15] M. Prokopenko, ed., *Guided Self-Organization: Inception*, New York: Springer, 2014. link.springer.com/10.1007/978-3-642-53734-9.

[16] M. Prokopenko, F. Boschetti and A. J. Ryan, "An Information-Theoretic Primer on Complexity, Self-Organization, and Emergence," *Complexity*, **15**(1), 2009 pp. 11–28. doi:10.1002/cplx.20249.

[17] H. Buhrman, M. Li, J. Tromp and P. Vitányi, "Kolmogorov Random Graphs and the Incompressibility Method," *SIAM Journal on Computing*, **29**(2), 1999 pp. 590–599. doi:10.1137/S0097539797327805.

[18] A. Mowshowitz and M. Dehmer, "Entropy and the Complexity of Graphs Revisited," *Entropy*, **14**(3), 2012 pp. 559–570. doi:10.3390/e14030559.

[19] R. V. Solé and S. Valverde, "Information Theory of Complex Networks: On Evolution and Architectural Constraints," *Complex Networks* (E. Ben-Naim, H. Frauenfelder and Z. Toroczkai, eds.), Berlin, Heidelberg: Springer, 2004 pp. 189–207. doi:10.1007/978-3-540-44485-5_9.

[20] H. Zenil, F. Soler-Toscano, K. Dingle and A. A. Louis, "Correlation of Automorphism Group Size and Topological Properties with Program-Size Complexity Evaluations of Graphs and Complex Networks," *Physica A: Statistical Mechanics and Its Applications*, **404**, 2014 pp. 341–358. doi:10.1016/j.physa.2014.02.060.

[21] A.-L. Barabási, R. Albert and H. Jeong, "Mean-Field Theory for Scale-Free Random Networks," *Physica A: Statistical Mechanics and Its Applications*, **272**(1–2), 1999 pp. 173–187. doi:10.1016/S0378-4371(99)00291-5.

[22] F. S. Abrahão, K. Wehmuth and A. Ziviani, *Emergent Open-Endedness from Contagion of the Fittest*, Research Report no. 5/2018, National Laboratory for Scientific Computing (LNCC), Petrópolis, 2018. arxiv.org/abs/1806.07254.

[23] K. Wehmuth, É. Fleury and A. Ziviani, "MultiAspect Graphs: Algebraic Representation and Algorithms," *Algorithms*, **10**(1), 2017 p. 1. doi:10.3390/a10010001.

[24] K. Wehmuth, É. Fleury and A. Ziviani, "On MultiAspect Graphs," *Theoretical Computer Science*, **651**, 2016 pp. 50–61. doi:10.1016/j.tcs.2016.08.017.

[25] K. Wehmuth and A. Ziviani, "Centralities in High Order Networks," *Encontro de Teoria da Computação (ETC_CSBC)*, **3**(1/2018). portaldeconteudo.sbc.org.br/index.php/etc/article/view/3147.

[26] G. Chaitin, V. M. F. G. Chaitin and F. S. Abrahão, "Metabiología: los orígenes de la creatividad biológica," *Investigación y Ciencia*, **448**, 2014 pp. 74–80. www.investigacionyciencia.es/revistas/investigacion-y-ciencia/la-era-de-los-macrodatos-591/metabiologa-los-orgenes-de-la-creatividad-biolgica-11696.

[27] F. S. Abrahão, "The 'Paradox' of Computability and a Recursive Relative Version of the Busy Beaver Function," *Information and Complexity* (C. Calude and M. Burgin, eds.), Singapore: World Scientific Publishing, 2017 pp. 3–15. doi:10.1142/9789813109032_0001.

[28] V. M. F. G. Chaitin and G. J. Chaitin, "A Philosophical Perspective on a Metatheory of Biological Evolution," *The Map and the Territory: Exploring the Foundations of Science, Thought and Reality* (S. Wuppuluri and F. A. Doria, eds.), Cham: Springer, 2018 pp. 513–532. doi:10.1007/978-3-319-72478-2_ 29.

[29] E. C. Costa, A. B. Vieira, K. Wehmuth, A. Ziviani and A. P. C. da Silva, "Time Centrality in Dynamic Complex Networks," *Advances in Complex Systems*, **18**(07n08), 2015 1550023. doi:10.1142/S021952591550023X.

[30] K. Wehmuth, A. Ziviani and E. Fleury, "A Unifying Model for Representing Time-Varying Graphs," in *IEEE International Conference on Data Science and Advanced Analytics (DSAA 2015)*, Paris, France, 2015, IEEE, 2015. doi:10.1109/DSAA.2015.7344810.

[31] A. Adams, H. Zenil, P. C. W. Davies and S. I. Walker, "Formal Definitions of Unbounded Evolution and Innovation Reveal Universal Mechanisms for Open-Ended Evolution in Dynamical Systems," *Scientific Reports*, **7**(1), 2017 p. 997. doi:10.1038/s41598-017-00810-8.

[32] G. Chaitin, "Life as Evolving Software," *A Computable Universe: Understanding and Exploring Nature as Computation* (H. Zenil, ed.), Hackensack, NJ: World Scientific Publishing, 2012 pp. 277–302. www.worldscientific.com/doi/abs/10.1142/9789814374309_ 0015.

[33] G. Chaitin, *Proving Darwin: Making Biology Mathematical*, New York: Pantheon Books, 2012.

[34] S. Hernández-Orozco, N. A. Kiani and H. Zenil, "Algorithmically Probable Mutations Reproduce Aspects of Evolution Such as Convergence Rate, Genetic Memory, and Modularity." arxiv.org/abs/1709.00268.

[35] I. M. L. D'Ottaviano and E. B. Filho, "Auto-Organização e Criação," *MultiCiência: Revista Interdisciplinar dos Centros e Núcleos da Unicamp*, **3**, 2004 pp. 1–23. www.multiciencia.unicamp.br/rede_2_ 3.htm.

[36] N. Fernández, C. Maldonado and C. Gershenson, "Information Measures of Complexity, Emergence, Self-Organization, Homeostasis, and Autopoiesis," *Guided Self-Organization: Inception*, Vol. 9 (M. Prokopenko, ed.), Berlin, Heidelberg: Springer, 2014 pp. 19–51. link.springer.com/10.1007/978-3-642-53734-9_2.

[37] C. Gershenson, D. Aerts and B. Edmonds, eds., *Worldviews, Science and Us: Philosophy and Complexity*, New Jersey: World Scientific, 2007.

[38] A. Guimarães, A. B. Vieira, A. P. C. Silva and A. Ziviani, "Fast Centrality-Driven Diffusion in Dynamic Networks," in *Proceedings of the 22nd International Conference on World Wide Web (WWW'13 Companion)*, New York: ACM Press, 2013 pp. 821–828. doi:10.1145/2487788.2488055.

[39] A.-L. Barabási and R. Albert, "Emergence of Scaling in Random Networks," *Science*, **286**(5439), 1999 pp. 509–512. doi:10.1126/science.286.5439.509.

[40] B. Bollobás and O. Riordan, "The Diameter of a Scale-Free Random Graph," *Combinatorica*, **24**(1), 2004 pp. 5–34. doi:10.1007/s00493-004-0002-2.

[41] T. G. Lewis, *Network Science*, Hoboken, NJ: John Wiley & Sons, Inc., 2009. doi:10.1002/9780470400791.