

RESEARCH ARTICLE

The role of the interaction network in the emergence of diversity of behavior

Alan Godoy^{1,2*}, Pedro Tabacof³, Fernando J. Von Zuben¹

1 Laboratory of Bioinformatics and Bioinspired Computing, School of Electrical and Computer Engineering, University of Campinas, Campinas, São Paulo, Brazil, **2** CPqD Foundation, Campinas, São Paulo, Brazil, **3** RECOD Lab, School of Electrical and Computer Engineering, University of Campinas, Campinas, São Paulo, Brazil

* godoy@dca.fee.unicamp.br



Abstract

How can systems in which individuals' inner workings are very similar to each other, as neural networks or ant colonies, produce so many qualitatively different behaviors, giving rise to roles and specialization? In this work, we bring new perspectives to this question by focusing on the underlying network that defines how individuals in these systems interact. We applied a genetic algorithm to optimize rules and connections of cellular automata in order to solve the density classification task, a classical problem used to study emergent behaviors in decentralized computational systems. The networks used were all generated by the introduction of shortcuts in an originally regular topology, following the small-world model. Even though all cells follow the exact same rules, we observed the existence of different classes of cells' behaviors in the best cellular automata found—most cells were responsible for memory and others for integration of information. Through the analysis of structural measures and patterns of connections (motifs) in successful cellular automata, we observed that the distribution of shortcuts between distant regions and the speed in which a cell can gather information from different parts of the system seem to be the main factors for the specialization we observed, demonstrating how heterogeneity in a network can create heterogeneity of behavior.

OPEN ACCESS

Citation: Godoy A, Tabacof P, Von Zuben FJ (2017) The role of the interaction network in the emergence of diversity of behavior. PLoS ONE 12 (2): e0172073. doi:10.1371/journal.pone.0172073

Editor: Dante R. Chialvo, Consejo Nacional de Investigaciones Científicas y Técnicas, ARGENTINA

Received: October 11, 2016

Accepted: January 30, 2017

Published: February 24, 2017

Copyright: © 2017 Godoy et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All code used for simulation and analysis are available at: https://github.com/unicamp-lbic/small_world_ca.

Funding: Conselho Nacional de Desenvolvimento Científico e Tecnológico (<http://cnpq.br/>) provided a doctorate fellowship to Alan Godoy (142118/2010-9). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Introduction

“All animals are equal, but some animals are more equal than others.”

In the book *Animal Farm* [1], George Orwell tells us a story in which animals in a farm, tired of being exploited by humans, got rid of the farmer and created an egalitarian society, totally controlled by animals. Some time after such rebellion, however, a special class appeared—the pigs –, which did not produce any food and only took care of bureaucratic work. At one point the pigs acquired so many privileges and got so similar to humans that they replaced all revolutionary principles by just the one that opens this paper.

Apart from the political allegory expressed in the book, *Animal Farm* can draw our attention to a question relevant to many disciplines: how can a system composed of equal (or very similar) agents display qualitatively different behaviors? This is a particularly relevant question because, in most cases, instead of being only deleterious, as in Orwell's book, differences in individual behavior may have positive effects. Consider, for instance, the human brain: while there are only a few hundred different types of neurons [2], the number of behavioral patterns in the brain is countless. Some neurons process auditory information, others are related to vision and others to empathy [2, 3]. Some people even have neurons that react only to images of specific celebrities, like Jennifer Aniston or Bill Clinton [4]. Human groups are another example of systems that can find some benefits in heterogeneity, as there is compelling evidence that diversity can improve a team's performance in collective problem solving [5]. Indeed, human societies are remarkable for collectively solving complex problems, as the use of markets to organize labor division and control shared resources, and for learning and diffusion of information through science and culture [6, 7].

Heterogeneity is particularly intriguing in self-organized systems, as, having no central element controlling the role to be performed by each agent or how individual results should be combined, any necessary decision about task allocation or specialization is distributed among all agents [8].

Surely, in many cases heterogeneity is influenced by diversity in intrinsic features of its components—e.g.: the different types of neurons or genetic differences between humans in our two previous examples—, but in many situations these inner characteristics are not sufficient to explain the wide range of behaviors and specialization observed. A neuron is activated when one sees a picture of a celebrity not because there is a specific type of neuron that is good at identifying whether a person is famous or not, but as a consequence of its position within the neural network that connects the eye to the visual cortex to other regions of the brain [2]. Similarly, one would hardly agree, for instance, with the hypothesis of the existence of a “butcher gene” or a “lawyer gene”, i.e., that our genetic codes are the only factor determining the professional roles humans take in society. Physical and social environments have a crucial part in defining how we humans act and differentiate from each other [9, 10], so that the structure of our social connections has great influence on individuality [11].

Communication between agents is essential in self-organized systems [12] and it can happen according to a more or less rigid structure: in animal groups there are few restrictions limiting the potential contacts of an individual, so that an ant can interact with any other nestmate when looking for food sources or for potential new nests [13, 14]. In neural networks, on the other hand, synaptic connections are well delimited and stable in time [2, 15, 16], so that neurons can only transmit signals to their peers to which it is connected by an axon terminal. These synapses are persistent but dynamic, as connections can be created, tuned, or pruned to encode new knowledge or ability learned by the network.

Different patterns of connection can provide different functionalities in systems structured according to a network [17], like resilience to random failure of nodes (e.g.: neurons, people, computers) or connections (e.g.: synapses, social contacts, wires), more information storage capacity, ease of navigation within the network, influence concentration, and/or low average distance between each pair of nodes [18–21]. Local structures can have large effects on a network's behavior: *motifs* (see Fig 1)—small patterns of connection that occur in a network more commonly than expected in a random network with similar characteristics—are believed to act as basic functional units in complex systems, having a similar role to that performed by logic gates in electronic circuits [22, 23].

Therefore, if we want to understand collective computation we must first understand how connections affect the flow and processing of information in complex systems. In this study,

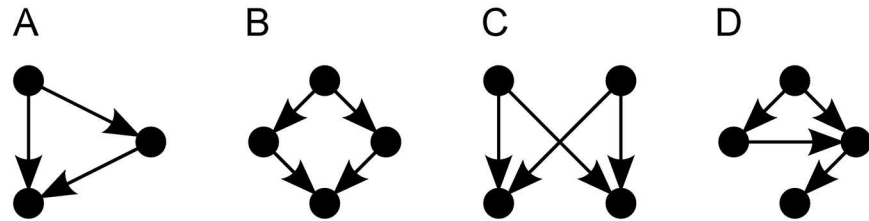


Fig 1. Motifs found in the neural network of the worm *C. elegans*. Motifs names: (A) Feed-forward loop, (B) Bi-fan and (C) Bi-parallel; motif (D) has no common aliases. After Milo *et al.* [23].

doi:10.1371/journal.pone.0172073.g001

we want to shed some light on how network randomness can promote individual behavior heterogeneity in a system in which all individuals follow exactly the same rules, studying also how this randomness affects collective computation in complex systems and can induce specialization. For this, we combined a simple model of complex system known as *elementary cellular automaton* (ECA) with small-world networks—a family of networks capable of displaying some non-trivial characteristics exhibited by networked systems in the real-world –, using evolutionary computation to explore how individual behaviors and collective performance are correlated to communication structure, specially to the presence of direct connections between otherwise distant agents.

As we explain in the next section, most previous works studied computation in complex systems by analyzing global properties of cellular automata, which were obtained by the evolution only of rules followed by individuals. We, in turn, choose to evolve both networks and rules and to focus our analysis on the differences between behaviors of individual cells in a cellular automaton. For this we used information theoretical measures to describe such behaviors, and investigated how topological characteristics and local patterns of connections between individuals shape these differences.

Background

As previously stated, we used in this work cellular automaton, a classic agent-based model. However, instead of following the common regular network, the communication in this study was structured according to a small-world network. If you are familiar with cellular automata, complex networks and previous works on the evolution of rules and networks for elementary cellular automata, you can gladly jump to the next section.

Simple models for distributed systems

In this work, we used computation as an analogy to understand the information processing happening in self-organized systems. Currently, the most common computer architecture is the von Neumann architecture, in which one processor (CPU) has access to a memory (RAM) where program instructions and data are stored. Roughly speaking, the CPU fetches instructions from the memory and executes them, using this same memory to store all inputs and outputs of these computations. Complex and other massively distributed systems, however, are very different from such usual model of computation: while in a von Neumann architecture the CPU has access to the whole memory, in a complex system each individual (or processing unit) has complete access only to its own state (we can refer to this state as the individual memory) and can gather information solely from those individuals to which it is directly connected [8]. To exemplify such limitation, we can think about neurons: when deciding whether to fire

or not, a neuron does not know the state of the whole brain but only of the neurons to which it is connected [2].

So, if we hope to study processes happening in complex systems we will need to use a model other than the von Neumann architecture. One of the simplest yet most powerful models for this kind of systems is the *cellular automaton* (known as CA). A CA is composed of a set of cells, each of which works like this:

- The cell has an *internal state*, which works as a (very) small memory—the number of possible states in which a cell may be is finite and fixed;
- In addition to its own state, the cell knows the current state of a given set of other cells, known as *neighbors* or contacts;
- At each step, the cell will update its state following a *rule table* that has as input the current state of the cell and of its neighbors and yields as output the next state for this cell.

Each cell may have a particular set of possible states or follow a different rule table [24]. However, the most common model, that was also used in this work, is the *uniform* cellular automaton, in which all cells are equal to each other. As each piece of this kind of cellular automaton is too simple, you might wonder about its ability to perform complex computation. Surprisingly (at least for some people), even for cellular automata in which the cells have only two possible states, there are combinations of neighborhoods and rule tables that can enable such CAs to solve each and every problem that can be solved by the most powerful computational model, the universal Turing machine [8, 25].

The simplest “flavor” of cellular automaton is the elementary cellular automaton [26]. As shown in Fig 2, an ECA is composed of a fixed number of uniform cells, organized one beside the other, forming a (one dimensional) ring. Each cell can be in one of two states (0 or 1) and has direct contact only to their k closest peers, updating their states synchronously. Even though agents operate asynchronously in nature, this kind of CA has a dynamic rich enough to model many systems in the real-world. Even when $k = 3$ (i.e., a cell can only communicate to itself and its immediate neighbors to the left and to the right) some ECAs are able to perform universal computation [27], being some of the simplest machines able to do so. In Fig 3, you can see an example of the execution of one of such rules and the elaborate patterns it is able to produce. This combination of simplicity and ability to produce complex behavior, embedded within an inherently distributed system, is so impressive that some researchers, like Stephen Wolfram, see ECAs as key to explain how complexity arises from simple rules in nature [26]. With such model we can simulate idealized versions of complex systems to study the origins of *self-organized* processes and *emergent behaviors*—hard to predict behaviors that arise from the interaction of many individuals following simple rule –, being able, thus, to analyze the collective computation we see in systems like human brains and animal societies.

A model problem commonly used to study computation in cellular automata is the *density classification task* (also known by its acronym, DCT, or by a different name, *majority problem*) [8]: we initialize each cell in a CA with an arbitrary state, 0 or 1, and, after a fixed number of steps, all the cells must have converged to 1 if more than half of them were initialized with 1 (i.e., the initial density ρ_0 was greater than 0.5), or 0 otherwise ($\rho_0 < 0.5$). This sounds as a pretty simple problem, but it has been proven that no ECA with finite k can solve the DCT perfectly [28]. This is an easy task for a machine with access to the whole memory, but not for a distributed system where each piece knows only local information: you can understand the difficulty of such problem, for instance, by thinking about the challenge of knowing the average opinion of a country’s population by talking only to people who live near you.

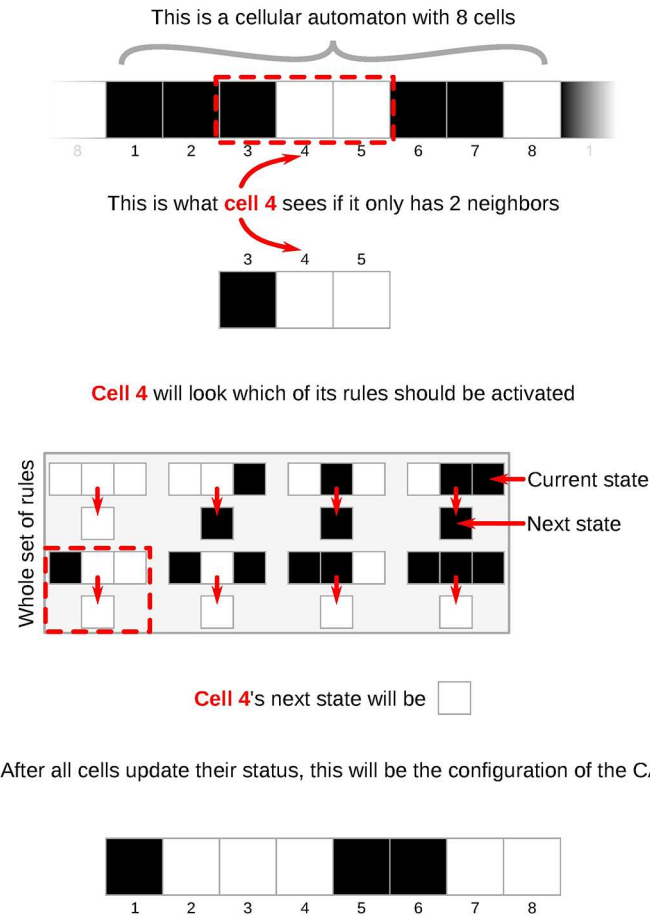


Fig 2. How elementary cellular automata work. Notice that all cells follow the same set of rules. The state of each cell is black (0) or white (1). The cellular automaton is uniform in the sense that all cells follow the same rule table to update their states.

doi:10.1371/journal.pone.0172073.g002

Instead of reducing the interest for the DCT, the non-existence of a perfect solution for this task puts it in a privileged position for the study of complex systems, as it is both a very simple problem to formulate and a computationally difficult task. Thus, many researchers have looked for ECAs able to correctly perform density classification for as many initial configurations as possible, the first of which was Norman Packard. Packard analyzed DCT's rule space [29] and used evolutionary computation to list some rules able to achieve good performance in that problem, studying their common characteristics [30].

The use of evolutionary techniques to search for efficient rules for the DCT (and other problems) was further explored by Melanie Mitchell, Peter Hrabar, Rajarshi Das and James Crutchfield in a series of influential papers [31–33]. Two years after that, David Andre, Forrest Bennet and John Koza [34], using genetic programming, discovered the first computer-designed rule able to beat the performance of hand-coded ECAs. Before that, one of the best results was achieved by the rule GKL [35], which can be summarized as follows: if a cell is in state 0, then its next state is defined by the majority among itself, its nearest neighbor to the left and the neighbor three places to the left; if it is in state 1, symmetrically, it follows the

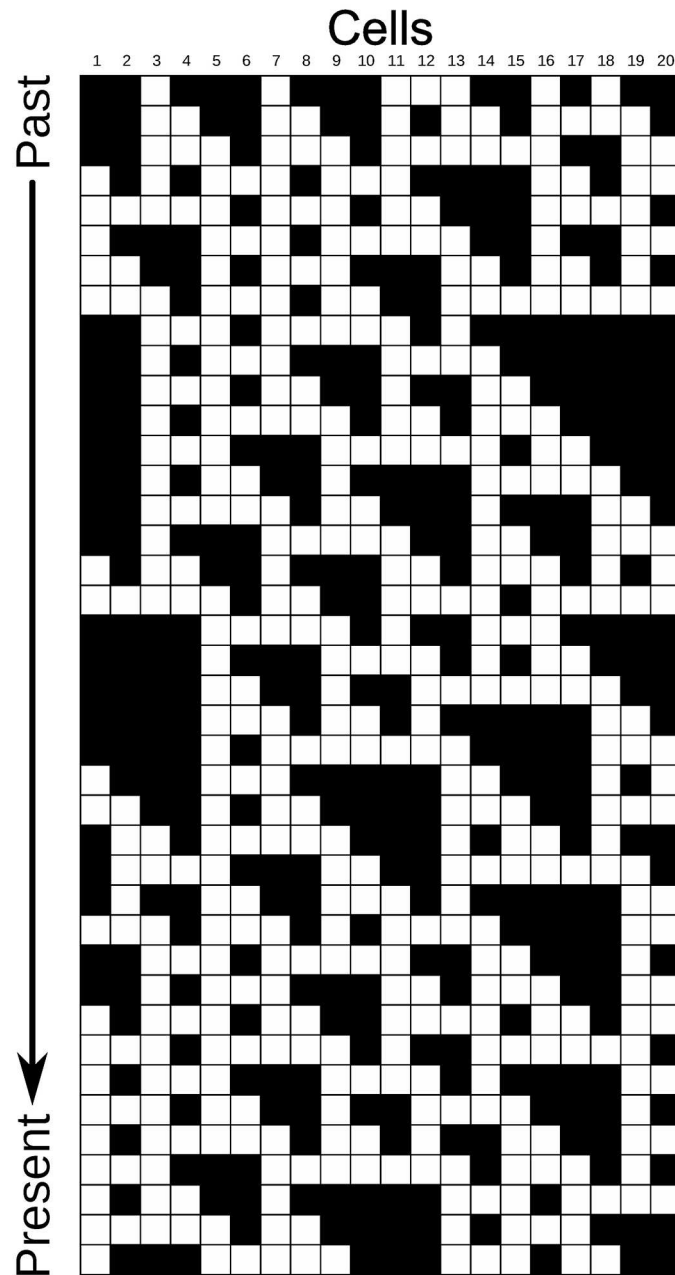


Fig 3. Space-time diagram of an ECA composed of 20 cells, each following the rule 110. See subsection “Searching good CAs” for an explanation on rule numbering. The initial state is exhibited at the top row. Cells 1 and 20 are neighbors.

doi:10.1371/journal.pone.0172073.g003

majority among itself, its nearest neighbor to the right and the neighbor three places to the right.

In 1998, Juille and Pollack [36] obtained a better rule using a coevolutionary approach. Currently, due to an emergent phenomenon, two rules share the best results in the DCT problem [37]. While the best solution in grids with an even number of cells is still the JP rule, for odd-sized CAs the best results are obtained by the WO rule [38], which was found in 2008,

Table 1. Some of the rules with best performance in the DCT (in a grid with $N = 149$ and $k = 7$). Results are ordered according to the year they were discovered.

Rule	Year	Performance
GKL [35]	1978	81.6%
Mitchell et al. [31]	1993	76.9%
ABK [34]	1996	82.3%
JP [36]	1998	85.9%
WO [38]	2008	89.0%

doi:10.1371/journal.pone.0172073.t001

combining a multi-level selection evolutionary algorithm with a bias favoring the selection of symmetric rules.

The results achieved by these rules are summarized in Table 1.

Networks in a nutshell

Real-world networks rarely are as organized as the ring topologies used in the studies we have just presented. The reality is much “messier”! So let us make a brief pause in our discussion about cellular automata and focus for a moment on networks.

Until few years ago, networks seen in real systems, like neural and social networks, were considered too complex and, thus, were modeled as if they were completely random [19]. As we had access to more data, more computing power and better analytical tools, however, researchers started noticing that many real-world networks share some common properties that are not expected in purely random or regular networks. An example is the *small-world effect*, formally defined by Duncan Watts and Steven Strogatz [39] in a seminal paper published in 1998, which is the simultaneous occurrence in a network of a *small average distance* between each pair of nodes and a *high clustering coefficient* (i.e., a high probability that two vertices that are both neighbors to the same third vertex are also directly connected to each other). Watts and Strogatz showed that, while this effect is present, for instance, in human social networks, in power grids and in the neural network of the worm *Caenorhabditis elegans*, no lattice, ring or completely random graph is able to display both small average distances and high clustering. To produce more realistic networks they proposed a new model to generate networks, the *small-world model*, summarized in two components (see Fig 4):

- **Initialization:** the network is started as a ring in which each node is connected to its k closest neighbors;
- **Rewiring:** every edge in the initial network has a probability p of being *rewired*, i.e., of being disconnected from one of its ends and attached to a randomly chosen vertex (avoiding any self-loop or repeated edges).

By choosing a small value for p (usually less than 0.1), it is possible to preserve the local structure with high clustering of the initial ring, while the few long-distance *shortcuts* added are able to dramatically reduce the average distance between nodes.

The small-world effect is not the only common phenomenon missed by random and regular networks: the distribution of degrees (i.e., the distribution of the number of edges connected to each node) in many real-world networks deviate significantly from the Poisson or delta distributions expected for completely random graphs, lattices, rings and small-world networks. Instead, their degree distributions follow power laws—in these networks the probability

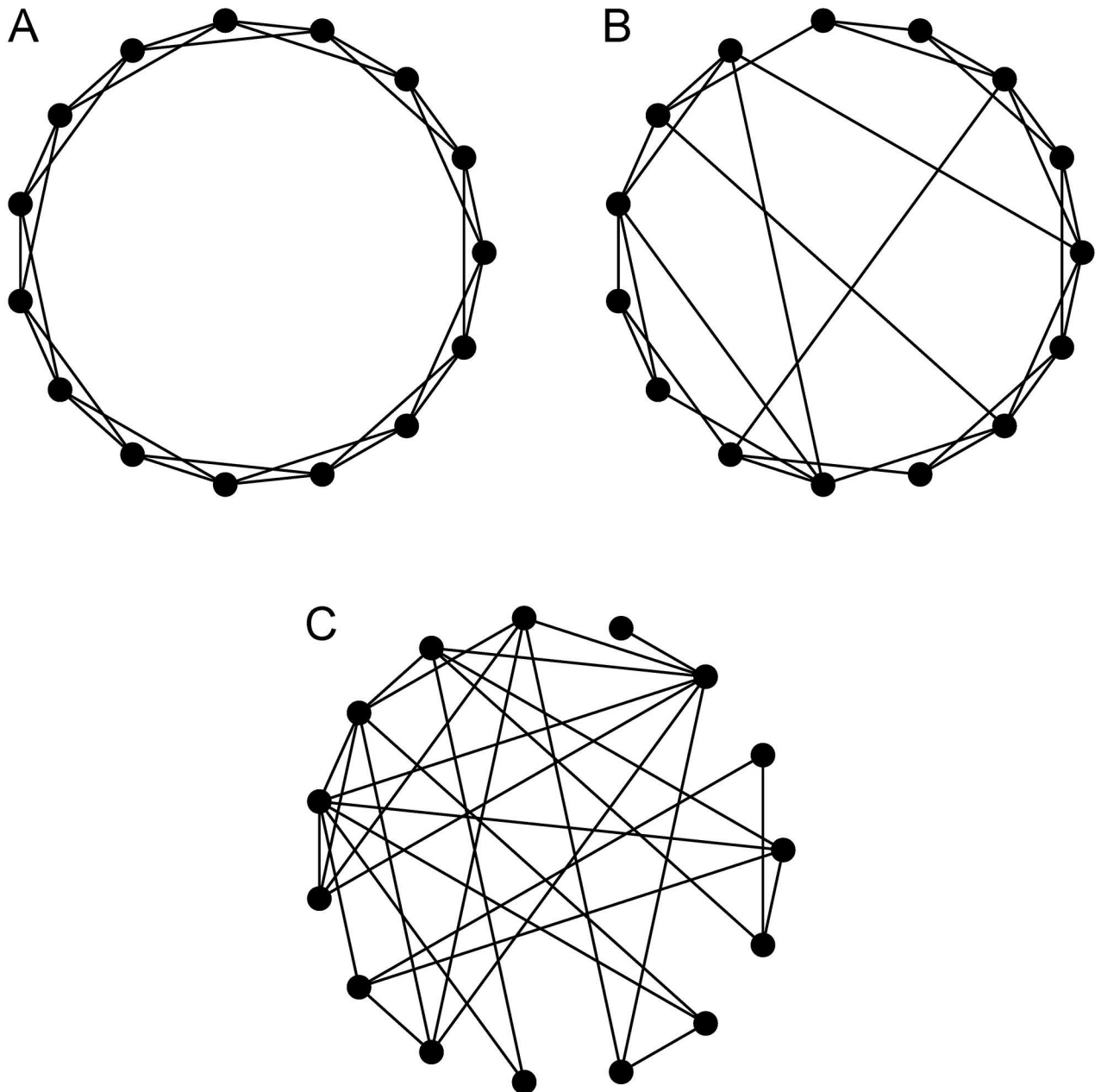


Fig 4. Graphs generated using the small-world model with different rewiring probabilities p . (A) $p = 0.00$, (B) $p = 0.15$ and (C) $p = 1.00$.

doi:10.1371/journal.pone.0172073.g004

that a random node has exactly k connections is given by $p(k) \propto k^{-\gamma}$, for a given positive γ . This result was published in 1999 by Albert-László Barabási and Réka Albert [40], after studying networks as those formed by citations between scientific publications, links between web pages and electric power grids. Like Watts and Strogatz, they also proposed a new network model, the scale-free model, which is able to produce networks with degrees following a power-law. Its basic components are the continuous inclusion of new nodes and edges and preferential attachment, a mechanism that selects nodes to receive new edges with probability proportional to the number of connections these nodes already have.

As we stated before, topologies with different characteristics can provide different functionalities to a system. For instance, the connectivity of networks with degrees following a power law has higher resilience against random removal of nodes, when compared to purely random graphs. However, this connectivity degenerates very quickly when the most connected nodes are removed, which makes them vulnerable to targeted attacks [19]. Initial studies also suggest that topologies combining scale-free networks and the small-world effect may be the result of an optimization process [41, 42], as they provide a good compromise between high connectivity between nodes and low demand for links. It is reasonable, thus, to suppose that the behavior of a distributed system, as an ECA, may differ a lot when we replace the usual regular topologies by more realistic networks to define how its units interact.

The impact of topology on computation

Most of the research about collective computation has been centered on understanding the impact of individual behaviors on the global dynamics. Some works, however, studied the impact of modifying the topology connecting the individuals. Maybe the earliest work on this subject was the study of random Boolean networks (RBNs) by Stuart Kauffman [43]. In Kauffman's original model, which was created to model genetic regulatory networks, each individual has a randomly defined Boolean activation function, being connected to each other according to a completely random topology. In these networks, we observe cyclic behaviors that may be classified as robust or chaotic, according to the system's reaction to perturbations (which, in turn, is determined by the interplay between topological and behavioral parameters) [44].

More recently, in 1997, Moshe Sipper and Eytan Ruppin [45] studied a class of cellular automata that, differently from ECAs, allow each cell to evolve a different rule table. After evaluating how different topologies affect the ability of CAs to solve the DCT problem, they identified that a cellular automaton's performance is strongly dependent on the average distance between its cells—the smaller the distance, the higher is the fitness. This is not an unexpected result: going back to our analogy of the DCT problem with opinion averaging, the insertion of long-range connections is similar to allowing you to talk not only to people in your neighborhood, but also to friends that live in different cities. The more shortcuts one has access to, the more one has access to diverse, less correlated opinions.

When studying small-world networks and their possible impacts on dynamical systems, Watts and Strogatz used this class of networks to define the neighborhood in ECAs, while keeping the individual behaviors fixed [39, 46]. They analyzed the effect of these topologies on the ability of such systems to solve the density classification task. They found that, for the same number of cells and average degree, such small-world ECAs were able to outperform even the best rules at that time which made use of the traditional ring topology.

Digging deeper on the effects of different neighborhood structures on the ability of cellular automata to solve computational problems, Marco Tomassini, Christian Darabos and Mario Giacobini [47, 48] used evolutionary computation to look for classes of topologies with which ECAs were able to achieve good performance in the density classification and synchronization tasks. As in Watts' work, they kept fixed the rule used by the cells to update their states. The authors found that the networks able to produce the best results had some characteristics similar to small-world networks, both when they started the evolution from lattices or from random graphs. In agreement to Watts [46], the authors suggested that it is easier to get good results evolving networks when compared to evolving rules, as they obtained many ECAs able to solve more than 80% of the initial configurations, while it is very rare for lattices to achieve such performance. For instance, Crutchfield et al. [49] evolved high-scoring strategies only in 9 out of 300 runs of the evolutionary algorithm. A second interesting result they observed was

that ECAs using small-world neighborhoods are more robust to random failures in cells (i.e., a cell updates its state to the incorrect output with probability p_{fail}) than regular lattices. In subsequent works [50–52], they compared the effects of using small-world and scale-free networks to structure the communication in cellular automata and random Boolean networks, observing that, when solving the DCT or the synchronization problems, small-world networks achieve higher performance and are more robust to random failures in cells than Barabási-Albert networks, though both networks have similar resilience to link removal in networks with many connections.

More evidence about the impact of topology on a system's dynamics is given by Macêdo *et al.* [53] and Oikonomou and Cluzel [54]. Heverton Macêdo, Gina Oliveira and Carlos Ribeiro [53] evaluated the effect of adding random shortcuts to a network, going from lattices to small-world networks, noticing that more shortcuts lead to more rapid deviation between two cellular automata with similar initial states. Panos Oikonomou and Philippe Cluzel [54] used Boolean threshold networks—a framework similar to cellular automata, but in which connections have weights that can be either positive (excitatory) or negative (inhibitory)—to study how topology may affect the natural evolution of networked systems. Their results indicate that the evolution of systems structured according to scale-free networks is faster and more continuous than that of random networks, which is characterized by large plateaus with sparse improvements in evolution.

A broad analysis performed by Carsten Marr and Marc-Thorsten Hütt [55] used different classes of networks—e.g., rings with different k 's and small-world networks—to study how topology can change the behavior of a dynamic system. Their results indicate that, simply by changing the network used to structure interactions between agents, one can make, for instance, an initially homogeneous system to behave chaotically or a system with complex dynamics to enter a periodic regime. In their work, the authors also found that the lower the degree of a node, the more uncertain its behavior—as they have few influences, the aggregated information of these individuals is subject to larger fluctuations. This heterogeneity of behaviors among individuals following the same set of rules is a very interesting finding that we will further explore in this paper.

Methods

In this work we intended to further investigate how networks defining the interactions in complex systems impact the computation occurring in such systems and their performance when trying to solve problems. In a previous study [17], we have already showed that networks with small-world and scale-free characteristics could enhance the ability of multi-agent systems in collective problem solving. Now we are interested, more precisely, in understanding how information processing is affected by the presence of shortcuts in small-world networks and how topology is correlated with individual behaviors and collective performance.

Therefore, we replicated the experiment proposed by Melanie Mitchell and her colleagues [31], evolving simple CAs to achieve good performance in the density classification task, but with a small and crucial change relative to most of the previous works in literature: instead of evolving only rules, we also evolved the networks that define interactions between cells, keeping fixed, however, the number of shortcuts a CA was able to add to its basic ring network. Evolving rules and network together has a profound impact on the observed results because there is interaction between topology and rules when defining a group's global behavior [17]: while a network may improve the performance of a system using a specific rule, it can worsen the results achieved by another system in which elements have a slightly different individual behavior.

In many other works trying to explain computation in complex systems, researchers were interested in improving the performance of some CAs and in studying their global properties. We, on the other hand, wanted mainly to observe which kind of individual behaviors emerge and the mechanisms by which the evolved systems shape individuals in order to achieve good collective results. We kept the search procedure as simple as possible, and neither adjusted it nor repeated the process extensively to find the best possible CAs. Thus, we did not expect to achieve results similar to those shown in [Table 1](#). Of particular interest to us is whether the shortcuts affect individual computations, creating asymmetries between cells in cellular automata.

To complete this research we had to choose the mechanism to be used to search for CAs with good performance, the encoding used by such mechanism for rules and networks, and the theoretical framework to understand the evolved behaviors. In the following subsections we will present all decisions made and techniques used.

Searching good CAs

The space we explored to find cellular automata able to achieve good performance in the density classification problem is overwhelming—in a CA with 149 cells each with 7 connections (the configuration used in this work), in which a ratio of only 0.01 of its connections are rewired, there are about $1.6 * 10^{70}$ different combinations of rules and topologies. Evaluating all possible CA was not an option! Thus, as many other researchers working with cellular automata, we decided to look for rules with good performance using a genetic algorithm. Beyond the difficulty of designing rules for cellular automata that perform specific computations, a good aspect of using automated search procedures in comparison to using predefined rules is to make us able to find non-intuitive—but suitable—rules about which we would not think by ourselves, showing us some alternative ways a complex system could explore the interactions between multiple agents to achieve collective computation.

Genetic algorithms (GA) [56] are optimization methods inspired by biological evolution. A solution is encoded as a string, known as chromosome, and at each epoch many solutions are evaluated. Those chromosomes with best fitness are preferentially selected for reproduction, producing the population for the next epoch. New chromosomes are created by the combination of genes coming from a pair of parents through crossover and, after that, a small number of randomly selected genes of these new solutions are mutated, aiming at introducing diversity in the population. This process is repeated until a termination criterion is reached (e.g., the maximum number of epochs is reached or the search process stopped finding better solutions). In this process, the better the gene the higher its chance of spreading throughout the population, so that when the execution is finished the population, which was randomly initialized, usually contains high-quality solutions for the problem being solved. It is worth noting, however, that being a metaheuristic, no genetic algorithm can guarantee to find the global optimal solution, whenever a finite amount of computational resources is employed.

In our study, the search process was repeated 11 times, each allowing the CAs to have a different number of rewired edges, starting with networks with 0.0 of rewiring, and increasing by 0.01 this ratio at each search, until the ratio of 0.1 was reached. This decision was taken to allow the evaluation of the impact of adding different numbers of shortcuts to the interaction structure of the CA. The maximum value of rewiring ratios was limited to 0.1 because, for a network similar to those used in this experiment, this is the range in which changes in rewiring ratio p have the highest impact on topological features, so that small-world characteristics are more prominent, as one can see in [Fig 5](#).

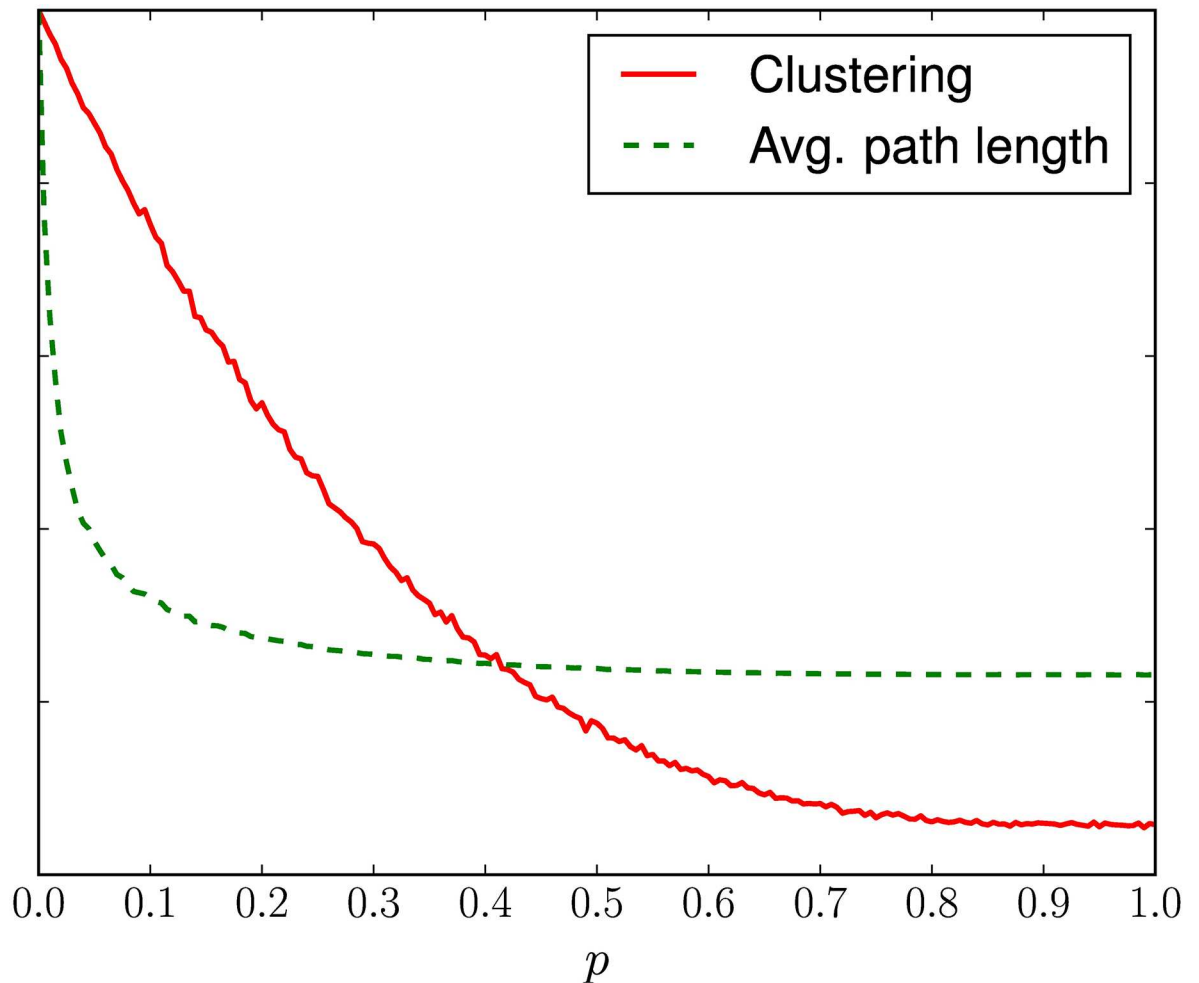


Fig 5. Relative average path length and clustering coefficient in networks generated with the small-world model, according to the rewiring probability p .

doi:10.1371/journal.pone.0172073.g005

The power of genetic algorithms resides in their ability to explore characteristics common to the structure of many problems, thus we need a proper way to encode solutions into a chromosome for an effective search. Some desirable features of such encoding are: the ability to represent virtually any possible solution of interest to the problem at hand; solutions generated through crossover and mutation must satisfy any existing constraint (e.g.: the in-degree of every cell must be equal to k); and solutions with representations close to each other have similar fitness (i.e., the fitness function is smooth). As we needed to represent two different aspects of the cellular automata, we opted for a chromosome composed of two disjoint parts, one representing the topology and the other representing the rule table used.

Rule table encoding and rule crossover. In an ECA, each possible set of states a cell and its neighbors may take can be encoded as arrays of zeros and ones [57]: if $k = 3$ and a cell is, currently, in state 0, while its left and right neighbors are, respectively, in states 1 and 0, this input set can be represented by the array 100, or the number 4 if we read this array as a binary number. This means we can list every possible input a cell can see and associate with each of them a unique number. In the case of $k = 3$ we would have $2^k = 8$ possible inputs, numbered

between 0 and 7. An even nicer possibility for ECAs is that if we list all output states of a given rule table, sorted by the number associated with its input set, we would have again a binary array and, thus, we could represent each possible rule table a cell can have by a unique number—a representation widely used in literature regarding elementary cellular automata. For instance, if we consider white cells as ones and black cells as zeros, the rule set adopted in Fig 2 would be represented by the array 10001111, and its number would be 143, while the number of the rule in Fig 3 would be 110 (array 01101110).

With the use of irregular networks comes the possibility of cells in a CA having different number of inputs. It would make necessary a different way to encode a rule table. Some researchers [46, 51, 58] solved this problem by adopting a simple rule in which a cell goes to state 1 if a ratio larger than a threshold τ of its neighbors are in state 1 and goes to 0 otherwise. However, to keep local characteristics of cells—like rules and number of inputs—as similar as possible, we forced every cell to have exactly $k = 7$ inputs, implying the existence of $2^{2^7} \approx 3.4 \times 10^{38}$ possible rule tables for CAs like those we explored in this work. This uniformity imposed that most of the behavioral differences comes from factors external to the cell (externalities) and also allowed us to borrow the same encoding used for ordinary ECAs, making it possible for us to explore more rules than a threshold-based rule would allow.

When generating a new individual, for each entry in the rule table we randomly selected one of the two parents from which we copied the associated output. After that, mutation was performed by flipping each bit with a small probability π .

Network encoding and topology crossover. As explained, we decided to explore small-world networks [39], but using rewires selected by an evolutionary method, instead of random rewires. In each CA we allowed a fixed number of edges to be reconnected, what means that a CA originally based on a ring network could replace some cells' original contacts for contacts located anywhere in the system, creating, thus, some shortcuts. Compared to previous analysis, which usually evolved only the amount of rewiring in the network—a parameter not evolvable in our experiments—but not which edge to reconnect, this approach has the advantage of making possible shortcuts to be more carefully placed in the network as well as the evolution of motifs and other control structures.

As a cell can be a source of information for a variable number of peers but can read the state from exactly $k = 7$ neighbors, all networks studied here were directed and edges could only be disconnected from their origins, not from their destinations.

In a CA's chromosome, the network was encoded by representing only its rewirings. For a given ratio p of rewirings, a network is represented by a list with $\lfloor p * n * k \rfloor$ triples, one indicating each reconnected edge. The first position of such triples stores the index $i \in [1, n]$ of the destination cell of this edge. The second position contains the index $j \in [-r, -1] \cup [1, r]$ of the input edge of the destination cell to be rewired. Here the j -th edge is the one that originally pointed from cell $i + j$ to cell i . Edge $j = 0$ represents a self-loop and is not allowed to be rewired, so a cell always considers its own current state when calculating the next state. Finally, the third position indicates the cell $l \in [1, n]$ which is the edge's new origin. It is relevant to notice that an edge could be rewired back to its previous origin, so that this encoding also allows the evolution to “cancel” rewirings, if it is found to be beneficial to CAs. We did not take any special measure to avoid the occurrence of different representations for isomorphic graphs, considering the computational costs involved in such operation.

Given a pair of parents, a new chromosome was generated by combining the lists of triples from the parents and selecting $\lfloor p * n * k \rfloor$ unique elements from it. Rewires that occurred in both the parents were twice as likely to be selected for the new individual. Mutation was performed by replacing with probability π each triple in the rewire list by a new one randomly selected among all possible rewires. No edge was allowed to be rewired twice. For a given set of

parameters p , n and k , these evolutionary operators always generate individuals with the same number of distinct rewirings, so that all solutions produced by crossover and mutation are feasible.

Fitness evaluation. Ideally, an individual’s fitness would be given by the proportion of all possible configurations such cellular automaton is able to correctly solve. This is a value extremely difficult to calculate, though, considering that the CAs we evolved have around $7.1 * 10^{44}$ different possible initial configurations and that, as far as we know, there is no closed method to perform such calculation directly from a CA’s encoding. To estimate an automaton fitness, following Lizier *et al.* [59] we tested it over $c = 4480$ randomly sampled initial configurations. Only tests in which all cells converged to the correct state were considered successful —i.e., a configuration in which more than 0.5 of the cells were initially in state 1 was considered as a failure even if only one cell was in state 0 at the end of the execution. Also following the common practice in literature [31, 34, 37], each CA was executed for $m = 2 * n + 1 = 299$ iterations and an execution was considered to converge only if all cells are in the same state during the last and the second-to-last iterations.

The most straightforward way to sample the initial configurations is by defining independently the state of each cell with uniform probability, which yields each possible initial configuration with the same probability. However, as this process gives rise to a binomial distribution of densities with probability $p = 0.5$ of cell being in state 1, most of the configurations generated using this model have densities near $\rho_0 = 0.5$, the hardest region for the DCT, as changes in initial states of few cells can invert the desired output. To provide an easier landscape to optimize, we used a different sampling mechanism to calculate a CA’s fitness for the evolutionary algorithm: we used a different probability, for each repetition, of initializing a cell with state 1. These probabilities were uniformly spaced in the range (0, 1), as suggested by Melanie Mitchell *et al.* [31].

Selection. At each epoch a complete new population was produced. For each of such new chromosomes, its parents were selected based on two different tournaments between 5 random individuals from the previous epoch. The candidate with highest fitness was considered the tournament’s winner and, thus, selected for reproduction.

Table 2 summarizes all configurations used in the experiment.

Table 2. Parameters and configurations used in the experiment.

Genetic algorithm	
Population	100
Number of epochs	50
Selection method	Tournament
Tournament size	5
Crossover type	Uniform
Mutation rate (π)	2%
Rewiring ratios	{0.0, 0.01, ..., 0.09, 0.1}
Cellular automata	
Size (n)	149
Radius (r)	3 ($k = 2 * r + 1 = 7$ neighbors)
Boundary	Periodic (ring)
Update	Synchronous
Iterations	$2 * n + 1 = 299$
# of repetitions	4480

doi:10.1371/journal.pone.0172073.t002

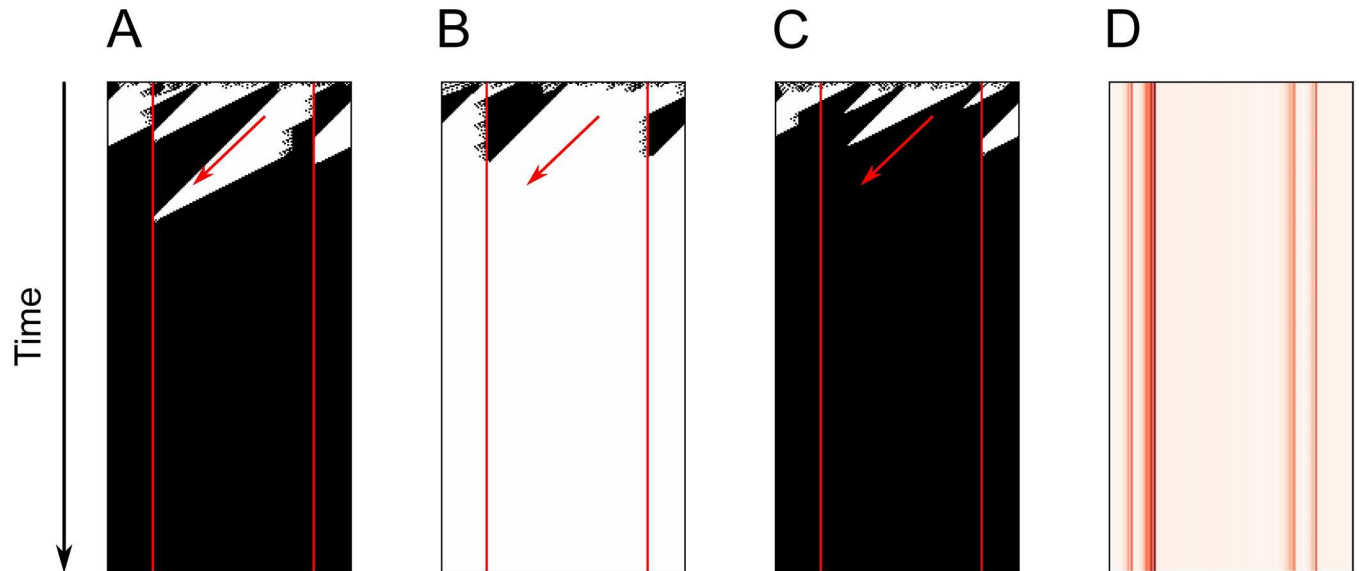


Fig 6. Space-time diagram of executions of the best cellular automaton found with $p = 0.01$ rewired connections, for three different initial configurations. In (A), (B) and (C), limits and flow directions are indicated, respectively, by the vertical lines and diagonal arrows. In (D), darker colors indicate that a cell acted more frequently as a limit.

doi:10.1371/journal.pone.0172073.g006

All simulations were performed using Python 2.7 with NumPy 1.8.2, SciPy 0.13.3 (for statistical calculations) and CUDA 6.5 (for GPU parallel computing) in Ubuntu Linux 14.04 running in an Intel Core i7 CPU (3.50GHz) and a GPU NVIDIA GeForce GTX 660 Ti. The source code for this experiments is available at https://github.com/unicamp-lbic/small_world_ca.

Understanding collective computation

After the examination of space-time diagrams depicting the execution of cellular automata evolved in an initial, exploratory step, we noticed that, unlike reported in works using fixed networks, computation in the evolved CAs was characterized by the occurrence of directed flows and limits—information flows almost freely in a specific direction until it encounters a limit, a cell that “decides” whether the flow will continue or will be interrupted. In the evolved CAs, the cells which act as limits are very stable over many different initial configurations (see Fig 6).

To understand such different behaviors, in the second phase of our work we evolved new CAs following the same setup used before, but evaluating topological conditions—as the occurrence of rewiring, degrees, clustering coefficient, distances and centrality measures—for the emergence of such limits and whether they were related to higher performance. To determine whether a cell acted as a limit in a CA, we estimated the direction and velocity that better approximate the flow of information in the CA and, then, counted in all repetitions how frequently a cell interrupted the flow of information, acting as a limit. For each CA, the estimated flow was given by the integer f that minimized the difference between the automaton states in iteration t rotated by $f \in [-r, r]$ positions and the automaton states in iteration $t + 1$. Given f , a cell i is considered to interrupt the flow of information in a given iteration if its state differs from the state of cell $i - f$ in the previous iteration. Fig 6 shows the limit measure calculated for a CA with ratio 0.01 of rewiring.

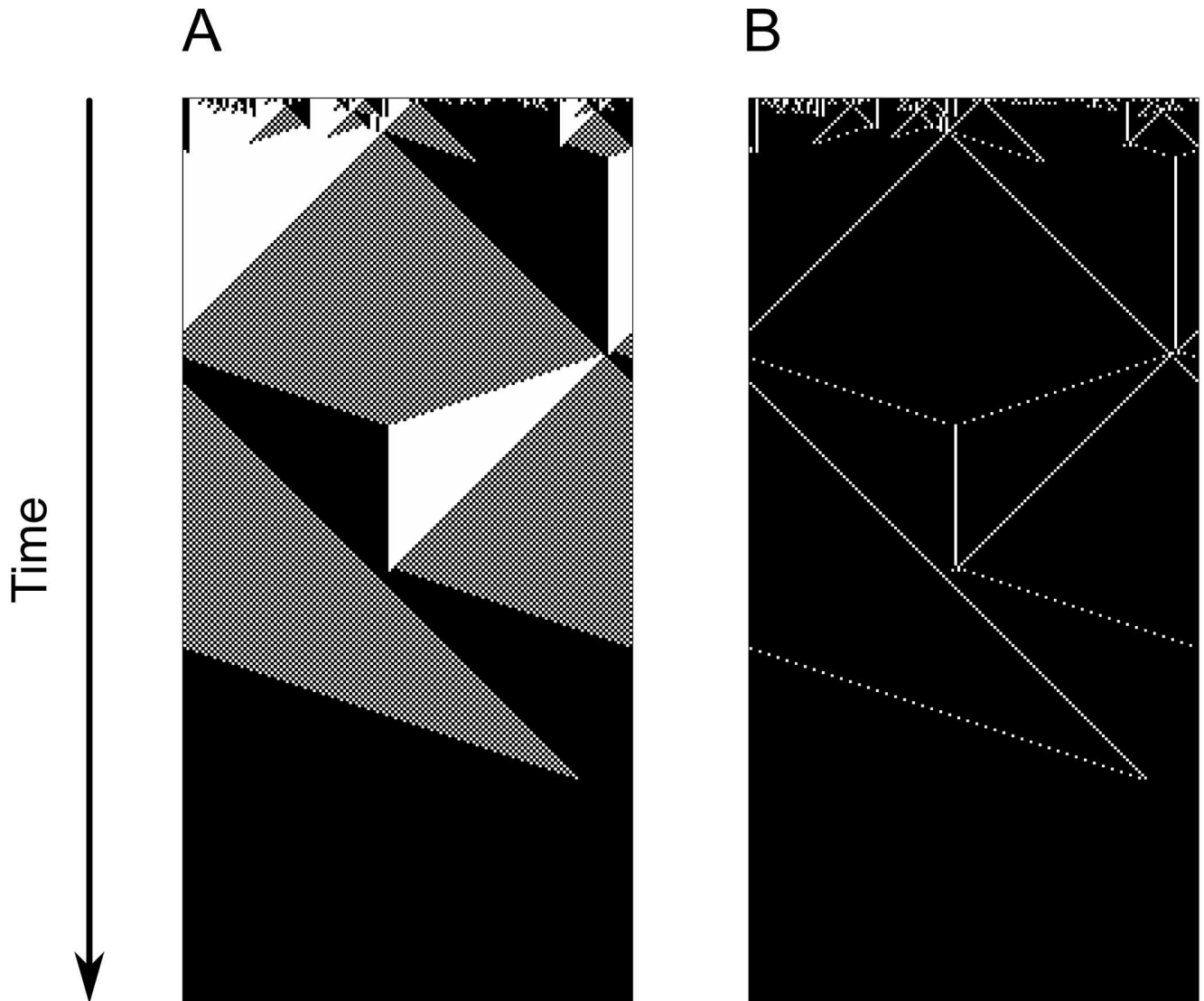


Fig 7. Space-time diagram for an execution of the GKL rule [35] (with regular ring topology) and particles observed during such execution.

doi:10.1371/journal.pone.0172073.g007

Previous works with uniform and regular ECAs have established different analytic frameworks to explain how computation happens in cellular automata. *Computational mechanics* [33, 49, 60] is an analogy with particle physics which explains computation in CAs by using particles as information carriers and collisions between particles as computational events. As the reader can see in Fig 7, particles are defined by boundaries between two different kinds of domains and they travel through the CA transporting information about local densities and sizes of high- and low-density regions. When particles collide, they can both annihilate each other or emit new particles, depending on the types of the colliding particles. However, despite being helpful to predict the behavior of a CA from its rule table and understand its errors [61], this approach is not suited to explain the behavior of irregular networks as the notion of space, central to computation mechanics, is destroyed in small-world networks with the addition of connections between cells not adjacent in the original ring.

Manuel Marques-Pita and Luis Rocha [62] proposed a second way to analyze cellular automata, known as *schema redescription*. The main idea is to eliminate redundancy in rule tables for cellular automata, leaving only input states that are relevant to define a cell's next state and grouping states in which the exact position of an input is irrelevant (the only information that matters in this group is the number of cells in state 0 or in state 1). After this redescription, the rule table keeps only the input states that effectively determine the automaton's transitions, which reduces the number of rules to analyze, making it easier to detect symmetries in a CA's behavior. Using this technique, Marques-Pita and Rocha were able to show that, despite showing different global behaviors, the hand-designed rule GKL [35] is extremely similar at local level to the rule GP, found by Andre, Bennet and Koza [34] using genetic programming.

Considering its ability to evaluate cells based on their observed behaviors and not only on their rule tables, our framework of choice is a set of information theoretic measures proposed by Joseph Lizier, Mikhail Prokopenko and Albert Zomaya [63, 64] to analyze local dynamics of information in distributed systems like cellular automata and random Boolean networks. They presented a set of measures intended to indicate how information is stored, transferred and modified in the system—operations required for universal computation.

The *stored information* that is currently used by a cell to calculate its next state is given by its *active information storage*. It is defined as the average mutual information between the semi-infinite past of cell i and its next state at time step $t + 1$:

$$A_i = \lim_{j \rightarrow \infty} \left\langle \log_2 \frac{p(x_{i,t}, x_{i,t+1})}{p(x_{i,t}^{(j)})p(x_{i,t+1})} \right\rangle, \tag{1}$$

in which $x_{i,t}$ is the state of the cell i at time t and $x_{i,t}^{(j)}$ is the j past states of cell i , from time $t - j + 1$ to time t . Considering that it is not feasible to compute A_i in the limit $j \rightarrow \infty$, the approximation $A_i(j)$ with finite history length j is used. A_i can be either positive or negative indicating, respectively, if the cell's current state is relatively likely or unlikely, considering its past. The reader can also notice that A_i can assume high values even if the cell i does not have a connection to itself through a self-loop, as it can store information in a distributed manner, using its neighbors.

Different measures have been proposed to evaluate different aspects of the *information transfer* to a cell from its neighbors. We decide to use the *local collective transfer entropy*, which captures the amount of information transferred from all causal sources that were not already contained in the cell's past. In deterministic systems, like CAs, its value is equal to the *local entropy rate* [64], which can be calculated as:

$$H_{\mu i} = \lim_{j \rightarrow \infty} H_{x_{i,t+1}|x_{i,t}^{(j)}}, \tag{2}$$

in which $H_{x_{i,t+1}|x_{i,t}^{(j)}}$ is cell i 's *conditional entropy*, the uncertainty about the cell next state when its j previous states are known:

$$H_{x_{i,t+1}|x_{i,t}^{(j)}} = - \sum_{x_{i,t+1}, x_{i,t}^{(j)}} p(x_{i,t+1}, x_{i,t}^{(j)}) \log_2 p(x_{i,t+1}|x_{i,t}^{(j)}). \tag{3}$$

Again, as the computation of the limit for $j \rightarrow \infty$ is not feasible, a finite j is used to calculate the local collective transfer entropy.

The local entropy for a cell is the sum of the local active information and the local entropy rate:

$$H_i = A_i + H_{\mu i}, \quad (4)$$

a relation preserved even for approximations with finite j .

To measure *information modification*, the framework proposes a metric named *local separable information* which, however, can be considered more a heuristic, as it double-counts parts of the information in the next state of the destination. A proper metric is still under development [65] and, therefore, local separable information is not used in our work.

Lizier and colleagues applied their framework to study information flow in cellular automata and random Boolean networks [63, 65], quantitatively verifying, for instance, the conjectures about roles of particles and collisions in the computational mechanics framework [63]. They also evaluated information dynamics in random Boolean networks [59], discovering that, while regular and random topologies are usually dominated, respectively, by information storage and information transfer, RBNs based on small-world networks balance both storage and transfer capacity.

Results

In Fig 8 we show, for each different rewiring probability p , the evolution of the distribution of fitness in the population during the search. The first thing the reader can notice is that, similarly to results seen in the literature [47], the evolution was faster in finding multiple good solutions when we allowed the topology to be rewired: in searches with $p > 0$ we observed a general improvement of fitness around the 4th epoch, while for fixed topologies such improvement was seen only after 8 epochs. It is worth noticing that all searches were successful in finding rules able to solve more than 90% of the initial configurations, which were sampled with densities evenly distributed in the range $\rho_0 \in (0, 1)$. We also show in Fig 8 the performance of the best individual at each epoch, testing them in initial configurations sampled with density $\rho_0 \approx 0.5$, the most common and, also, most difficult situation in the DCT. Again, all the best results achieved when allowing rewiring—respectively, 72%, 71%, 71%, 69%, 71%, 67%, 74%, 72%, 71% and 73%—were better than the obtained with fixed topology—when 66% of all initializations were correctly solved.

We exhibit in Fig 9 space-time diagrams with examples of executions of the best individuals found for each different p . When $p = 0$ the observed behavior is very similar to that of particles from computational mechanics, though we did not achieve rules so refined as those obtained by Melanie Mitchell and colleagues [31], with many different domains and particles. We remember, however, that in their work, from hundreds of executions, only few searches were able to find such rules with high performance. As more edges are allowed to be rewired, we can observe the loss of spatial coherence.

At the top of each space-time diagram, we also plotted a bar graph indicating how often a cell acted as a limit. Even though such measure is only a heuristic, in these graphs it is possible to see that, in each CA, there are groups with few cells that act as limit at least twice as often than the others, a fact even more prominent for smaller p . This is evidenced in Fig 10, which displays the evolution of inequality in the role performed by the cells during evolution, measured with the Gini coefficient [66]. Gini coefficient is a summary statistic that measure the inequality in a population of non-negative reals, assuming values between zero (when all individuals are equal) and one (when only one individual has value different from 0). We can notice that, consistently across all executions with $p > 0$, as the population's fitness increases fewer cells will act as limits (apparently) “deciding” the fate of information flows. At the final

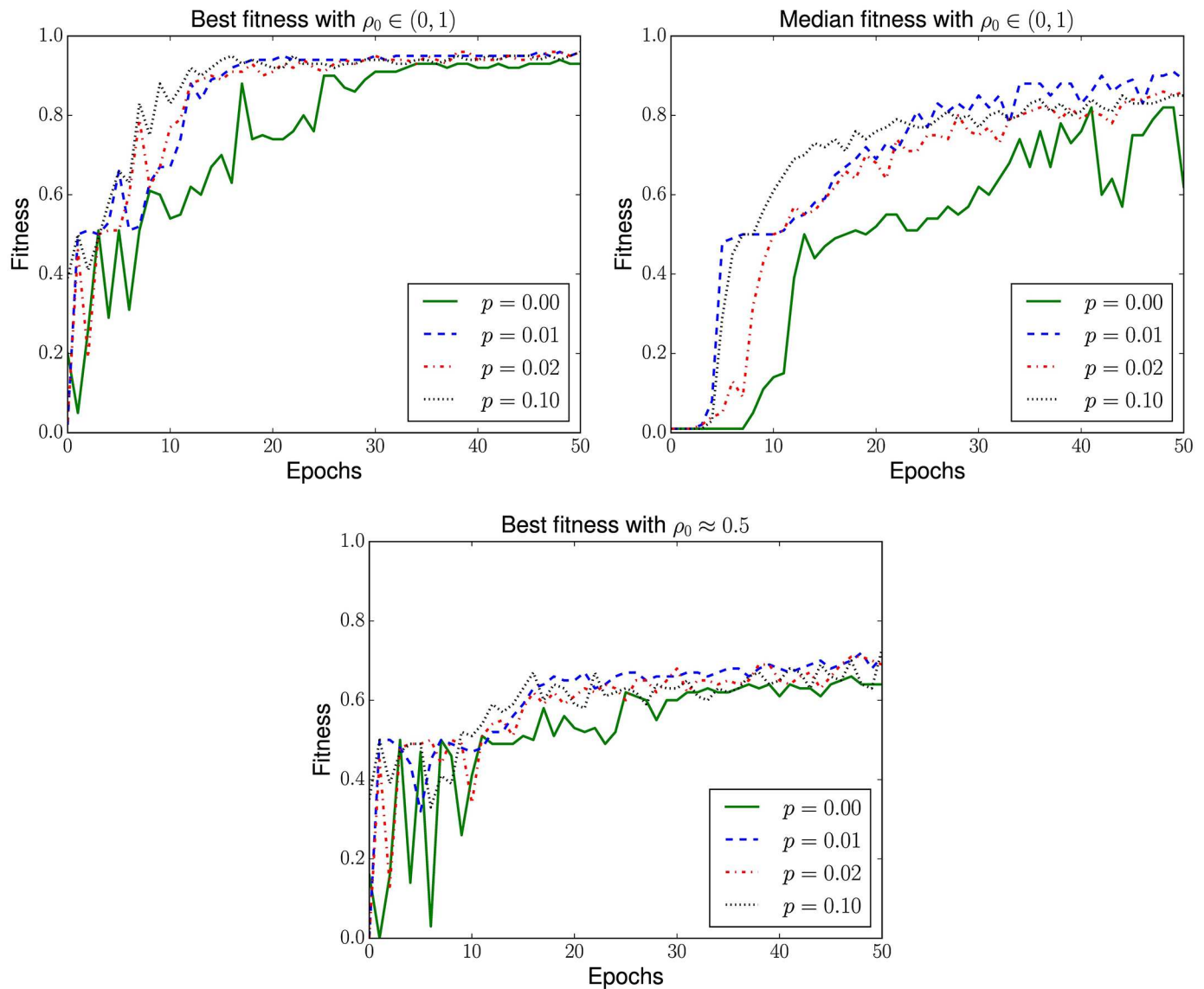


Fig 8. Evolution of fitness during the searches with different rewiring probabilities p . The plots indicate, for each epoch, the best and the median fitness of the population, calculated with initial configurations with densities evenly distributed in the range $\rho_0 \in (0, 1)$. We also show the results achieved by the best individuals in each epoch when initial configurations are sampled with density $\rho_0 \approx 0.5$.

doi:10.1371/journal.pone.0172073.g008

epochs of the evolutionary search, such most decisive cells act, usually, 4 to 10 times more often as limits than the least decisive ones (see Fig 11).

Inequality, topology and fitness in cellular automata

In Table 3, we can see how some general metrics regarding the cellular automata's efficacy and structure correlate with the inequality in behavior among cells. In addition to clustering coefficient and average path length, we also indicate inequality in out-degrees (the number of edges originated in each node) and in closeness centrality (a measure that indicates how close a given node is to all other nodes in the network). The first point to notice is the high correlation

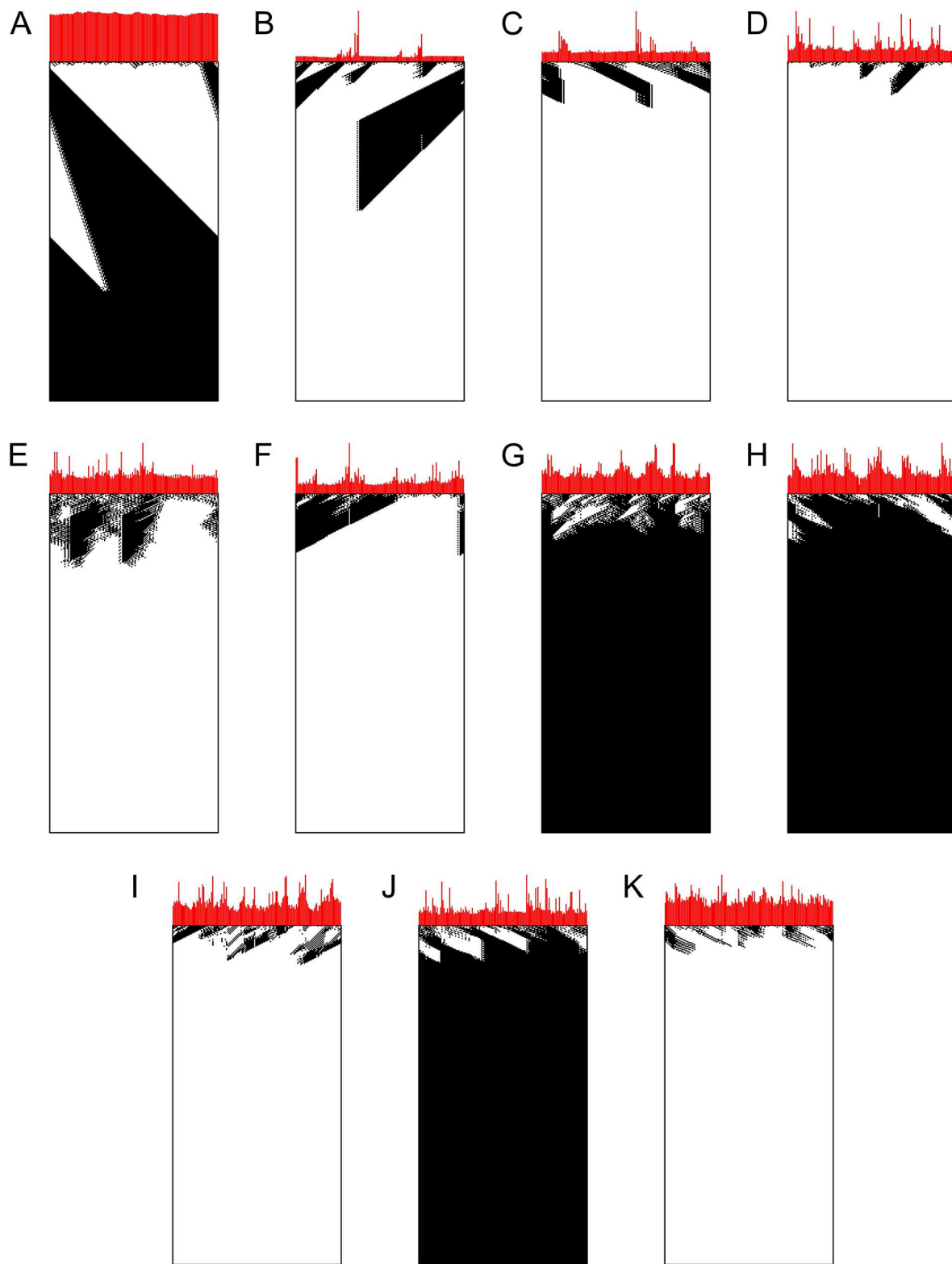


Fig 9. Space-time diagram of executions of the best cellular automaton found at each execution. Above each space-time diagram is depicted a bar graph indicating how often each cell acted as a limit. To improve visualization the bar graphs were scaled so the maximum height is equal across all CAs, so that bar heights should be compared solely within each graph. **(A)** $p = 0.00$, **(B)** $p = 0.01$, **(C)** $p = 0.02$, **(D)** $p = 0.03$, **(E)** $p = 0.04$, **(F)** $p = 0.05$, **(G)** $p = 0.06$, **(H)** $p = 0.07$, **(I)** $p = 0.08$, **(J)** $p = 0.09$ and **(K)** $p = 0.10$.

doi:10.1371/journal.pone.0172073.g009

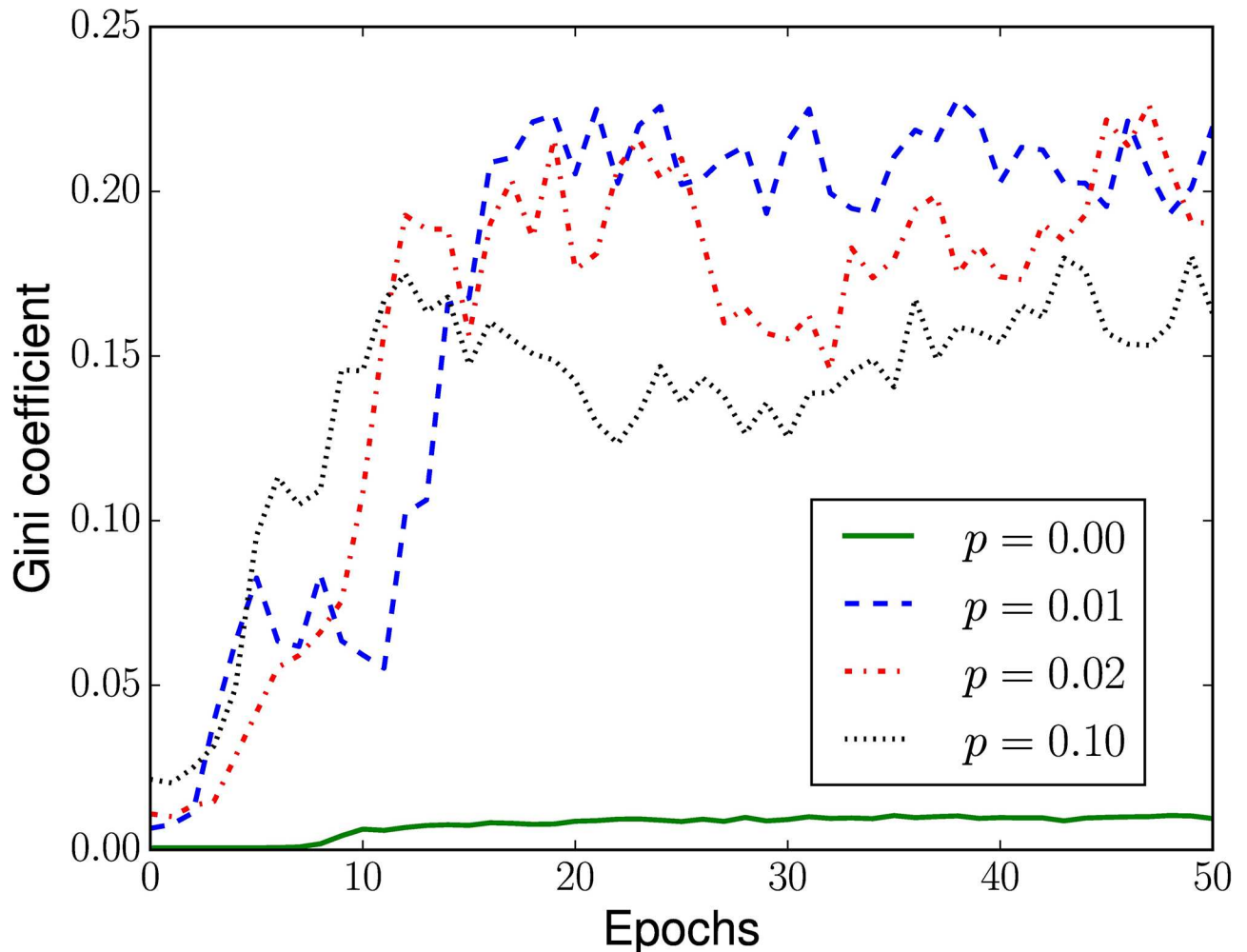


Fig 10. Evolution of the Gini coefficient. Evolution of median inequality of frequencies that cells in each CA acted as limits during the search with different rewiring probabilities p .

doi:10.1371/journal.pone.0172073.g010

between inequality and a CA's fitness, agreeing with what was seen in Figs 8 and 10. For every search, such correlation was stronger than that between fitness and average path length ($\max|\rho| = 0.302$), which may indicate that, in the evolved CAs, the presence of limits was more relevant for a successful automaton than a small average distance between cells. Also, in almost every execution, there was statistically significant indication of a weak negative correlation between inequality in the behavior of the cells and both average clustering coefficient and average path length. Such results indicate that CAs where the occurrence of limits was concentrated in fewer cells also favored rewires in which the new source of information was not in the neighborhood of any other source of the destination cell (i.e., the rewired connection brings new information to such cell). Regarding inequality in out-degrees, also measured using the Gini coefficient, in 7 of 10 executions we can see a weak negative correlation with concentration of limits in fewer cells. This may indicate that inequality in the behavior of the cells is associated with a lower presence of hubs that broadcast information to many cells; however, more research is needed to confirm such hypothesis. We highlight that most of the correlations

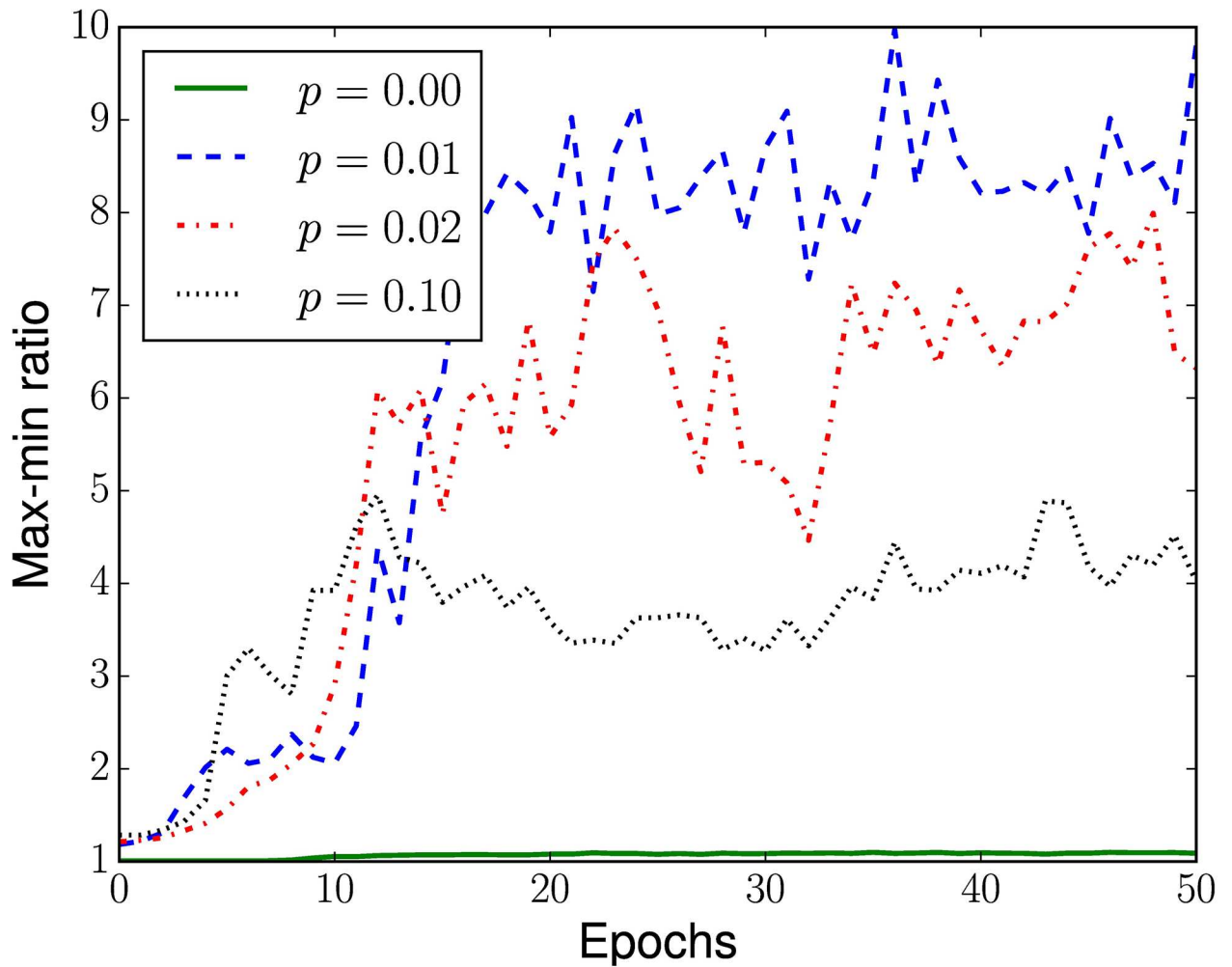


Fig 11. Evolution of the ratio between the maximum and minimum frequencies a cell acted as a limit. Evolution of median ratio between the maximum and minimum frequencies that cells in each CA acted as limits during the search with different rewiring probabilities p .

doi:10.1371/journal.pone.0172073.g011

Table 3. Spearman’s rank correlation between the inequality of limits distribution and CA’s metrics. Statistically significant correlations ($p_{value} < 10^{-5}$) are marked with ‘*’.

Rewire (p)	Fitness	Clustering	Avg. Path Length	Out-degree (Gini)	Closeness (Gini)
0.00	0.609*	–	–	–	–
0.01	0.512*	0.157*	0.064*	0.015	0.130*
0.02	0.555*	-0.073*	-0.143*	0.058	-0.034
0.03	0.537*	-0.079*	-0.066*	-0.144*	-0.042
0.04	0.531*	-0.106*	-0.072*	0.071*	-0.075*
0.05	0.549*	-0.134*	-0.173*	-0.162*	-0.049
0.06	0.414*	-0.035	0.006	-0.007	0.026
0.07	0.539*	-0.172*	-0.155*	-0.083*	-0.014
0.08	0.569*	-0.158*	-0.110*	-0.217*	0.027
0.09	0.554*	-0.060	-0.047	-0.118*	0.064*
0.10	0.419*	-0.017	-0.008	-0.008	0.014

doi:10.1371/journal.pone.0172073.t003

discussed before were not seen in the execution for $p = 0.01$. We conjecture that the evolutionary process took a different and (apparently) less likely path during this search. It is important to have in mind that most of the observed correlations are weak and that the experimental design does not allow us to look for causal relations between the metrics, so that we cannot confirm or rule out whether these associations are mediated only by the correlation to a third variable (fitness, for instance).

The results for $p = 0$ need a more careful explanation: in such situation, despite all cells having the same pattern of connections and following the same rule table—which makes inequality in behavior virtually impossible—we see a strong correlation between behavior inequality in a CA and its fitness. This is an artifact of the heuristic used to detect limits in the space-time diagram, as it also identifies as limits many domain boundaries in the computational mechanics paradigm. However, as the reader can see in Fig 10, the Gini coefficient when $p = 0$ is much smaller than the values observed when $p > 0$. Indeed, this correlation is a spurious result from the finite number c of repetitions used to estimate fitness and behaviors and it fades away as $c \rightarrow \infty$.

Cells' roles and their position within the network

Now, it is important to understand whether such limits, in fact, play a relevant computational role in the evolved cellular automata. For this, we used the information theoretical metrics presented in the previous section and evaluated their Spearman's rank correlation with the value indicated by the limit heuristic. For each different CA, the information theoretical metrics were calculated for each cell based on their behaviors in all of the 4480 different initial configurations tested. We observed a strong correlation between the frequency a cell acts as a limit and its local entropy rate ($\rho = 0.691$, $p_{value} \ll 10^{-10}$), while the limit frequency is negatively correlated with the active information storage ($\rho = -0.473$, $p \ll 10^{-10}$). Such results agree with our initial hypothesis that cells that work more as limits have a different role when compared to the other cells. The high correlation with local entropy indicates that the limits are important for the integration of information from different places, while the other cells function more as the cellular automaton's memory.

Let us then investigate possible reasons for this difference in behavior between cells. The first thing we will do is to evaluate the Spearman's rank correlation between the value indicated by our limit heuristic and structural metrics about the underlying topology of the CA (for $p > 0$). For this, we used basic local metrics—a cell's in- and out-degrees, its local clustering coefficient, the average out-degree of its neighbors and the number of rewired input edges—and also classical centrality measures [67, 68], that are affected by a cell's position in relation to the whole network. The centrality measures used are:

- PageRank, HITS and eigenvalue centrality, which, in our design, measure how much information can flow to each node. PageRank and eigenvalue centrality are computed using the recursive assumption that the information a node can gather is proportional to the information gathered by this node's neighbors. HITS (Hypertext Induced Topic Search), in turn, provides two different values for each node ("authority" and "hub" scores) indicating respectively how influenced and how influential is a node. The basic (recursive) idea in HITS is that nodes with more "authority" are pointed by good "hubs" and that good "hubs" point to nodes with more "authority".
- Closeness centrality considering both the original and the reversed network, to evaluate respectively how fast information flows from and to the node.

- Load and betweenness, that indicate the fraction of shortest paths in the network that pass through each node, measuring how central a node is for efficient connections in the network.
- Eccentricity, i.e., the node's maximum distance to all other nodes.

Despite all correlations achieving statistical significance ($p_{value} \ll 10^{-10}$), we did not observe any strong correlation between the measures and the frequency a cell acts as a limit. However, we saw a weak correlation ($|\rho| \approx 0.1$) of limits with the number of rewired inputs ($\rho = 0.141$) and with a node's closeness centrality in the reversed network ($\rho = 0.130$). We also observed a negative correlation with clustering coefficient ($\rho = -0.113$), which is expected as the rewiring process tends to reduce local clustering coefficient.

We also looked at this relationship between a cell's behavior and its position within the CA's topology from a different perspective: instead of considering all cells in all CAs to calculate the correlations between the frequency a cell is a limit and its structural metrics, we computed such correlations for each CA separately (for $p > 0$). Our goal is to better understand the factors that explain why, in each specific CA, some cells act regularly as limits and others do not. Fig 12 shows histograms depicting the distribution of correlations we observed. As we can see, the number of rewired inputs and the closeness centrality in the reverse graph have a moderate median correlation with limits ($\tilde{\rho} = 0.327$ and $\tilde{\rho} = 0.326$, respectively), agreeing with our previous observations. As closeness in the reverse graph and number of rewires are strongly correlated ($\tilde{\rho} = 0.530$), we evaluated Spearman's partial rank correlation between frequency as limit and closeness in the reverse graph, controlling for rewired inputs, obtaining a relevant residual effect ($\tilde{\rho} = 0.187$). Weaker correlations are also observed with betweenness ($\tilde{\rho} = 0.197$) and load ($\tilde{\rho} = 0.196$) centralities. Such results suggests that cells that act more frequently as limits are usually those able to gather information from different regions of the automaton more quickly, being relevant not only the number of rewired inputs a cell has, but also the regions accessed by the cell through such links. As a side effect of the rewiring process, negative correlations are observed for clustering coefficient ($\tilde{\rho} = -0.216$) and average neighbors out-degree ($\tilde{\rho} = -0.186$).

How edges are reconnected

After looking at the cellular automata from macroscopic (focusing CAs' general metrics) and microscopic perspectives (analyzing cells' characteristics), we will assume an intermediary point of view, analyzing local patterns of connections between cells (motifs) associated with CAs with high fitness. Using the ESU algorithm [69], which enumerates all subgraphs with size k , we counted how many times each pattern of connections with $k = 3$ and $k = 4$ cells occurred in the best CA found in each execution with $p > 0$. Subgraphs in the same isomorphism classes were grouped. We used only one graph per search to avoid counts to be artificially inflated by the dependency between solutions in an execution of an evolutionary search, as these methods work by replicating blocks found in the previous epoch, usually (but not exclusively) associated with good fitness. Therefore, even a harmful motif, found in the initial population or generated by mutation, could be present in many CAs of a progeny (and, thus, identified as a motif) before the evolutionary pressure eliminated it.

For each selected graph, we generated an exact copy, but with random rewires, which we used as a baseline to check whether a subgraph is a motif or an *anti-motif* (i.e., it appears significantly less than the expected). Such evaluation was made using a paired t-test for each subgraph, comparing the number of occurrences in the original graph and in the baseline. In Fig 13 we exhibit all the subgraphs of size 3 or 4 whose number of occurrences in the high fitness

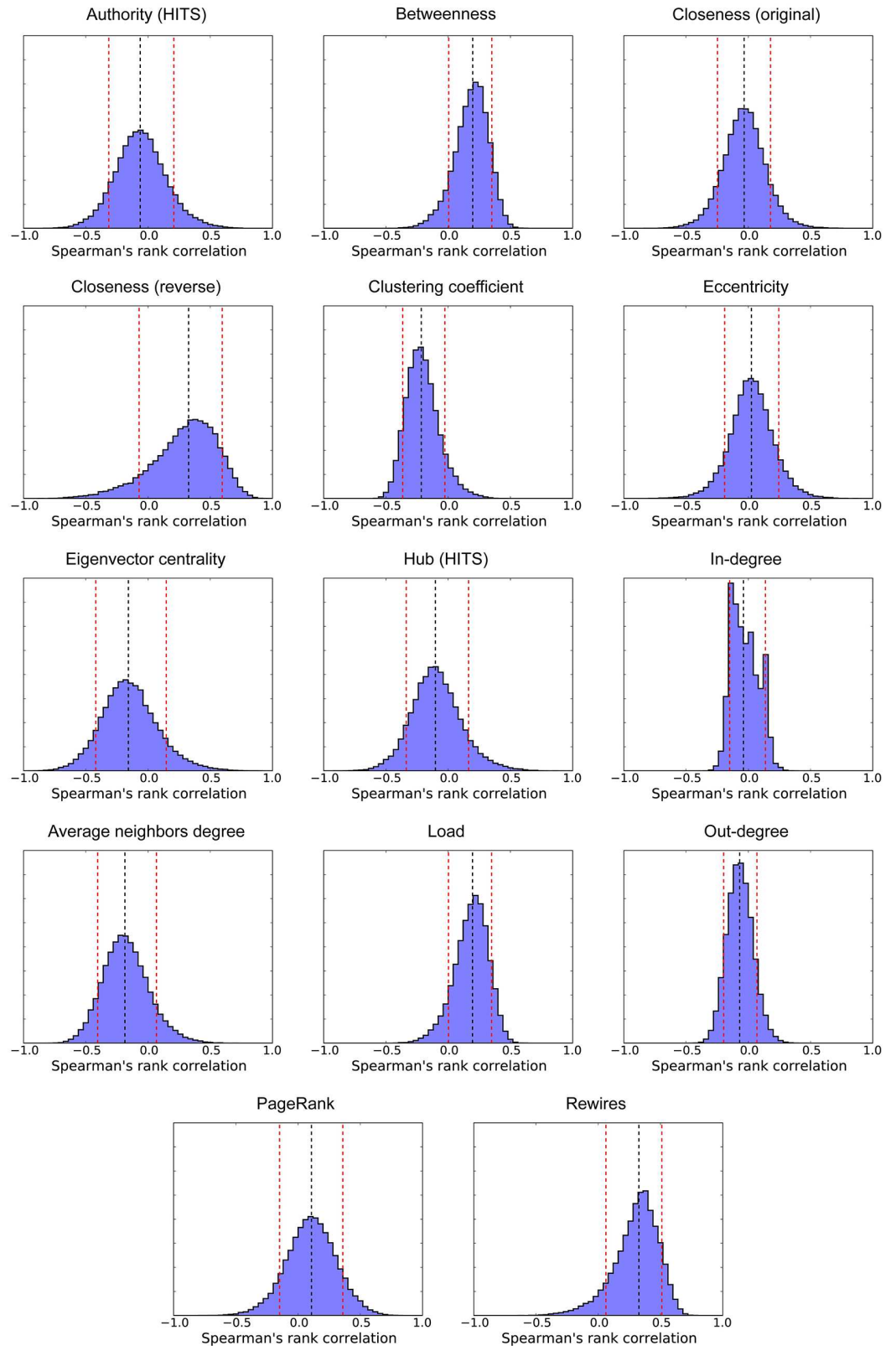


Fig 12. Distribution of spearman's rank correlation, calculated for each CA, between the frequency a cell acts as a limit and the cell's structural metrics. The black vertical line indicates median value and the red vertical lines indicate, respectively, the 10th and the 90th percentiles ($N = 51000$).

doi:10.1371/journal.pone.0172073.g012

graphs was significantly different from the occurrences in the randomized graphs ($p_{value} < 10^{-2}$). We call attention to the fact that we lowered the bar for statistical significance in this analysis considering the low number of graphs analyzed (only 10 for each motif). Even though we raised the significance threshold, we point that the probability of false positives producing results similar to those reported in this work, with at least 3 false positives in 13 comparisons (for motifs of size 3) and 11 false positives in 199 comparisons (for motifs of size 4) with $p_{value} \leq 10^{-2}$, is of only $2.653 * 10^{-4}$ and $6.563 * 10^{-6}$, respectively (assuming independent comparisons).

From the motifs depicted in Fig 13, we can infer the occurrence of four different general patterns of reconnections in cellular automata with high fitness:

- Long range rewires: motifs A, C, D, E, F and H indicate that rewires creating edges between nodes that were not previously in the same neighborhood are more likely in cellular automata with higher fitness.
- Multiple rewires per cell: motif B can be created either by the addition of edges from the central node or the removal of edges directed to it. Considering the absence of any motif of size 4 corroborating to the additive hypothesis and also motifs I and M, the most likely process is the rewiring of at least two of the central node edges originally coming from nodes that had no direct connection between each other.
- Centralization of rewires in a neighborhood: the observed anti-motifs J, L and N share one feature—all of them are likely to be generated by at least two different nodes that were directly connected in the original topology rewiring some of their input edges. Such situation may be the result of benefits obtained by a CA when it centralizes in only one node per neighborhood the task of integrating information arriving from different places of the CA.
- There and back again: motif K is likely to be created by the addition of a size 2 path by which the information coming from a neighborhood flows to a distant node, where it is integrated with information from other parts of the CA, and then it flows back to its original neighborhood.

We highlight that all the patterns observed in this motif analysis are in line with the correlations pointed in the previous subsections, as both analyses point to the importance of rewires to bring information from other regions of the CA and to the concentration of multiple rewires towards few cells, which integrate information from different places.

Discussion

We observed in the experiments performed during this study that, unlike previous works that used traditional elementary cellular automata with ring topology, when making the topology more flexible different classes of cells emerge with qualitatively different computational behaviors. In traditional ECAs that solve the density classification task, the movement of information structures—particles and domains—and their collisions are the main mechanisms for information transmission and modification, so that any type of computational process can happen anywhere in the CA. On the other hand, in cellular automata that allow edges to be rewired we saw that, usually, information flows in specific directions, and decisions about interrupting or allowing the propagation of such flows are made by specific cells, which we named limits. It is remarkable that even systems with individuals as simple as the cells in ECAs can achieve labor division—and obtain the benefits that come with it—only by creating and removing few connections between cells.

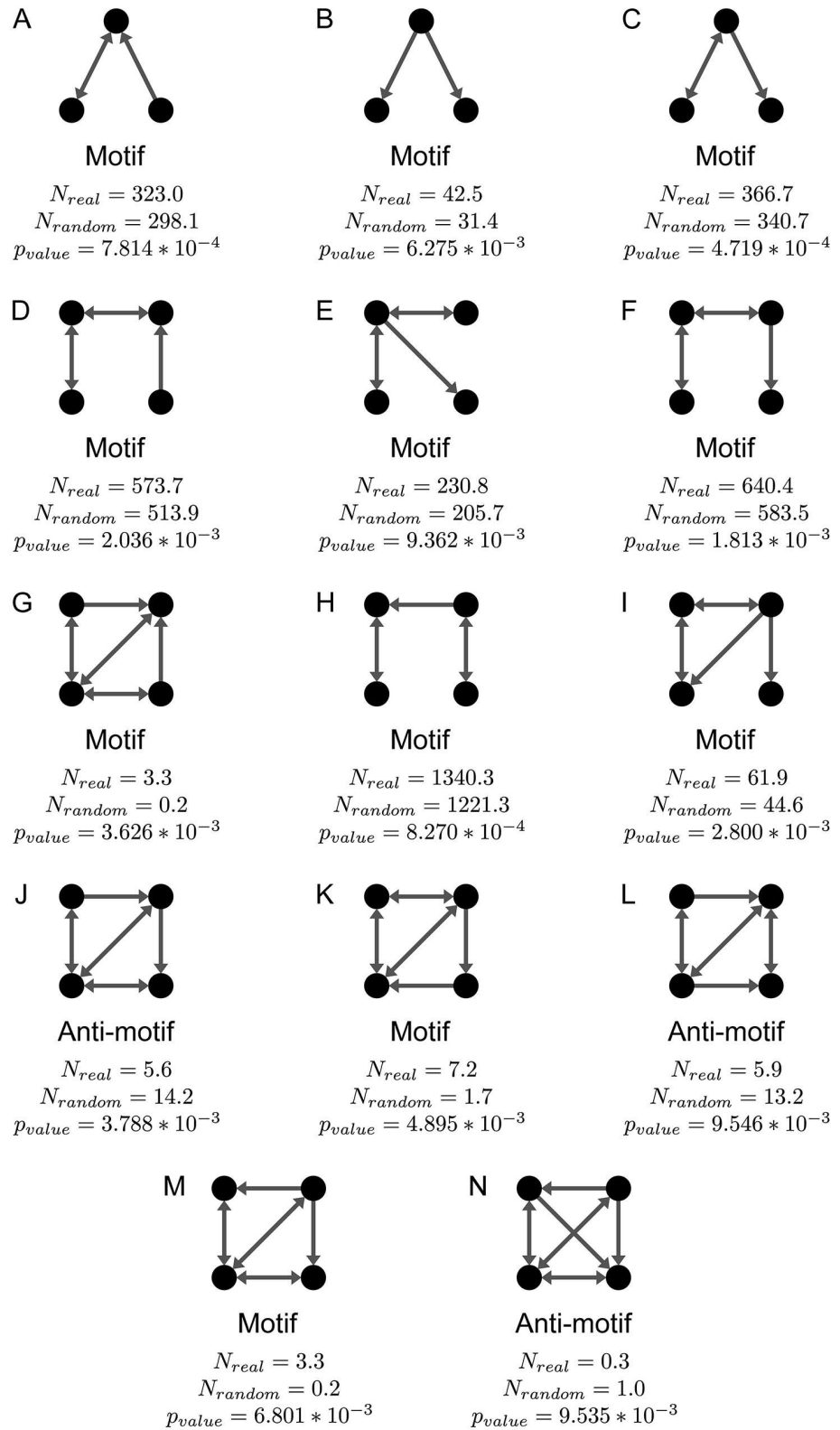


Fig 13. Motifs identified in the best topologies found in each search. N_{real} is the average count of the respective subgraph in each network. N_{random} is the average number of occurrences of such subgraph in similar networks, but with random rewires.

doi:10.1371/journal.pone.0172073.g013

Differently from most works in emergent computation [62], we focused not on general characteristics of the systems under analysis, but on understanding how local interactions can introduce specialization and, thus, produce the desired collective computation. After analyzing the behavior of limits according to information theoretical measures, we saw that they have the role of integrating information from multiple regions of the cellular automaton while other cells are, mainly, devoted to store information. This is a very interesting result from the point of view of network science, considering that all cells follow the exact same rules and have the exact same number of inputs, so any difference in behavior that is persistent over multiple initializations of a CA can only be attributed to differences among the positions these cells occupy in the CA's underlying structure. The observed specialization is also curious, as the evolutionary algorithm found, with no guidance, a method to overcome the lack of separation between memory and processing units of cellular automata, partially mimicking the separation between storage and data processing seen in systems as von Neumann computers.

Our analyses were not conclusive about which network characteristics contribute for the emergence of specialization in a cellular automaton. We were able, however, to gather coherent evidence pointing to some patterns and properties that are likely to have an important role in defining whether a CA will show limits and, also, which cells will act as limits. Considering the negative correlation of inequality with average distance between cells and clustering coefficient and, also, the patterns of connections in motifs found in CAs with high fitness, the presence of long-distance rewires seems to be a relevant factor for the occurrence of stronger limits. Limits are usually cells that have the ability to quickly gather signals coming from different regions—what is reflected in a positive correlation of limits with closeness centrality in the reverse graph—and, thus, tend to have many of their input edges rewired to distant cells and to centralize all rewires in their neighborhoods.

These results are in tune with those reported by Lizier and colleagues [59] about computational capabilities of Boolean networks based on small-world, regular and random topologies. They observed a strong correlation between networks with high clustering coefficient and information storage and a negative correlation between information transfer and average distances in a network, which points to the importance of long links for information transfer in a decentralized system. Small-world networks, according to their analysis, display a balance between information storage and transfer, which may render them suitable for complex computation. Our results indicate that these computational capabilities are not uniformly distributed across the network, but rely on network-induced specialization. Combining this capacity of interaction networks to affect global computation and individual behaviors with other results from network science [39, 48, 70, 71], we can wonder whether networks structuring many complex systems have some similar properties not only due to coincidence or to physical restrictions, but that they were positively selected by evolution or learning to do (at least) part of the computation performed collectively. There is already evidence of this evolutionary design of social networks in humans, for instance, as researchers have observed direct influence of some genes on the propensity of two individuals developing a long-term relationship [72].

While in neuroscience the idea of heterogeneity emerging from differences in connections is not new, they bring a new perspective when applied to other complex systems, particularly to social networks (both involving humans or other animals). Considering the variety and intricacy of individual behaviors, when analyzing social systems scientists usually think that complexity can prescind network structure or regard it to a secondary role in relation to individuals. However, as the difference of position in a network can partially explain the existence of individual diversity even in systems as simple as cellular automata, a proper understanding of collective behaviors may require consideration of the structure of interactions between agents, even when evidence indicates that individual behaviors are the sole responsible for a

computation. This focus on social relations echoes many ideas from distributed cognition [9, 73], which argues that mental processes are not restricted to single human brains but extend also to interactions with other people and the environment.

Finally, as the correlations we observed between behavior and network properties are either weak or moderate, a deeper study is necessary in the future in order to investigate other requisites for the emergence of inequality and specialization in elementary cellular automata with non-uniform topology. Despite not being able to generate limits when there is no heterogeneity in topology, a CA's rule table is also crucial for the existence of different classes of cells—as a trivial example, a rule in which a cell only repeats its previous state would never allow the emergence of limits, no matter the underlying network used. So, in future works it is relevant to better explore whether different types of specialization may emerge when different rules are applied and how sensitive are cells' computational behaviors to changes in the rules they follow. In order to advance the study reported here towards more complex systems, particularly social networks, it is also important to investigate how collective computation and the emergence of diverse behaviors is impacted by richer underlying networks—e.g.: networks that include more than one type of interaction, allow the exchange of complex signals, and support relations with different intensities.

Author Contributions

Conceptualization: AG PT.

Data curation: AG.

Formal analysis: AG.

Investigation: AG FJVZ.

Methodology: AG.

Resources: AG FJVZ.

Software: AG PT.

Supervision: FJVZ.

Validation: AG.

Visualization: AG.

Writing – original draft: AG.

Writing – review & editing: AG PT FJVZ.

References

1. Orwell G. *Animal Farm: a fairy story*. Secker & Warburg; 1945.
2. Seung S. *Connectome: How the brain's wiring makes us who we are*. A Mariner Book. Houghton Mifflin Harcourt; 2012.
3. Preston SD, de Waal FBM. Empathy: Its ultimate and proximate bases. *Behavioral and Brain Sciences*. 2002; 25:1–20. PMID: [12625087](#)
4. Quiroga RQ, Reddy L, Kreiman G, Koch C, Fried I. Invariant visual representation by single neurons in the human brain. *Nature*. 2005; 435(7045):1102–1107. doi: [10.1038/nature03687](#) PMID: [15973409](#)
5. Page SE. *The difference: How the power of diversity creates better groups, firms, schools, and societies*. Princeton University Press; 2007.
6. Miller JH, Page SE. *Complex adaptive systems: An introduction to computational models of social life*. Princeton University Press; 2009.

7. Kennedy J. Thinking is social: Experiments with the adaptive culture model. *Journal of Conflict Resolution*. 1998; 42(1):56–76. doi: [10.1177/0022002798042001003](https://doi.org/10.1177/0022002798042001003)
8. Mitchell M. *Complexity: A guided tour*. Oxford University Press; 2009.
9. Hutchins E. Organizing work by adaptation. *Organization Science*. 1991; 2(1):14–39. doi: [10.1287/orsc.2.1.14](https://doi.org/10.1287/orsc.2.1.14)
10. Levine JM, Resnick LB, Higgins ET. Social Foundations of Cognition. *Annual Review of Psychology*. 1993; 44(1):585–612. doi: [10.1146/annurev.ps.44.020193.003101](https://doi.org/10.1146/annurev.ps.44.020193.003101) PMID: [8434897](https://pubmed.ncbi.nlm.nih.gov/8434897/)
11. Baronchelli A, Ferrer i Cancho R, Pastor-Satorras R, Chater N, Christiansen MH. Networks in cognitive science. *Trends in Cognitive Sciences*. 2013; 17:348–360. doi: [10.1016/j.tics.2013.04.010](https://doi.org/10.1016/j.tics.2013.04.010) PMID: [23726319](https://pubmed.ncbi.nlm.nih.gov/23726319/)
12. Blum C, Merkle D. *Swarm intelligence: Introduction and applications*. Springer; 2008.
13. Franks NR, Pratt SC, Mallon EB, Britton NF, Sumpter DJT. Information flow, opinion polling and collective intelligence in house-hunting social insects. *Philosophical Transactions of the Royal Society of London Series B: Biological Sciences*. 2002; 357(1427):1567–1583. doi: [10.1098/rstb.2002.1066](https://doi.org/10.1098/rstb.2002.1066) PMID: [12495514](https://pubmed.ncbi.nlm.nih.gov/12495514/)
14. Pratt SC. Efficiency and regulation of recruitment during colony emigration by the ant *Temnothorax curvispinosus*. *Behavioral Ecology and Sociobiology*. 2008; 62(8):1369–1376. doi: [10.1007/s00265-008-0565-9](https://doi.org/10.1007/s00265-008-0565-9)
15. Agnati LF, Zoli M, Strömberg I, Fuxe K. Intercellular communication in the brain: Wiring versus volume transmission. *Neuroscience*. 1995; 69(3):711–726. doi: [10.1016/0306-4522\(95\)00308-6](https://doi.org/10.1016/0306-4522(95)00308-6) PMID: [8596642](https://pubmed.ncbi.nlm.nih.gov/8596642/)
16. Agnati LF, Guidolin D, Guescini M, Genedani S, Fuxe K. Understanding wiring and volume transmission. *Brain Research Reviews*. 2010; 64(1):137–159. doi: [10.1016/j.brainresrev.2010.03.003](https://doi.org/10.1016/j.brainresrev.2010.03.003) PMID: [20347870](https://pubmed.ncbi.nlm.nih.gov/20347870/)
17. Godoy A, Von Zuben FJ. Topology of social networks and efficiency of collective intelligence methods. In: *Proceedings of the 15th Annual Conference Companion on Genetic and Evolutionary Computation. GECCO'13 Companion*. New York, USA: ACM; 2013. p. 1415–1422. Available from: <http://dx.doi.org/10.1145/2464576.2482721>.
18. Page L, Brin S, Motwani R, Winograd T. The PageRank citation ranking: Bringing order to the web. *Stanford InfoLab*; 1999. 1999–66. Available from: <http://ilpubs.stanford.edu:8090/422/>
19. Newman MEJ. The structure and function of complex networks. *SIAM Review*. 2003; 45(2):167–256. doi: [10.1137/S003614450342480](https://doi.org/10.1137/S003614450342480).
20. Chen DB, Gao H, Lü L, Zhou T. Identifying influential nodes in large-scale directed networks: The role of clustering. *PLoS ONE*. 2013; 8(10):e77455. doi: [10.1371/journal.pone.0077455](https://doi.org/10.1371/journal.pone.0077455) PMID: [24204833](https://pubmed.ncbi.nlm.nih.gov/24204833/)
21. Oshima H, Odagaki T. Storage capacity and retrieval time of small-world neural networks. *Phys Rev E*. 2007; 76:036114. doi: [10.1103/PhysRevE.76.036114](https://doi.org/10.1103/PhysRevE.76.036114)
22. Qian J, Hintze A, Adami C. Colored motifs reveal computational building blocks in the *C. elegans* brain. *PLoS ONE*. 2011; 6(3):e17013. doi: [10.1371/journal.pone.0017013](https://doi.org/10.1371/journal.pone.0017013) PMID: [21408227](https://pubmed.ncbi.nlm.nih.gov/21408227/)
23. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U. Network motifs: Simple building blocks of complex networks. *Science*. 2002; 298(5594):824–827. doi: [10.1126/science.298.5594.824](https://doi.org/10.1126/science.298.5594.824) PMID: [12399590](https://pubmed.ncbi.nlm.nih.gov/12399590/)
24. Romano ALT, Villanueva WJP, Zanetti MS, Von Zuben FJ. Synthesis of spatio-temporal models by the evolution of non-uniform cellular automata. In: Abraham A, Hassanien AE, de Carvalho APLF, editors. *Foundations of Computational Intelligence Volume 4*. vol. 204 of *Studies in Computational Intelligence*. Springer Berlin Heidelberg; 2009. p. 85–104. Available from: http://dx.doi.org/10.1007/978-3-642-01088-0_4.
25. Turing AM. On computable numbers, with an application to the *Entscheidungsproblem*. *Proceedings of the London Mathematical Society*. 1937; s2-42(1):230–265.
26. Wolfram S. *A new kind of science*. Champaign, USA: Wolfram Media Inc.; 2002.
27. Cook M. Universality in elementary cellular automata. *Complex Systems*. 2004; 15(1):1–40.
28. Land M, Belew RK. No perfect two-state cellular automata for density classification exists. *Phys Rev Lett*. 1995; 74:5148–5150. doi: [10.1103/PhysRevLett.74.5148](https://doi.org/10.1103/PhysRevLett.74.5148) PMID: [10058695](https://pubmed.ncbi.nlm.nih.gov/10058695/)
29. Li W, Packard N. The structure of the elementary cellular automata rule space. *Complex Systems*. 1990; 4(3):281–297.
30. Packard NH. Adaptation toward the edge of chaos. In: Kelso JAS, Mandell AJ, Shlesinger MF, editors. *Dynamic Patterns in Complex Systems*. World Scientific; 1988. p. 293–301.
31. Mitchell M, Hraber P, Crutchfield JP. Revisiting the Edge of Chaos: Evolving Cellular Automata to Perform Computations. *Complex Systems*. 1993; 7(2):89–130.

32. Mitchell M, Crutchfield JP, Hraber PT. Evolving cellular automata to perform computations: Mechanisms and impediments. *Physica D: Nonlinear Phenomena*. 1994; 75(1-3):361–391. doi: [10.1016/0167-2789\(94\)90293-3](https://doi.org/10.1016/0167-2789(94)90293-3)
33. Das R, Mitchell M, Crutchfield JP. A genetic algorithm discovers particle-based computation in cellular automata. In: Davidor Y, Schwefel HP, Männer R, editors. *Parallel Problem Solving from Nature – PPSN III*. vol. 866 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg; 1994. p. 344–353. Available from: http://dx.doi.org/10.1007/3-540-58484-6_278.
34. Andre D, Bennett FH III, Koza JR. Discovery by genetic programming of a cellular automata rule that is better than any known rule for the majority classification problem. In: *Proceedings of the 1st Annual Conference on Genetic Programming*. Cambridge, USA: MIT Press; 1996. p. 3–11. Available from: <http://dl.acm.org/citation.cfm?id=1595536.1595538>.
35. Gács P, Kurdyumov GL, Levin LA. One-dimensional uniform arrays that wash out finite islands. *Problemy Peredachi Informatsii*. 1978; 14(3):92–96.
36. Juille H, Pollack JB. Coevolving the ideal trainer: Application to the discovery of cellular automata rules. In: Koza JR, Banzhaf W, Chellapilla K, Deb K, Dorigo M, Fogel DB, et al., editors. *Genetic Programming 1998: Proceedings of the Third Annual Conference*. San Francisco, USA: Morgan Kaufmann; 1998. p. 519–527. Available from: http://www.cs.brandeis.edu/~hugues/papers/GP_98.ps.gz.
37. Oliveira GMB, Martins LGA, de Carvalho LB, Fynn E. Some investigations about synchronization and density classification tasks in one-dimensional and two-dimensional cellular automata rule spaces. *Electronic Notes in Theoretical Computer Science*. 2009; 252(0):121–142. doi: [10.1016/j.entcs.2009.09.018](https://doi.org/10.1016/j.entcs.2009.09.018)
38. Wolz D, de Oliveira PPB. Very effective evolutionary techniques for searching cellular automata rule spaces. *J Cellular Automata*. 2008; 3(4):289–312.
39. Watts DJ, Strogatz SH. Collective dynamics of ‘small-world’ networks. *Nature*. 1998; 393(6684):440–442. doi: [10.1038/30918](https://doi.org/10.1038/30918) PMID: [9623998](https://pubmed.ncbi.nlm.nih.gov/9623998/)
40. Barabási AL, Albert R. Emergence of scaling in random networks. *Science*. 1999; 286(5439):509–512. doi: [10.1126/science.286.5439.509](https://doi.org/10.1126/science.286.5439.509) PMID: [10521342](https://pubmed.ncbi.nlm.nih.gov/10521342/)
41. Mathias N, Gopal V. Small worlds: How and why. *Phys Rev E*. 2001; 63:021117. doi: [10.1103/PhysRevE.63.021117](https://doi.org/10.1103/PhysRevE.63.021117)
42. Ferrer i, Cancho R, Solé RV. Optimization in complex networks. In: Pastor-Satorras R, Rubi M, Diaz-Guilera A, editors. *Statistical Mechanics of Complex Networks*. vol. 625 of *Lecture Notes in Physics*. Springer Berlin Heidelberg; 2003. p. 114–126. Available from: http://dx.doi.org/10.1007/978-3-540-44943-0_7.
43. Kauffman S. Homeostasis and differentiation in random genetic control networks. *Nature*. 1969; 224(5215):177–178. doi: [10.1038/224177a0](https://doi.org/10.1038/224177a0) PMID: [5343519](https://pubmed.ncbi.nlm.nih.gov/5343519/)
44. Aldana M, Cluzel P. A natural class of robust networks. *Proceedings of the National Academy of Sciences*. 2003; 100(15):8710–8714. doi: [10.1073/pnas.1536783100](https://doi.org/10.1073/pnas.1536783100)
45. Sipper M, Ruppel E. Co-evolving architectures for cellular machines. *Physica D: Nonlinear Phenomena*. 1997; 99(4):428–441. doi: [10.1016/S0167-2789\(96\)00172-8](https://doi.org/10.1016/S0167-2789(96)00172-8)
46. Watts DJ. *Small worlds: The dynamics of networks between order and randomness*. Princeton University Press; 2003.
47. Tomassini M, Giacobini M, Darabos C. Evolution of small-world networks of automata for computation. In: Yao X, Burke EK, Lozano JA, Smith J, Merelo-Guervós JJ, Bullinaria JA, et al., editors. *Parallel Problem Solving from Nature—PPSN VIII*. vol. 3242 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg; 2004. p. 672–681. Available from: http://dx.doi.org/10.1007/978-3-540-30217-9_68.
48. Tomassini M, Giacobini M, Darabos C. Evolution and dynamics of small-world cellular automata. *Complex Systems*. 2005; 15(4):261–284.
49. Crutchfield JP, Mitchell M, Das R. Evolutionary design of collective computation in cellular automata. In: Crutchfield JP, Schuster P, editors. *Evolutionary Dynamics: Exploring the Interplay of Selection, Accident, Neutrality, and Function*. Santa Fe Institute Studies in the Sciences of Complexity. Oxford University Press; 2003. p. 361–411.
50. Darabos C, Giacobini M, Tomassini M. Scale-free automata networks are not robust in a collective computational task. In: El Yacoubi S, Chopard B, Bandini S, editors. *Cellular Automata*. vol. 4173 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg; 2006. p. 512–521. Available from: http://dx.doi.org/10.1007/11861201_59.
51. Darabos C, Giacobini M, Tomassini M. Performance and robustness of cellular automata computation on irregular networks. *Advances in Complex Systems*. 2007; 10(supp01):85–110. doi: [10.1142/S0219525907001124](https://doi.org/10.1142/S0219525907001124)

52. Darabos C, Tomassini M, Di Cunto F, Provero P, Moore JH, Giacobini M. Toward robust network based complex systems: From evolutionary cellular automata to biological models. *Intelligenza Artificiale*. 2011; 5(1):37–47.
53. Macêdo HB, Oliveira GMB, Ribeiro CHC. Dynamic behaviour of chaotic cellular automata—A comparative entropy analysis of regular lattices and small-world structures. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC); 2013. p. 1566–1571. Available from: <http://dx.doi.org/10.1109/SMC.2013.270>.
54. Oikonomou P, Cluzel P. Effects of topology on network evolution. *Nat Phys*. 2006; 2:532–536. doi: [10.1038/nphys359](https://doi.org/10.1038/nphys359)
55. Marr C, Hütt MT. Topology regulates pattern formation capacity of binary cellular automata on graphs. *Physica A: Statistical Mechanics and its Applications*. 2005; 354(0):641–662. doi: [10.1016/j.physa.2005.02.019](https://doi.org/10.1016/j.physa.2005.02.019)
56. Goldberg DE. Genetic algorithms in search, optimization and machine learning. 1st ed. Boston, USA: Addison-Wesley Longman Publishing Co., Inc.; 1989.
57. Wolfram S. Statistical mechanics of cellular automata. *Rev Mod Phys*. 1983; 55:601–644. doi: [10.1103/RevModPhys.55.601](https://doi.org/10.1103/RevModPhys.55.601)
58. Andreica A, Chira C. New majority rule for network based cellular automata. *Studia Universitatis Babeş-Bolyai, Informatica*. 2012; 57(3).
59. Lizier JT, Pritam S, Prokopenko M. Information dynamics in small-world Boolean networks. *Artificial Life*. 2011; 17:293–314. doi: [10.1162/artl_a_00040](https://doi.org/10.1162/artl_a_00040) PMID: [21762020](https://pubmed.ncbi.nlm.nih.gov/21762020/)
60. Crutchfield JP, Hanson JE. Turbulent pattern bases for cellular automata. *Physica D: Nonlinear Phenomena*. 1993; 69(3-4):279–301. doi: [10.1016/0167-2789\(93\)90092-F](https://doi.org/10.1016/0167-2789(93)90092-F)
61. Hordijk W, Crutchfield JP, Mitchell M. Embedded particle computation in evolved cellular automata. Santa Fe Institute; 1996. 96-08-073. Available from: <http://www.santafe.edu/research/working-papers/abstract/cd50a40a90d73e724bd648fce91b53c6/>.
62. Marques-Pita M, Rocha LM. Schema redescription in cellular automata: Revisiting emergence in complex systems. In: 2011 IEEE Symposium on Artificial Life (ALIFE); 2011. p. 233–240. Available from: <http://dx.doi.org/10.1109/ALIFE.2011.5954670>.
63. Lizier JT, Prokopenko M, Zomaya AY. Detecting non-trivial computation in complex dynamics. In: Proceedings of the 9th European Conference on Advances in Artificial Life. ECAL'07. Springer Berlin Heidelberg; 2007. p. 895–904. Available from: http://dx.doi.org/10.1007/978-3-540-74913-4_90.
64. Lizier JT, Prokopenko M, Zomaya AY. A framework for the local information dynamics of distributed computation in complex systems. In: Prokopenko M, editor. Guided Self-Organization: Inception. vol. 9 of Emergence, Complexity and Computation. Springer Berlin Heidelberg; 2014. p. 115–158. Available from: http://dx.doi.org/10.1007/978-3-642-53734-9_5.
65. Lizier JT, Flecker B, Williams PL. Towards a synergy-based approach to measuring information modification. In: 2013 IEEE Symposium on Artificial Life (ALIFE); 2013. p. 43–51. Available from: <http://dx.doi.org/10.1109/ALIFE.2013.6602430>.
66. Gastwirth JL. The estimation of the Lorenz curve and Gini index. *The Review of Economics and Statistics*. 1972; 54:306–316. doi: [10.2307/1937992](https://doi.org/10.2307/1937992)
67. Langville AN, Meyer CD. A survey of eigenvector methods for web information retrieval. *SIAM Review*. 2005; 47(1):135–161. doi: [10.1137/S0036144503424786](https://doi.org/10.1137/S0036144503424786)
68. Newman MEJ. Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality. *Phys Rev E*. 2001; 64:016132. doi: [10.1103/PhysRevE.64.016132](https://doi.org/10.1103/PhysRevE.64.016132)
69. Wernicke S. Efficient Detection of Network Motifs. *IEEE/ACM Trans Comput Biol Bioinformatics*. 2006; 3(4):347–359. doi: [10.1109/TCBB.2006.51](https://doi.org/10.1109/TCBB.2006.51)
70. Katare S, West DH. Optimal complex networks spontaneously emerge when information transfer is maximized at least expense: A design perspective. *Complexity*. 2006; 11(4):26–35. doi: [10.1002/cplx.20119](https://doi.org/10.1002/cplx.20119)
71. Tassier T, Menczer F. Emerging small-world referral networks in evolutionary labor markets. *IEEE Transactions on Evolutionary Computation*. 2001; 5(5):482–492. doi: [10.1109/4235.956712](https://doi.org/10.1109/4235.956712)
72. Fowler JH, Settle JE, Christakis NA. Correlated genotypes in friendship networks. *Proceedings of the National Academy of Sciences*. 2011; 108(5):1993–1997. doi: [10.1073/pnas.1011687108](https://doi.org/10.1073/pnas.1011687108) PMID: [21245293](https://pubmed.ncbi.nlm.nih.gov/21245293/)
73. Heylighen F, Heath M, Van Overwalle F. The emergence of distributed cognition: a conceptual framework. In: Proceedings of Collective Intentionality IV; 2004.