Similar local neuronal dynamics may lead to different collective behavior

Margarita M. Sánchez Díaz[1,3], Eyisto J. Aguilar Trejo[1,2,3], Daniel A. Martin[1,2,3], Sergio A. Cannas[2,4], Tomás S. Grigera[2,5,6], Dante R. Chialvo[1,2,3]

[1] *Center for Complex Systems and Brain Sciences (CEMSC^3), Universidad Nacional de San Martín, Campus Miguelete, 25 de Mayo y Francia, (1650), San Martín, Buenos Aires, Argentina.*

[2] *Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Godoy Cruz 2290, (1425), Buenos Aires, Argentina.*

[3] *Instituto de Ciencias Físicas (ICIFI), CONICET and Universidad Nacional de San Martín 25 de Mayo y Francia, (1650), San Martín, Buenos Aires, Argentina.*

[4] *Instituto de Física Enrique Gaviola (IFEG-CONICET), Facultad de Matemática Astronomía Física y Computación, Universidad Nacional de Córdoba, 5000 Córdoba, Argentina.*

[5] *Instituto de Física de Líquidos y Sistemas Biológicos (IFLySiB), CONICET and Universidad Nacional de La Plata, Calle 59 no. 789, B1900BTE, La Plata, Buenos Aires, Argentina. and*

[6] *Departamento de Física, Facultad de Ciencias Exactas, Universidad Nacional de La Plata, 1900, La Plata, Buenos Aires, Argentina.*

Abstract

This report is concerned with the relevance of the microscopic rules, that implement individual neuronal activation, in determining the collective dynamics, under variations of the network topology. To fix ideas we study the dynamics of two cellular automaton models, commonly used, rather in-distinctively, as the building blocks of large-scale neuronal networks. One model, due to Greenberg & Hastings (GH), can be described by evolution equations mimicking an integrate-and-fire process, while the other model, due to Kinouchi & Copelli (KC), represents an abstract branching process, where a single active neuron activates a given number of postsynaptic neurons according to a prescribed "activity" branching ratio. Despite the apparent similarity between the local neuronal dynamics of the two models, it is shown that they exhibit very different collective dynamics as a function of the network topology. The GH model shows qualitatively different dynamical regimes as the network topology is varied, including transients to a ground (inactive) state, continuous and discontinuous dynamical phase transitions. In contrast, the KC model only exhibits a continuous phase transition, independently of the network topology. These results highlight the importance of paying attention to the microscopic rules chosen to model the inter-neuronal interactions in large-scale numerical simulations, in particular when the network topology is far from a mean-field description. One such case is the extensive work being done in the context of the Human Connectome, where a wide variety of types of models are being used to understand the brain collective dynamics.

I. INTRODUCTION

The animal brain is composed of billions of neurons, which interact with each other through thousands of synapses per neuron. The results of such interaction is the emergence of complex spatio-temporal patterns of neuronal activity supporting perception, action and behavior. A recent proposal considers the brain as a network of neurons poised near a dynamical transition [1–4], a view which is supported by experimental results gathered from animals both *in vitro*[5] and *in vivo* [6] as well as from whole brain neuroimaging human experiments [7–9].

The potential existence of critical phenomena in the brain motivated, during the last decade, the study of mathematical models to explore better the large-scale brain dynamics. A distinctive difference between the diversity of models is at the microscopic level. Some models consist of networks of simplified neurons, in which neurons themselves are represented by a wide variety of approaches, ranging from 2-state particles [10] through discrete cellular automatons [11–14], branching processes [6, 15–18], neural masses [19], coupled-maps [20–23], coupled Kuramoto oscillators [24] up to detailed equations describing the evolution and spiking of the membrane potential [25–27]. Thus, a natural question arises on how relevant may the microscopic process (used to represent the individual neuronal dynamics) be, and how they affect the dynamical collective repertoire exhibited by the network.

When focusing on collective properties, it is of course reasonable to seek minimal models which, even orphan of realistic microscopic rules, may reproduce relevant macroscopic behavior. However, it is not straightforward to determine in principle how general this assumption can be in the case of neuronal networks. Our point is that, even though the use of realistic microscopic dynamics is not necessarily a prerequisite to correctly describe universal macroscopic properties, *microscopic rules do matter* and eventually can lead to different universal behavior. As a clarifying metaphor, consider the Ising model. It is well known that algorithms with unrealistic non-local moves (so-called cluster algorithms [28]) can correctly describe the static critical behavior. However, if the dynamic rule did not follow detailed balance, then the modified dynamics would fail to reproduce equilibrium behavior, even if it could reproduce some sort of critical dynamics. And of course, even with detailed balance, the dynamical universality is altered by the non-local rule. A similar correspondence among microscopic rules and system's dynamics appears when modeling brain dynamics, which is

rarely considered, thus some extrapolations to real brain dynamics taken from numerical simulations in the current literature, may be hampered by the limitations of the microscopic details of the neuronal models employed. We are purposely not considering here a large chapter of models that include synaptic plasticity.

In this article, we illustrate the problem by studying the dynamics of two apparently similar neural network models: that of Greenberg & Hastings [29] as described in [11, 12] and that of Kinouchi & Copelli [15]. The main difference between these two models is related to the microscopic rule that propagates the activity: the first proposes (as many others of the same kind) a neuronal interaction rule that depends on the state of its presynaptic neighbors, while the second introduces a rule that, regardless of the number of connections, maintains a prescribed branching of activity on the target neurons. At first sight the differences seem innocent-looking, but as it will be shown, they lead to completely different behavior of the network: the first model exhibits continuous or discontinuous phase transitions depending on the network topology, while the second is completely insensitive to it.

We remark from the outset that the article' aim is not to criticize any given model in particular, but to call the attention on the consequences of using them ignoring the limitations of the model's original formulation, including possible misinterpretations. The article is organized as follows: in Sec. II we describe both models and the observables that will be used to characterize the dynamical regimes, in Sec. III we show the results of the numerical simulations, in Sec. IV we discuss present results in the context of recent research and we summarize the conclusions.

II. NETWORK, MODELS AND OBSERVABLES

A. The interaction network

Both neuronal models are studied on an undirected Watts-Strogatz small-world network [30] with average connectivity $\langle k \rangle$ and rewiring probability π. The network is constructed as usual [30] by starting from a ring of N nodes (always $N = 20000$ in this report), each connected symmetrically to its $\langle k \rangle/2$ nearest neighbors; then each link connecting a node to a clockwise neighbor is rewired to a random node with probability π, so that average connectivity is preserved. The rewiring probability is a measure of the disorder in the network: for

$\pi = 0$ the network is circular and perfectly ordered, while for $\pi = 1$ it becomes completely random. In both models neurons are represented as nodes on a weighted undirected random graph with an associated discrete state variable, $S_i = 0, \ldots, n$, where $i = 1, \ldots, N$ identifies the node and $S_i = \{0, \ldots, n\}$. State 0 represents a quiescent (but excitable) neuron, 1 is the active state, and $2 \ldots n$ are refractory states. The links of the graphs are represented by the $N \times N$ connectivity matrix W. Nonzero matrix elements indicate the presence of a link with a given weight. Weights are positive reals, so $W_{ji} \geq 0$, and the connectivity matrix is symmetric, $W_{ij} = W_{ji}$, since the graph is undirected. In this context, symmetric connections need to be interpreted as two connections between any pair of nodes. Neither the connectivity nor the weights depend on time (i.e., we consider quenched disorder). The dynamical evolution is given by a discrete-time Markov process in which all sites are simultaneously updated, and with transition probabilities for each site given by the expressions below for each model.

1. *GH model:*

This model was introduced by Greenberg & Hastings [29] to mimic the excitable dynamics generically observed in neurons, forest fires, cardiac cells, chemical reactions and epidemic propagation. In the context of brain dynamics it was used recently by Haimovici *et al.* [11]. Here we follow closely the implementation of Zarepour *et al.* [12]. It is a cellular automaton endowed of the three states common to excitable dynamics: quiescent, active and refractory state, and the dynamics of site i is updated by

$$P_{i,0\to1} = 1 - [1 - r_1] \left[1 - \Theta \left(\sum_{j=1}^{k_{in,i}} W_{ji} \delta_{S_j,1} - T \right) \right], \tag{1a}$$

$$P_{i,1\to2} = 1, \tag{1b}$$

$$P_{i,2\to0} = r_2, \tag{1c}$$

where $P_{i,a\to b}$ is the probability that site i will transition from state a to state b, at time $t+1$, S_i is computed at time t, and the sum is performed over all j targeting i and $k_{in,i}$ is the in-degree of node i. $\Theta(x)$ is Heaviside's step function [$\Theta(x) = 1$ for $x \geq 0$ or 0 otherwise], $\delta_{i,j}$ is Kronecker's delta, and r_1, r_2 and T are control parameters which are set equal to all sites in the present work. Thus, an active site always turns refractory in the next time

step, and a refractory site becomes quiescent with probability r_2. The probability for a quiescent site to become active is written as 1 minus the product of the probabilities of *not* becoming active through the different mechanisms at work. In this model there are only two activation mechanisms: *spontaneous activation*, which occurs with a small probability r_1, or *transmitted activation,* which occurs deterministically[48]: neuron i will become active if and only if it is in the quiescent state and the sum of the weights of the links connecting i to its active neighbors exceeds a threshold T (see Panel (a) of Fig. 1). The non-null weights are drawn from an exponential distribution, $p(W_{ji} = w) = \lambda e^{-\lambda w}$, with $\lambda = 12.5$ chosen to mimic the weight distribution of the human connectome [12]. For the simulations described here, we use $r_1 = 0.001$, $r_2 = 0.3$ as in previous work [11, 12] which remain fixed in all simulations, while T is used as control parameter.

2. *KC model:*

This model was introduced by Kinouchi & Copelli [15] to show that a (Erdös–Renyi undirected) network of excitable elements has its sensitivity and dynamic range maximized at the critical point of a non-equilibrium phase transition. The model resembles a branching process [31, 32] in which the transition probabilities for neuron i at time $t + 1$ are:

$$P_{i,0\to 1} = 1 - [1 - r_1] \prod_{j=1}^{k_{in,i}} [1 - pW_{ji}\delta_{S_j,1}], \tag{2a}$$

$$P_{i,1\to 2} = 1, \tag{2b}$$

$$P_{i,2\to 3} = 1, \tag{2c}$$

$$\vdots$$

$$P_{i,n\to 0} = 1. \tag{2d}$$

S_j is evaluated at t, and the product is taken over all neurons j pointing to i.The interaction matrix is symmetric, $W_{ij} = W_{ji}$, and non-null elements are taken uniformly from $[0, 1]$. The interaction rule in Eq. 2a contains two parameters: r_1 which (as in the GH model) determines the spontaneous activity of any inactive neuron and p which controls the transmitted activation probability (see Panel (b) of Fig. 1). This rule makes the main difference with the GH model. If the variance of the chosen values for k_{out} and W are rel-

atively small (as in ref. [15]), and $\langle k \rangle$ is relatively large, each active neuron will excite, on average $\sigma_p := (\langle k \rangle - 1)p/2$ neurons. Thus, σ_p is an approximate of the desired branching ratio. Similar to the GH model, an active site always becomes refractory, but instead of recovering randomly, here it becomes quiescent deterministically after $n-1$ time steps. We note that this difference has no relevance for the present analysis.

It is known that, for a wide variety of conditions, critical dynamics is expected for $\sigma_p \simeq 1$ [31]. The critical value of σ_p will be exactly 1 only in the limit of very large networks ($N \to \infty$), with homogeneous degree distribution, without triangles (i.e., a clustering coefficient equal to zero), and where the dynamics is in the limit of negligible spontaneous activation ($r_1 \to 0$). For finite values of r_1, several unrelated activity (i.e. avalanches) may coexist. Under this situation, the system may belong to a different universality class [33]. For a comprehensive analysis on the role of spontaneous activation on modifying the universality class, and how it may also shift the critical value of the branching ratio, we refer the reader to [33] and references therein. In the following, we will use σ_p instead of p as the control parameter. This choice does not affect the dynamics since both quantities are proportional. Examples of the results as a function of p instead of σ_p can be found in the Supplemental Material [34].

For the simulations, values of $r_1 = 0.001$, and $n = 4$ (i.e., a fixed refractory period of 3 steps) are chosen, which remain fixed in all simulations. In passing, please notice that the interaction rule in the KC model is entirely stochastic and that neurons behave independently (as long as the spontaneous activity is relatively low as dictated by the value of r_1 used here). Additional details can be found in Costa et al. [35] and Campos et al. [36]. The numerical implementation of the KC model admits a few variations which, nonetheless, do not change the present results (see Supplemental Material [34]).

B. Observables

To describe the state of the network, for both models, we define an order parameter $f_S(t)$ which corresponds to the fraction of active neurons at time t,

$$f_S(t) = \frac{1}{N} \sum_i \delta_{S_i(t),1}. \tag{3}$$

After any transient dies out, we also compute its variance, $\sigma^2_{f_S} = \langle f_S^2 \rangle - \langle f_S \rangle^2$, where $\langle ... \rangle$ is a time average.

For the purposes of the present work, it is of particular interest the behavior of the (normalized) connected autocorrelation of the order parameter f_S,

$$AC(\Delta t) = \frac{1}{\sigma^2_{f_S}} \Big\langle \big(f_S(t) - \langle f_S \rangle\big) \times \big(f_S(t+\Delta t) - \langle f_S \rangle\big) \Big\rangle. \tag{4}$$

which estimates the linear correlation between the network state at times t and $t + \Delta t$, with $AC(\Delta t) \simeq 1$ for highly correlated consecutive configurations and $AC(\Delta t) \simeq 0$ when the configurations quickly decorrelate. It is known that the autocorrelation function is sensitive to the different dynamical regimes: close to a continuous phase transition, the dynamics undergoes critical slowing down, which implies that the autocorrelation function decays slower than in the supercritical or subcritical state [37]. For discontinuous phase transitions, a similar effect takes place at the spinodal points [38]. Here we focus on the autocorrelation at $\Delta t = 1$, $AC(1)$, also called first correlation coefficient, which has been shown to have a maximum at the transition point [37].

C. Parametric exploration

In this work we are interested in exploring the extent of the dynamical repertoire that each model is able to exhibit under a very wide range of: 1) neuronal dynamics and 2) topology of the underlying network. Thus, we proceed to scan the control parameter of the given neuron model for different network topologies (by varying $\langle k \rangle$ and π). This implies to explore three parameters while classifying the dynamical regimes observed.

To identify and classify the dynamical regimes, we track the behavior of $AC(1)$ as the control parameter (T or σ_p) is increased and decreased. This is repeated for each combination of network parameters $\langle k \rangle$ and π. The simulations start at T_0 (or σ_0) (using a random initial condition for each neuron) and then it is increased by ΔT (or $\Delta\sigma$) after a given number of steps, up to a final value T_F (or σ_F), without resetting the neuron states when changing the value of the control parameter. This parametric exploration allows us to determine the full repertoire of dynamical regimes which can emerge from the microscopic activity propagation rules acting on a given network topology.

III. RESULTS

A. Characterizing the transitions

Now we proceed to describe how the dynamical repertoire of each model is determined from a parametric exploration. An example is presented in Fig. 2 where panels on the left correspond to results obtained from the GH model and those on the right from the KC model. The figure shows that, as expected, the rate of activity changes as a function of its control parameter, but already demonstrating an important difference between the dynamical regimes exhibited by the two models. For this particular choice of topology, $\langle k \rangle = 30$ and $\pi = 0.6$, the GH model undergoes a discontinuous transition demonstrated by the abrupt change in f_S (also noted in the appearance of the raster plot) and the presence of hysteresis. In contrast, in response to similar parametric scan, the KC model exhibits a continuous transition and does not show hysteresis. In addition, it is important to note that the GH model shows a large increase in the variability of the order parameter f_S near the transition (see panel b), meanwhile the variance of the f_S fluctuations shown by the KC model is relatively constant (see panel e), regardless of the value of the control parameter σ_p. These observations point to important dynamical differences between the two models, as will be expanded in the next sections.

The behavior of the autocorrelation function of the order parameter helps to identify the type of phase transition because it is known to peak near a transition. We compute $AC(1)$ for each value of the control parameter, and define T_+ as the value that maximizes $AC(1)$ in a run when T is being increased, and T_- as the value that maximizes $AC(1)$ when decreasing T. An example for the GH model is presented in Fig. 3. For the KC model, we defined in the same way σ_p+ and σ_p-, although we never observed discontinuous transitions in that model. Results for the KC model for the same conditions and network topologies as in Fig. 3 are shown in the Supplemental Material [34].

Thus, according to the shape of the curves of $AC(1)$ vs. control parameters, we can classify the dynamical behavior: if $AC(1)$ is monotonic, then there is no phase transition, corresponding to the cases in which the network, after a transient, goes quiescent. For network topologies in which $|T_+ - T_-| \geq 2\Delta T$ (or $|\sigma_+ - \sigma_-| \geq \Delta\sigma_p$) the transition is considered discontinuous, and continuous otherwise. In other words, after exploring a reasonable range

of values of the control parameter, the existence of a maximum in the $AC(1)$ curve indicates (under the present context) a phase transition, which is considered continuous if there is no noticeable hysteresis or discontinuous otherwise.

An example of the behavior of $AC(1)$ in the case of a continuous transition is shown in Fig. 4. This type of transition is observed in both models for a wide range of $\langle k \rangle$ and π values, as will be described in the next section. It can be seen that a change of the control parameter on a range of values near the critical point is reflected on a non-monotonic change of the $AC(1)$. The plots in the bottom panels illustrate the typical autocorrelation function of the order parameter f_S. For control parameter values larger than T_c (or smaller than σ_c) the activity correlation vanishes quickly as indicated by the green triangle data points. In the other extreme, for control parameter values smaller than T_c (or larger than σ_c, i.e., data points plotted as blue circles) the function shows an oscillatory pattern. The first zero crossing of the function is dictated by the duration of the refractory period of the neuronal models which is one of the determinants of the collective oscillation frequency. Finally, for values sufficiently close to T_c (or σ_c) the function $AC(\Delta t)$ decays very slowly (as a power law) as shown in the figure by the data points plotted with red squares.

B. Models's dynamical repertoire on parameter space

Here we describe the results of a systematic exploration of the collective dynamics as a function of network topology in each model. For each value of $\langle k \rangle$ and π, we computed 5 realizations of Watts-Strogatz graphs. In each case we classified the regimes as a function of the control parameter, according to the behavior of $AC(1)$ as explained above. The dynamical regimes found include *transients to no-activity*, *continuous phase transition* or *discontinuous phase transition (from no-activity to collective oscillations)*.

The results in Fig. 5 show the regions of parameters at which each regime was observed. In brief, both models exhibit no-transition for network topologies with $\langle k \rangle = 2$ and connectivity disorder $\pi > 0$ (red zone with squares in Fig. 5). For $\pi = 0$ the same regime extends to $\langle k \rangle < 6$ in both models.

For networks with relatively large values of $\langle k \rangle$ both models exhibit a *continuous phase transition* as in the example featured already in Fig. 4. (black zone with circles in Fig. 5). The main difference between the models is found for relatively high values of degree and

disorder. At this region of parameters, the GH model shows a discontinuous phase transition (blue zone with triangles), while the KC model presents a continuous one. The inset in Fig. 5 (a) shows the border *continuous* and *discontinuous* regions, as a function of network parameters, in a double logarithmic scale. A straight separation line, similar to the one observed by *Zarepour et al.*[12] in the context of cluster size distribution, is observed.

Close to the border between the *continuous transitions* and the *continuous transitions* region, there is a narrow range of $\langle k \rangle$ values for which some realizations of KC model show *continuous transitions*, while other realizations show *discontinuous transitions*. In those cases, the reported kind of transition was determined by the most frequent result. We have added error bars to Fig. 5 (a), whose width is half of the range of $\langle k \rangle$ values for which this coexistence was observed. We did not observe a similar effect at the boundary between *no-activity* to *continuous transition* regions.

The results in Fig. 6 are representative examples of the behavior of $AC(1)$ as a function of the control parameter for selected values of $\langle k \rangle$ and π. For the GH model, the largest values of $\langle k \rangle$ and π show clear hysteresis, with the peaks for the case of increasing T at a higher value than the peak found when is decreased. In most cases, the increasing and decreasing sweeps of control parameter yield the same curve, with a maximum value of $AC(1)$ close to 1. For $\pi = 0$, the (single) peak tends to be rather broad. Finally, for $\langle k \rangle = 2$ and any value of π, and for $k = 4$, $\pi = 0$, $AC(1)$ behaves monotonically, which is indicative of no phase transition. The KC model shows less variation among the curves, with only a narrow range of monotonous curves, and most of the $\langle k \rangle$, π plane yielding continuous transitions, presenting very similar AC(1) curves as a function of σ_p.

Finally, the results on Fig. 7 show examples of the spiking patterns observed as the control parameter is increased, for the same selected values of $\langle k \rangle$ and π illustrated in Fig. 6. The patterns were obtained by varying the control parameters from 3% below T_c (or T_+ in the discontinuous case) to 3% above (or from 3% above to 3% below the critical value of σ_p for KC model), and recording the spikes of 300 neurons along 300 time steps. We have used $T = 0$ or $\sigma = 2$ for networks showing no phase transition. For the GH model, there are several cases (blue raster plots) of discontinuous transitions where there is a sharp decrease of activity after crossing T_+. Continuous transitions with large $\langle k \rangle$ (such as $\langle k \rangle = 20$, $\pi \geq 0.4$), show bursts of synchronized activity that disappear for T slightly above T_c (black raster plots). Smooth changes in neuron activity are observed for continuous transitions with

smaller values of $\langle k \rangle$. Finally, for $\pi = 0$ the network topology corresponds to a circle (or to a torus for larger $\langle k \rangle$ values), so that neurons spiking at time $t+1$ are close neighbors of those spiking at time t, leading to linear wave-like propagation. Similar to what is observed in the previous figure, KC results show a reduced variety of behavior, where neither discontinuous transitions, nor bursts of synchronized activity are observed close to the critical point.

IV. DISCUSSION

Summarizing, we have revisited two simplified models of neuronal activation to show that subtle differences in the local dynamical rules may result in very different collective dynamics when embedded on networks. We found that the KC model dynamical repertoire includes, as a function of its control parameter, only continuous phase transitions being, by design, insensitive to the network topology. This is at odds with the GH model, in which each neuron outcome is influenced by its connectivity degree and therefore by the overall network topology.

We have used the first autocorrelation coefficient $AC(1)$ of the order parameter fluctuations and the presence or absence of hysteresis to identify whether a dynamic transition is present, and to distinguish continuous from discontinuous transitions. This observable is sensitive enough to system's state: it can be used to tune a system towards criticality [37]. None of the present results depend on the use of the autocorrelation to track the dynamics. The presence of phase transitions and hysteresis in these models has been studied with other observables, such as the fraction of active sites f_S, the variance of activity fluctuations $\sigma^2_{f_S}$, or cluster quantities such as the size of the largest or the second largest cluster (S_1, or S_2) as in [11, 12], yielding similar results. We used $AC(1)$ because its computation is straightforward and easy to replicate, it is almost parameter free, therefore very convenient for comparing two different models.

The key difference between the two models is in the rule that determines how the activity propagates from a given neuron to its connected neighbors. The GH model mimics a discrete integrate-and-fire process taking place in real neurons. There, the "decision" to fire is post-synaptic, based on the amount of total depolarization, on a small patch of membrane, produced by the contribution of hundreds to thousands of impinging neurons (notice the sum in Eq. (1a)). Disregarding the spontaneous activation term, the rule in the GH model is

completely deterministic. In fact it is equivalent to a discretized partial differential equation, where the state $S_I(t+1)$ of a post-synaptic neuron I at time $t+1$ depends only on local quantities: it is determined by the previous state of that neuron, $S_I(t)$, the contribution of all other presynaptic neurons J_1 to J_k at time t, the weights W_{ji} of the connections, and the excitation threshold, i.e., $S_I(t+1) = f\left[S_I(t), S_{J_1}(t), \ldots, S_{J_k}(t), W_{J_{i=1..k},J}, T\right]$.

In contrast, in the KC model the propagation rule is a probabilistic (notice the product Eq. (2a)) contagion-like process [31], where a single excited neuron determines, according to a prescribed value of σ_p, how many of all of the neurons that connects to will fire next time. Thus, here the decision of how many neurons will be activated is presynaptic: a spiking neuron J excites on average σ_p post-synaptic I neurons, independently of the state of the other neurons connected to the same I neuron. Since σ_p accounts already for the average out degree of the network, the rule in Eq. (2a) determines, *for each J neuron independently*, the probability that such active neuron will have no-descendants, one, or more than one descendant. It is then rather unsurprising that the KC model is insensitive to the network topology.

While GH model exhibits a wealth of possible behaviors as a function of network topology, such as synchronized activity close to the critical point, a critical region for a broad range of control parameters, or the existence of discontinuous transitions, the same is not observed in KC model. An important ingredient for having such behavior is the existence of a deterministic activation rule: in the Supplemental Material [34] we show results for a *probabilistic postsynaptic* model, inspired by the model in [39], that may be tuned from discontinuous to continuous transitions as the degree of randomness in the *transmitted activation* rule is increased. Also, to gain an insight on the source of the differences among this models, a simplified, mean-field version, of KC and GH models (where there is no refractory period, nor spontaneous activation) is analytically solved in the Supplemental Material [34]. It can be seen that the simplified *stochastic presynaptic* model has only one stable solution for each value of the control parameter σ, while the simplified *deterministic postsynaptic* model may have several solutions for the same value of T, as a function of the initial conditions.

Although this is not the focus of the present report, it is worth to mention that the KC model rule is biologically implausible. As explained above, the actual activation mechanism of a given neuron involves a myriad of influences (thousands in mammalian brains) over a very small area. In addition, one has to consider that the output of an active neuron,

after propagating trough its axon from hundreds of microns to millimeters, will stimulate all its contacts roughly *equally*. Thus, in order to excite a given number of neighbors, a given presynaptic neuron would actually have to have information on the number and state (active, refractory or silent) of the post-synaptic neurons and also of other presynaptic neurons attempting to excite the same neuron. This is biologically highly unrealistic, because the involved neurons may be centimeters away from each other, without a direct connection among them. Moreover, such hypothetically very well informed J neuron will have to selectively cancel its stimulation strength with certain I neurons, as dictated by the value of σ_p, a requirement completely impossible for biological neurons.

Of course, the discussion above is not affecting the valuable points made in ref. [15] where the KC model was introduced to show that a network of identical excitable elements achieves maximum dynamic range at criticality. Since the simulations in Ref. [15] where made on fixed topology networks (Erdös-Renyi) the present results are not affecting any of its conclusions which, is worth noticing, were replicated in many other models as well as in experiments. Less clear are the results in other cases, when the KC model was used to study the dynamics of non random topologies. These include the deviations from mean-field behavior found in scale free networks, as reported by Copelli & Campos [40] or Mosqueiro & Maia [41]. We note in passing that there are multiple instances in which the KC model was (mis)named as "Greenberg & Hastings stochastic model", somewhat confusing according to the present results, as in the reports by Copelli & Campos [40], Wu *et al.*, [42], Asis & Copelli [43], Mosqueiro & Maia [41], to name only a few.

It is worth to note that the crucial influence of topology on the type of dynamics exhibited by excitable models has been discussed earlier by Kuperman & Abramson [44] in the context of epidemics. They found that the network degree and disorder determines the conditions at which endemic or epidemic situations occur.

To conclude, the point here is not that one needs super-realistic microscopic rules to build a valid model, but that microscopic rules do matter, and may lead to different collective behavior. In particular, we have shown that the KC model exhibits only one type of transition, independent of a large variation in the network topology. The observation that topology has an influence on the dynamics of certain models is fundamental at the present time, where several large-scale international scientific collaborations are devoted to map and study the consequences of features of the human brain connectome [45–47]. Examples in-

clude the numerical simulations using simplified models over derived connectomes, in order to understand brain functioning [10]. However, *not all neural models are the same*, and special care should be taken on the biases and limitations introduced by the applied models, before drawing conclusions on real brains.

V. ACKNOWLEDGEMENTS

Work supported by the BRAIN initiative Grant U19 NS107464-01. DAM acknowledges financial support from ANPCyT Grant PICT-2016-3874 (AR) and the use of computational resources of the IFIMAR (UNMdP-CONICET) cluster.

---

[1] P. Bak, *How nature works: The science of self-organized criticality* (Springer Science, New York, 1996).

[2] D. R. Chialvo, Physica A **340**, 756 (2004).

[3] D. R. Chialvo, Nat. Phys. **6**, 744 (2010).

[4] T. Mora and W. Bialek, J. Stat. Phys. **144**, 268 (2011).

[5] J. M. Beggs and D. Plenz, J. Neurosci. **23**, 11167 (2003).

[6] T. L. Ribeiro, S. Ribeiro, H. Belchior, F. Caixeta, and M. Copelli, PloS one **9**, e94992 (2014).

[7] P. Expert, R. Lambiotte, D. R. Chialvo, K. Christensen, H. J. Jensen, D. J. Sharp, and F. Turkheimer, J. R. Soc. Interface **8**, 472 (2011).

[8] D. Fraiman and D. Chialvo, Front. physiol. **3**, 307 (2012).

[9] E. Tagliazucchi, P. Balenzuela, D. Fraiman, and D. Chialvo, Front. physiol. **3**, 15 (2012).

[10] G. Ódor, Phys. Rev. E **94**, 062411 (2016).

[11] A. Haimovici, E. Tagliazucchi, P. Balenzuela, and D. R. Chialvo, Phys. Rev. Lett. **110**, 178101 (2013).

[12] M. Zarepour, J. I. Perotti, O. V. Billoni, D. R. Chialvo, and S. A. Cannas, Phys. Rev. E **100**, 052138 (2019).

[13] R. P. Rocha, L. Koçillari, S. Suweis, M. Corbetta, and A. Maritan, Sci. Rep. **8**, 15682 (2018).

[14] S. A. Moosavi, A. Montakhab, and A. Valizadeh, Sci. Rep. **7**, 7107 (2017).

[15] O. Kinouchi and M. Copelli, Nat. Phys. **2**, 348 (2006).

[16] V. Priesemann, M. Wibral, M. Valderrama, R. Pröpper, M. Le Van Quyen, T. Geisel, J. Triesc, D. Nikolić, and M. H. J. Munk, Front. Syst. Neurosci. **8**, 108 (2014).

[17] W. L. Shew, H. Yang, T. Petermann, R. Roy, and D. Plenz, J. Neurosci. **29**, 15595 (2009).

[18] C. Haldeman and J. M. Beggs, Phys. Rev. Lett. **94**, 058101 (2005).

[19] N. Deschle, J. Ignacio Gossn, P. Tewarie, B. Schelter, and A. Daffertshofer, Front. Comput. Neurosci. **14**, 118 (2021).

[20] D. R. Chialvo, Chaos, Solitons & Fractals **5**, 461 (1995).

[21] N. F. Rulkov, Phys. Rev. E **65**, 041922 (2002).

[22] M. Girardi-Schappo, O. Kinouchi, and M. H. R. Tragtenberg, Phys. Rev. E **88**, 024701 (2013).

[23] B. Ibarz, J. Casado, and M. Sanjuán, Phys. Rep. **501**, 1 (2011).

[24] G. Ódor and J. Kelling, Sci. Rep. **9**, 19621 (2019).

[25] A. Levina, U. Ernst, and J. Michael Herrmann, Neurocomputing **70**, 1877 (2007).

[26] E. Izhikevich, IEEE Trans. Neural Netw. **14**, 1569 (2003).

[27] S.-S. Poil, R. Hardstone, H. D. Mansvelder, and K. Linkenkaer-Hansen, J. Neurosci. **32**, 9817 (2012).

[28] M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics* (Clarendon Press, Oxford, 2001).

[29] J. M. Greenberg and S. Hastings, SIAM J Appl Math. **34**, 515 (1978).

[30] D. J. Watts and S. H. Strogatz, Nature **393**, 440 (1998).

[31] S. Zapperi, K. B. Lauritsen, and H. E. Stanley, Phys. Rev. Lett. **75**, 4071 (1995).

[32] P. Alstrøm, Phys. Rev. A **38**, 4905 (1988).

[33] D. J. Korchinski, J. G. Orlandi, S. W. Son, and J. Davidsen, Physical Review X. **11**, 2 (2021) .

[34] See Supplemental Material at [URL will be inserted by publisher] for complementary results and a description of the Fortran 90 and Python 3 codes uploaded to github (https://github.com/DanielAlejandroMartin/Apparently-Similar-Neuronal).

[35] A. de Andrade Costa, M. Copelli, and O. Kinouchi, J. Stat. Mech. Theory Exp. **2015**, P06004 (2015).

[36] J. G. F. Campos, A. A. Costa, M. Copelli, and O. Kinouchi, Phys. Rev. E **95**, 042303 (2017).

[37] D. R. Chialvo, S. A. Cannas, T. S. Grigera, D. A. Martin, and D. Plenz, Sci. Rep. **10**, 12145 (2020).

[38] E. S. Loscar, E. E. Ferrero, T. S. Grigera, and S. A. Cannas, J. Chem. Phys. **131**, 024120 (2009).

[39] R. Pazzini, O. Kinouchi, and A.A. Costa, Physical Review E. **104**, 1 (2021) .

[40] M. Copelli and P. R. A. Campos, Eur. Phys. J. B **56**, 273 (2007).

[41] T. S. Mosqueiro and L. P. Maia, Phys. Rev. E **88**, 012712 (2013).

[42] A.-C. Wu, X.-J. Xu, and Y.-H. Wang, Phys. Rev. E **75**, 032901 (2007).

[43] V. R. V. Assis and M. Copelli, Phys. Rev. E **77**, 011923 (2008).

[44] M. Kuperman and G. Abramson, Phys. Rev. Lett. **86** 2909 (2001).

[45] H. Markram, Nat. Rev. Neurosci. **7**, 153 (2006).

[46] A. P. Alivisatos, M. Chun, G. M. Church, R. J. Greenspan, M. L. Roukes, and R. Yuste, Neuron **74**, 970 (2012).

[47] S. M. Sunkin, L. Ng, C. Lau, T. Dolbeare, T. L. Gilbert, C. L. Thompson, M. Hawrylycz, and C. Dang, Nucleic Acids Res. **41**, D996 (2012).

[48] Although the transmitted activation rule is clearly deterministic, we have decided to write it as a transition probability, to make a similar description for both models. This is achieved using a Heaviside function in Eq. 1a.

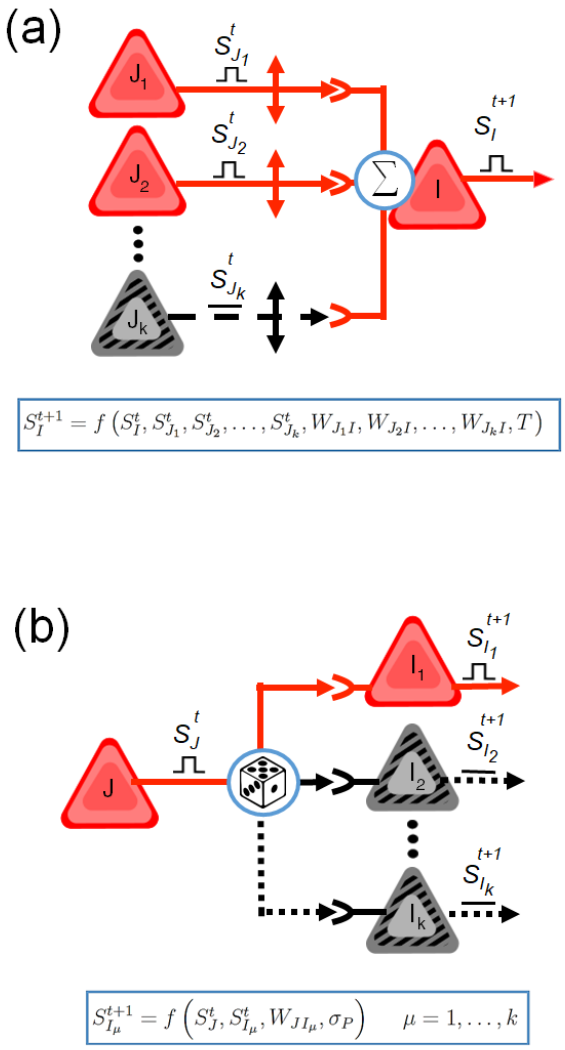


FIG. 1. Rule for the propagation of activity in both models. Panel (a): In the GH model, a *given neuron* I will become active at time $t+1$, if the contribution of all active presynaptic neurons (here J_1 and J_2), weighted by the interaction W_{JI} surpasses the threshold T. At each time step, this update is repeated for *all quiescent neurons* I. Panel (b): In the KC model *a given active neuron* J will activate at time step $t+1$ a given number of I neurons depending on the W_{JI} and σ_p value. Note that σ_p normalizes such probability by the number of interactions $\langle k \rangle$. At each time step, this update is repeated for *all active neurons* J. In both panels, triangles represent neurons and lines synaptic interactions. Red filled triangles denote active neurons and gray dashed ones inactive neurons, at times t and $t+1$. A summation term in panel (a) indicates that the outcome of neuron I depends on the sum of the contribution of neurons J_1 to J_n, while a dice in panel (b) indicates that stimulated neurons are randomly chosen by J.

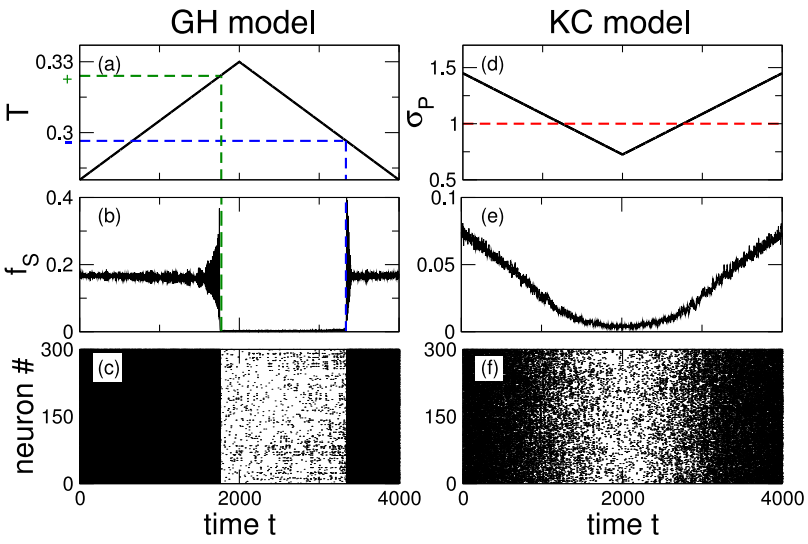


FIG. 2. Examples of the evolution of the activity as the respective control parameter (T or σ_p) is slowly varied. Left panels (a-c) correspond to the GH model and right panels (d-f) to the KC model. Top panels: value of the control parameters (T in (a) and σ_p in (d)) as a function of time t. Center panels: order parameter $f_S(t)$ for both models as a function of time. Bottom panels: raster plots of 300 selected neurons. Dashed lines in panels (a) and (b) mark the approximate values of $T_\pm$, and their respective times. In both cases $\langle k \rangle = 30$, $\pi = 0.6$. For the GH model, $\Delta T = \pm 0.000025$, and for the KC model $\Delta\sigma = \pm 0.00036$.

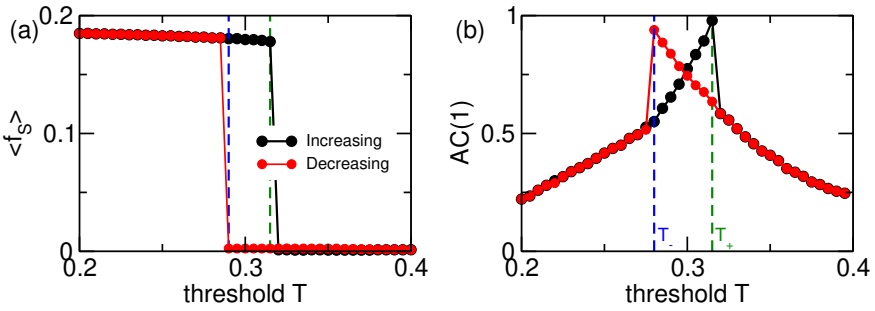


FIG. 3. Evolution of the order parameter $\langle f_S \rangle$ and its autocorrelation $AC(1)$ as a function of the control parameter T in the GH model. Panel (a): fraction of active neurons vs. threshold T. Panel (b): First autocorrelation coefficient $AC(1)$ vs. threshold T. In both panels the transition points, T_- and T_+, are marked with vertical dashed lines. Simulations were performed in a network with $\langle k \rangle = 30$ and $\pi = 0.6$. Simulations started at $T_0 = 0$ and neurons in a random state. T was slowly increased by $\Delta T = 0.0025$ every 10^5 time steps up to $T_F = 0.9$, then it was decreased back to T_0 in the same way.

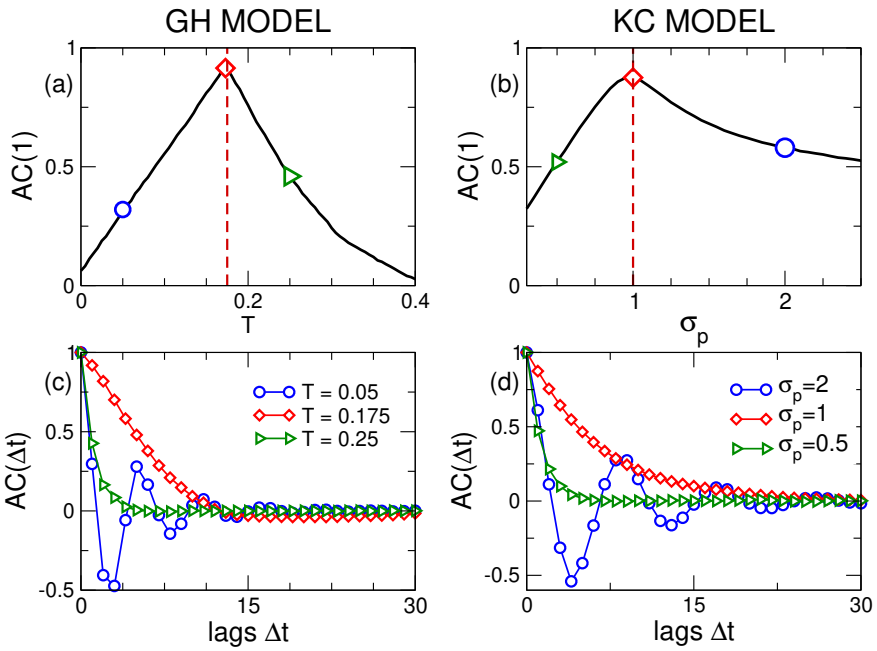


FIG. 4. Behavior of the first autocorrelation coefficient and the autocorrelation function for network parameters resulting in a continuous phase transition. Top panels: first autocorrelation coefficient AC(1) vs. the control parameter, T for the GH model (panel a) and σ_p for the KC model (panel b). Bottom panels: Autocorrelation function $AC(\Delta t)$ vs. time lag Δt, for three values of the control parameter, in the supercritical, critical, and subcritical phases, for the GH model (panel c) and the KC model (panel d). Dotted lines in panels (a) and (b) denote the critical point and symbols indicate the values of the control parameters used to compute the data in panels (c) and (d). Network parameters: $\langle k \rangle = 10$, $\pi = 0.6$.

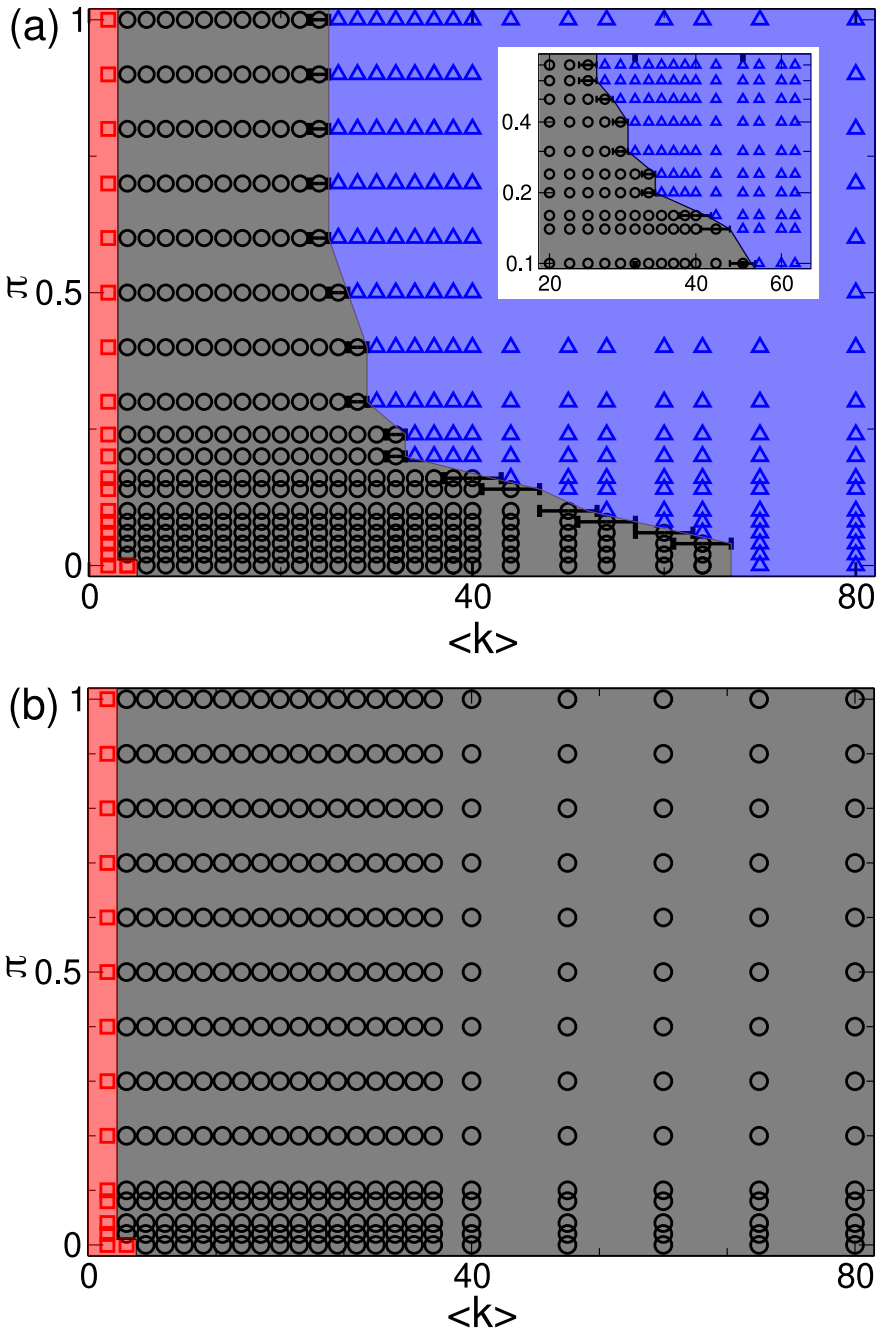


FIG. 5. Classes of collective dynamics emerging at different network topologies for both models (GH model in panel a, KC model in b). The graphs summarize the dynamical regimes observed for a wide range of network parameters (average connectivity $\langle k \rangle$ and degree of disorder π). Blue regions (triangles denote the values tested) indicate those values for which the networks exhibited discontinuous phase transitions to collective oscillations. Black regions (with circles) indicate continuous phase transitions, and red regions (with squares) transient dynamics to inactivity without a phase transition. The inset in (a) depicts a portion of the same data plot in the main panel in log-log scale.

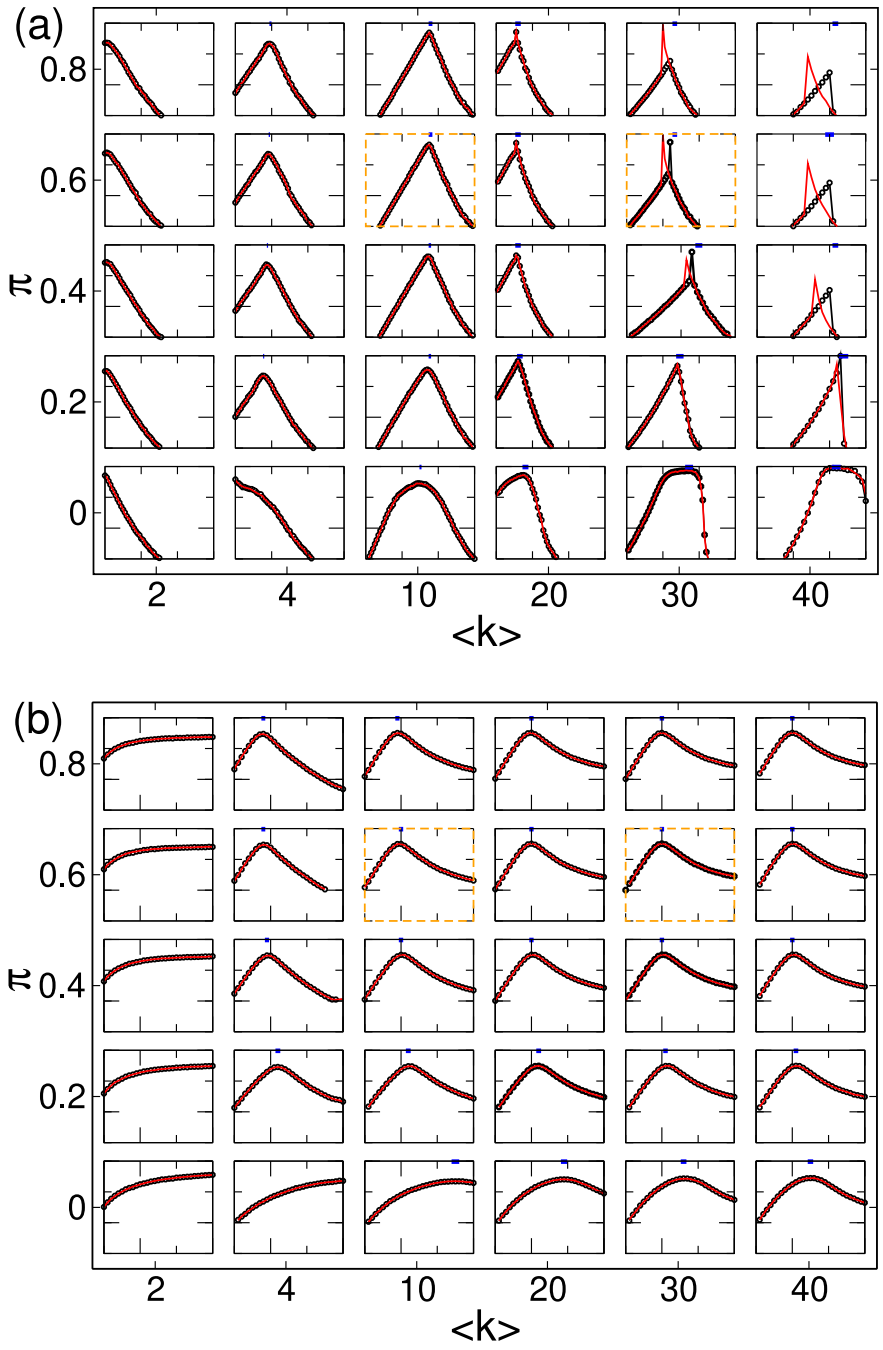


FIG. 6. Representative examples of the typical behavior of AC(1) as a function of the control parameter for diverse topologies in both models (GH model in panel a, KC model in b). Simulations where run for $5 \cdot 10^4$ time steps for each value of T or σ_p, increasing (black circles) or decreasing (red lines) by $\Delta T = 0.005$ or $\Delta\sigma = 0.05$. For the GH model, x-axis range is $[0 : 0.3]$, for $\langle k \rangle = 2$, 4 and 10, and $[0.2 : 0.5]$ for other $\langle k \rangle$ values. For the KC model, x-axis range is $\sigma_p = 0.5$ to $\sigma_p = 2$. y-axis range is $AC(1) = 0.25$ to $AC(1) = 1$ for both models. A thick blue dash over the upper x-axis marks a $\pm 3\%$ range of the control parameter, about it's critical value (or T_+ for discontinuous transitions). The boxes remarked with dashed lines correspond to the parameters used in Figs. 2-3-4.

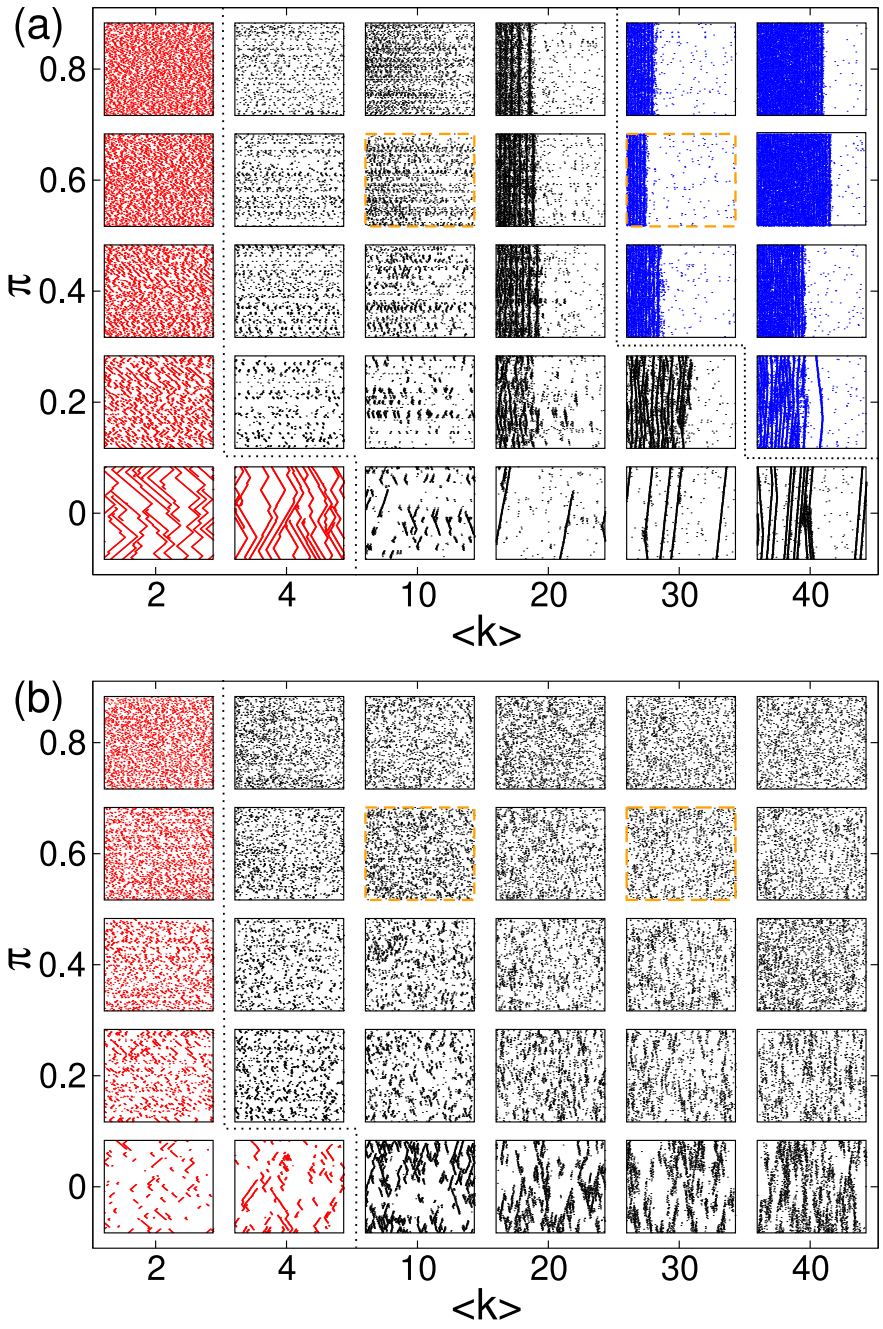


FIG. 7. Typical rasters of activity for networks with diverse topologies, while varying the control parameter around the critical value, for both models (GH model in panel a, KC model in b). The dots in each box denote activity of a subset of 300 neurons (ordered in the y axis), for 300 time steps (x axis), as the control parameter is changed continuously about the critical point. Other network parameters as in Fig. 6. Dot color matches the regions of Fig. 5. The dotted black lines indicate the limits of the parameters that exhibit phase transitions. The boxes remarked with dashed lines correspond to the parameters used in Figs. 2-3-4.

Supplemental material for: "Similar local neuronal dynamics may lead to different collective behavior" by Sánchez Díaz *et al.*

In the first part of this Supplemental Material, we show complements to the results on the main text.

In the second part, we comment on the implementation of the models used to generate the results in the main text. The respective codes in Fortran 90 and Python 3 can be found on GitHub [1].

I. COMPLEMENTARY RESULTS TO MAIN TEXT

A. Equivalence of p and σ_p as the control parameters for KC model

To avoid potential confusion on the role of σ_p as a control parameter, we would like to recall that it is proportional to p (i.e., $\sigma_p := p \cdot (\langle k \rangle - 1)/2$), so that KC dynamics does not depend on whether p or σ_p are chosen control parameter. In Fig. S1 we show AC(1) curves as a function of p and σ_p, showing that no qualitative difference is found when one or the other are used.

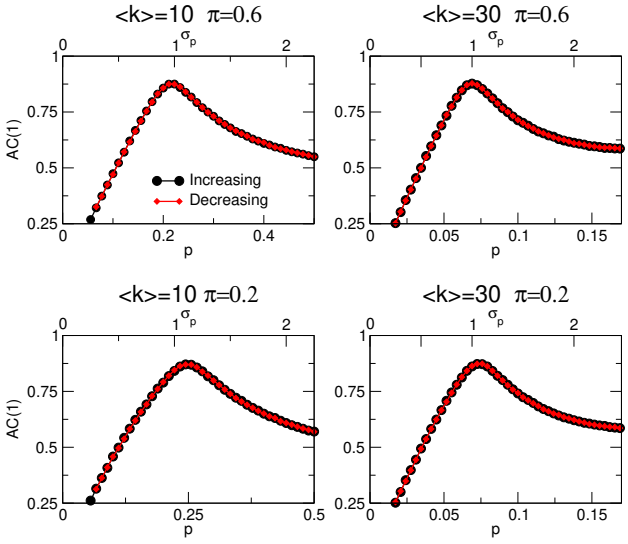


FIG. S1. Equivalence of the dynamics of KC model when expressed as a function of control parameter p or $\sigma_p = p \cdot (\langle k \rangle - 1)/2$. The figure shows AC(1) as a function of p as it is increased or decreased for four different network topologies. Upper $x-$axis shows the corresponding value of σ_p. The values of $\langle k \rangle$ and π for each network topology are shown on each panel heading. All other parameters are as in Fig. 4 of main text.

B. Results for KC model on the same conditions as Fig. 3

In Fig. S2 we show $\langle f_S \rangle$ and $AC(1)$ for KC model as σ_p is varied, for the same network topology as in Fig. 3 on main text (i.e., $\langle k \rangle = 30$ and $\pi = 0.6$), for 2 values of $\Delta\sigma_p$. Even for the smallest value of $\Delta\sigma_p$, no sign of discontinuity is observed.

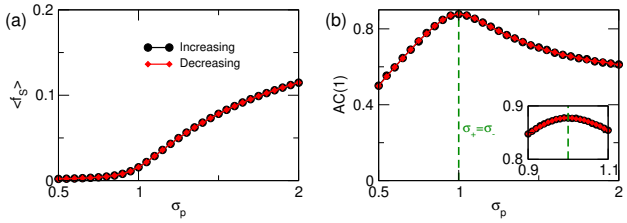


FIG. S2. Evolution of the order parameter f_S and its autocorrelation $AC(1)$ as a function of the control parameter σ_p in the KC model, for the same network as Fig. 3 on main text. Panel (a): fraction of active neurons as a function of σ_p. Panel (b): First autocorrelation coefficient $AC(1)$ as a function of σ_p. The value of $\sigma_- = \sigma_+$, is marked with a vertical dashed line. The inset in panel (b) shows $AC(1)$ using $\Delta\sigma = 0.01$ and 10^6 time steps for each σ_p value.

C. A stochastic postsynaptic model

Apart from the stochastic presynaptic KC model and the deterministic postsynaptic GH model, we have studied a modification of GH model where *transmitted activation* is stochastic. In this *stochastic postsynaptic* model, each neuron has the same three possible discrete states as in GH model, and transition probabilities are given by:

$$P_{i,0\to1} = 1 - [1 - r_1]\left[1 - S_q\left(\sum_{j=1}^{k_{in,i}} W_{ji}\delta_{S_j,1} - T\right)\right], \tag{1a}$$

$$P_{i,1\to2} = 1, \tag{1b}$$

$$P_{i,2\to0} = r_2. \tag{1c}$$

Notice that the only difference with respect to the GH model is in the *transmitted activation* rule, where the Heaviside function is replaced by a Sigmoidal function of the same arguments and stiffness q: $S_q(x) = \frac{1}{1+exp(q\cdot x)}$. GH model is recovered in the limit $q \to \infty$. Results for the *stochastic postsynaptic* for $\langle k \rangle = 30$ and $\pi = 0.6$ are shown in Fig. S3. In panel (a), we show AC(1) results

as T is increased and decreased for $q = 100$, while the same results for $q = 20$ are shown in panel (b). In one case the transition is discontinuous while in the other it is continuous.

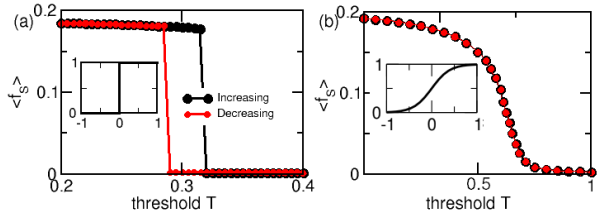


FIG. S3. Evolution of the order parameter $\langle f_s \rangle$ as a function of the control parameter T, for the *stochastic postsynaptic* model. Panel (a): results for a network with $\langle k \rangle = 30$, $\pi = 0.6$, using $q = 100$. Panel (b): results for the same network, but now with $q = 20$. The inset in each panel shows the respective activation probability function $S_q(x)$ used in Eq. (1a).

D. Simplified Mean Field Models

We study simplified versions of GH and KC models, where the dynamics can be solved analytically. They are defined in the mean field limit ($k = N$), where there is no refractory state (i.e., neurons can only be at quiescent or active state), no spontaneous activity (i.e., $r_1 = 0$) and all connections have the same weight W. Since the value of W does not change the behavior, we will use $W = 1/2$, so that definitions coincide with the KC model.

1. *Simplified GH model*

For simplified GH model, taking the equations (1a) and (1b) of the main text, it is straightforward to compute the evolution of the fraction of spiking neurons. We find that if there are $N_S(t) = N \cdot f_S$ spiking neurons (and $N - N_S(t)$ quiescent neurons) at time t, then at $t + 1$ there will be $N_S(t + 1) = N - N_S(t)$ spiking neurons (i.e., $f_S(t + 1) = 1 - f_S(t)$) if $N_S \cdot W > T$ and zero otherwise.

As a consequence, for $T > N \cdot W/2$ the only solution is $N_S = 0$ ($f_S = 0$). For larger values of T, there may be several solutions depending on the initial condition $N_S(t = 0)$: *no activity*, $N_S(t > 2) = 0$ (or equivalently $f_S = 0$) if $N_S(t = 0) < T/W$ or $N - T/W < N_S(t = 0)$; *stationary* if $N_S(t = 0) = N/2$ ($f_S = 1/2$) or *oscillatory* with period 2, from $N_S(t = 0)$ to $N - N_S(t = 0)$ (or from $f_S(t = 0)$ to $1 - f_S(t = 0)$) for other cases. Remarkably, this solutions are stable: a perturbation (i.e., externally turning a single quiescent neuron active or doing the opposite) does not change the kind of solution. The *no activity* solution will be recovered one step after a perturbation, while the result of the perturbation on a *oscillatory* solution will be a slight modification of the oscillation amplitude.

2. *Simplified KC model*

For simplified KC model, intuitively, we have 3 cases: for σ_p smaller than 1, the activity will tend to decrease independently of the initial value. For $\sigma_p = 1$, the fraction of spiking neurons will tend to be constant, while for $\sigma_p > 1$, we expect the activity to grow until it is bounded by (a fraction of the) total number of available neurons. The evolution equation for this simplified case (considering the availability of quiescent neurons), can be easily derived from Eqs. (2a)-(2b) from main text. It reads:

$$f_S^*(t + 1) = (1 - f_S(t)) \times \left(1 - (1 - p \cdot W)^{N \cdot f_S(t)}\right), \quad (2)$$

where $f_S^*(t+1)$ is the expected value of f_S at time $t+1$, for a given value of f_S. $f_S^*(t+1)$ is concave in $f_S(t)$ (proof is straightforward[2]). Because of that, there may be 1 or 2 stationary solutions (i.e., solutions of $f_S^*(t+1) = f_S(t)$). The first solution is $f_S = 0$ always. Since $f_S^*(t + 1)$ is concave on f_S, the condition for having two crossings of the curves (i.e., one more solution) is that $f_S^*(t + 1)$ has to be greater than f_S at least very close to zero (i.e., for one neuron, $f_S = 1/N$). But this is also the condition for which solution $f_S = 0$ is unstable, see Fig. S4. This means that, if there are two solutions, the first solution $f_S = 0$ is unstable. It is straightforward to show that the stability limit of $f_S = 0$ reads $p \cdot W \times (N - 1) = 1$ (i.e., $\sigma_p = 1$).

For very large values of $\sigma_p \gg 1$ (i.e., away from the critical transition) the previously stable solution may also destabilize, and oscillating dynamics may appear. This highly supercritical regime is not considered in the present manuscript.

In summary, for the simplified KC model, the value of σ_p prescribes at most one stable solution.

Despite lacking important ingredients of the original models, such as the spatial dependance or the quenched disorder, this models retain some of the main characteristics of the KC and GH presented on main text: the simplified GH model is deterministic and has a postsynaptic activation rule. As a consequence, the trend of the activity (i.e., of f_S at time $t + 1$) to grow or to decrease is not only determined by the control parameter T, but also by the state of the network at time t. This implies that different initial conditions will lead to different solutions, for the same value of the control parameter T. On the contrary, for the probabilistic, presynaptic, simplified KC model, the evolution of f_S is rather insensitive to the state of the network, and is governed by σ_p.

II. PROGRAM CODE IMPLEMENTATION

A. Implementation of the Watts-Strogatz Network

In Fortran 90, the code `Matrix.f90` is used to generate an undirected Watts-Strogatz network with parameters N, π, and $\langle k\rangle$ which is saved as a two columns ascii file `MyMatrix.txt` of length $N\cdot\langle k\rangle/2$. Each row has two integer numbers, i and j, which denote a symmetric edge between nodes i and j.

For Python, the Watts-Strogatz network is generated using the available code from the `networkx` package.

B. Implementation of the Greenberg & Hastings (GH) Model

1. *Fortran 90 code*

The dynamics of the GH model, given by Eqs. 1 (a)-(c) of the main text, is implemented by the program `GH_model.f90`.

The main code reads the file `MyMatrix.txt` (generated as explained in the previous section), and assigns weights to each connection according to an exponential distribution as explained in the main text. For each value of the control parameter T, it runs the `ConstantStep` subroutine for a given number of steps (typically 50.000), updating synchronously the network state using the `Step` subroutine. At the end of the `ConstantStep` subroutine, it computes the order parameter f_S and the autocorrelation coefficient $AC(1)$. At each time step t the `Step` subroutine computes the state of all neurons at time $t+1$, $S(t+1)$, from $S(t)$. For each neuron i:

1. If the neuron i was *quiescent* (i.e., $S_i(t)=0$), it will become *active* $S_i(t+1)=1$ with probability r_1. If that does not happen, a loop over all j connected to i is made. For each active j (i.e., $S_j(t)=1$), the weight W_{ij} is summed up. If the sum is larger than T, then $S_i(t+1)=1$.

2. If the neuron was *active*, (i.e., $S_i(t)=1$), it will become *refractory* (i.e., $S_i(t+1)=2$) always.

3. If the neuron was *refractory*, (i.e., $S_i(t)=2$), it will become *quiescent* (i.e., $S_i(t+1)=0$) with probability r_2.

All neurons are updated synchronously.

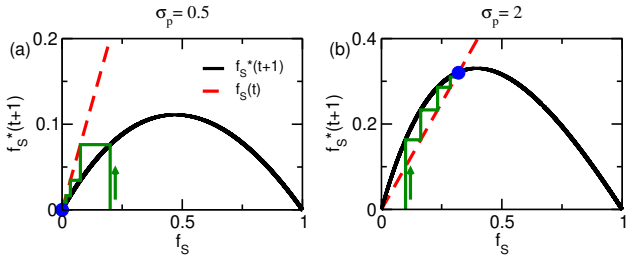


FIG. S4. Expected fraction of active neurons at time $t+1$, $f_S^*(t+1)$ as a function of $f_S(t)$, for the simplified KC model. Panel (a): example case for $\sigma_p<1$. If $\sigma_p<1$, the only solution is *no activity*: $f_S=0$. Panel (b) example case for $\sigma_p>1$, the solution $f_S=0$ is unstable and another solution exists. In both panels, red dashed line is the identity curve, a green line shows the evolution of f_S for a few time steps, and a blue circle shows the stable solution. Results for $N=20000$, $W=0.5$, $k=N$.

2. *Python 3 code*

The code `GH_model.py` in Python 3 follows the same steps and uses the same names for variables as in the Fortran 90 code. The code requires the following packages: `numpy`, `random` and `networkx`.

The neural activity as a function of time is saved on files named `actN(N)K(K)PI(100*pi)T(1000*T).txt`, where the value of π is multiplied by 100 and the value of T is multiplied by 1000, so that they are represented by integer numbers. For instance, for parameters $N=5000$, $\langle k\rangle=10$, $\pi=0.6$ and $T=0.2$ the output file will be named `actN5000K10PI60T200.txt`

C. Implementation of the Kinouchi & Copelli (KC) Model

In the following, we describe the implementation of the code for the KC model used in the main manuscript. Two possible variations of the model are described later in section II D. The KC code follows the same structure as the GH counterpart, starting by reading the file `MyMatrix.txt` but changing a few relevant parts: the `Step` subroutine is obviously different and the program cycles through values of σ_p instead of T.

Here we call KC_m to the code implementation used for the results described in the main text, which follows Eqs. 2 (a)-(d). The codes are `KC_m.f90` and `KC_m.py` for Fortran 90 and Python 3 respectively.

At each time step t the `Step` subroutine computes the state of all neurons at time $t+1$, $S(t+1)$, from $S(t)$. For each neuron the `Step` subroutine updates the states as follows:

1. For each active neuron j (i.e., $S_j(t)=1$) a loop over all output connections is made. Each *quiescent* neuron i (i.e., $S_i(t)=0$), will become *active* with probability pW_{ji}.

2. If the neuron j was *quiescent* (i.e., $S_j(t)=0$), it will become *active* ($S_j(t+1)=1$) with probability r_1.

3. If the neuron j was *active*, (i.e., $S_j(t) = 1$), it will become *refractory* (i.e., $S_j(t+1) = 2$) always.

4. If the neuron j was *refractory*, in state m, (i.e., $S_j(t) = m > 1$), it will become *refractory* with state $m+1$ (i.e., $S_j(t+1) = m+1$) unless $m = n$. In that case, if becomes *quiescent* (i.e., $S_j(t+1) = 0$).

All neurons are updated simultaneously. Within this implementation, it may happen that, in the same time step more than one active neuron activates the same quiescent neuron.

D. Two alternative implementations of the KC model.

As mentioned in the previous section, we explored some algorithmic alternatives for the implementation of the KC model. We will describe two of them here.

1. *Backward update implementation*

What we call the *backward* implementation (see codes `KC_b.f90` and `KC_b.py`) is described also by Eqs. 2 (a)-(d) of the main text. The only subtle difference with KC_m is in the activation loop within the `Step` subroutine: now it is performed not over the *active* neurons but over the *quiescent* neurons. Therefore, in this implementation the `Step` subroutine is:

1. If neuron i was *quiescent* (i.e., $S_i(t) = 0$), a loop over all input connections j is made. For each *active* neuron j (i.e., $S_j(t) = 1$), an attempt to *activate* i with probability pW_{ji} is made.

2. If the neuron j was *quiescent* (i.e., $S_j(t) = 0$), it will become *active* ($S_j(t+1) = 1$) with probability r_1.

3. If the neuron j was *active*, (i.e., $S_j(t) = 1$), it will become *refractory* (i.e., $S_j(t+1) = 2$) always.

4. If the neuron j was *refractory*, in state m, (i.e., $S_j(t) = m > 1$), it will become *refractory* with state $m+1$ (i.e., $S_j(t+1) = m+1$) unless $m = n$. In that case, if becomes *quiescent* (i.e., $S_j(t+1) = 0$).

Similar to the KC_m implementation, it may happen that neuron i is simultaneously activated by several j's.

An example of f_S as a function of σ_p computed with both KC_b and KC_m implementations in Fortran 90, is shown in Fig. S5 (a). The results are identical, nevertheless, one implementation may be considerably faster, due to the fact that, depending on the case, all or only the active neurons need to be updated. For instance, using `gfortran`, compiled with `-O3` flag, the KC_m implementation takes about 2/3 of the time required by the KC_b implementation to reproduce Fig. S5 (a).

2. *Local out-degree implementation*

We have also explored another implementation of the KC model, we called KC_l, by slightly modifying its activation rule. In this case, in each update, the activation probabilities of each neuron depend on its out-degree (and the state of its target neurons). Specifically, a given active neuron j counts how many connected quiescent neurons has at time t (i.e., $\sum_{m=1}^{k_{out,j}} \delta_{S_m,0}$), and re-normalizes its contribution among them.

All the other transition rules and parameters are the same as in the KC_m model. In this case the `Step` routine reads:

1. For each active neuron j (i.e., $S_j(t) = 1$) a loop over all output connections is made. The number of i neurons connected with j that are *quiescent* is counted and defined as NC. Each *quiescent* neuron i, will become *active* with probability $2\sigma_p W_{ij}/NC$.

2. If the neuron j was *quiescent* (i.e., $S_j(t) = 0$), it will become *active* ($S_j(t+1) = 1$) with probability r_1,

3. If the neuron j was *active*, (i.e., $S_j(t) = 1$), it will become *refractory* (i.e., $S_j(t+1) = 2$) always.

4. If the neuron j was *refractory*, in state m, (i.e., $S_j(t) = m > 1$), it will become *refractory* with state $m+1$ (i.e., $S_j(t+1) = m+1$) unless $m = n$. In that case, if becomes *quiescent* (i.e., $S_j(t+1) = 0$).

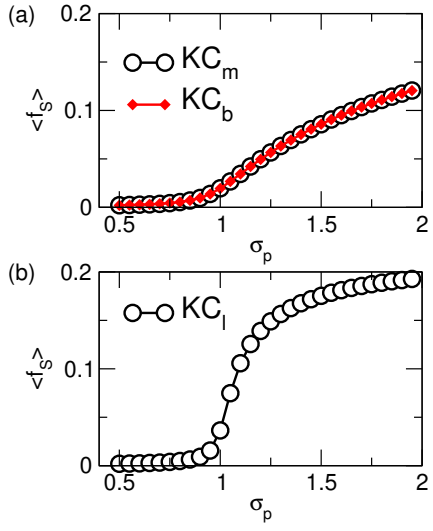


FIG. S5. Comparison of alternative implementations of the KC model. The fraction of active neurons, f_S as a function of the order parameter σ_p, using the KC_m (black circles), and the KC_b (red diamonds), models are shown in (a). KC_l results are shown in (b). In all the cases, the Fortran 90 codes, with $5 \cdot 10^4$ time steps and $\Delta\sigma_p = 0.05$ were used. A Watts-Strogatz network with parameters: $N = 20000$, $\langle k \rangle = 10$, $\pi = 0.6$ was used.

KC_l differs from $KC_{m/b}$ in that, in KC_l, the probability that a given neuron j activates neuron i depends of its degree (the number of connections of j), and the state of all other output connections. This introduces differences that deserve to be further explored. An example of f_S as a function σ_p, computed with Fortran 90 codes is shown in Fig. S5 (b). Further results for $AC(1)$ as a function of σ_p for KC_l (similar to Fig. 6-(b) on the main text), are shown in Fig. S6. Similar to KC_m, KC_l model does not present discontinuous transitions: for $\langle k \rangle = 2$, there are *transients to no-activity*, while for larger values of $\langle k \rangle$, there are *continuous phase transitions*. Notice that for this model the $AC(1)$ decays relatively faster for σ_p values away from the critical value $\sigma_p = 1$ (compare with Fig. 6 on main text).

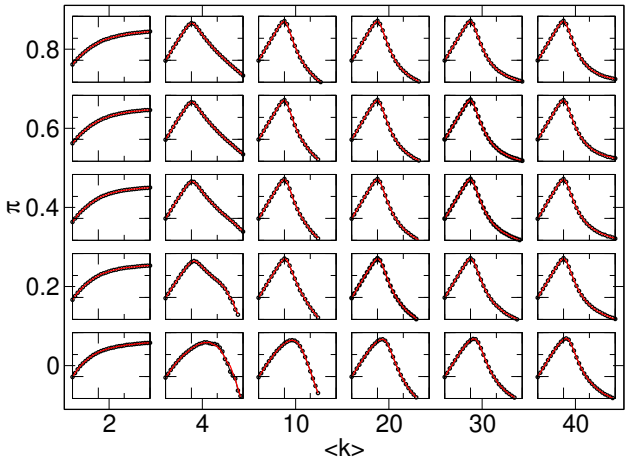


FIG. S6. Results obtained with the alternative KC_l described in section II D. The data shows representative examples of the typical behavior of $AC(1)$ as a function of σ_p for diverse topologies. Simulations were run increasing (black circles) or decreasing (red lines) σ_p values by $\Delta\sigma = 0.05$ every $5 \cdot 10^4$ time steps. In each box, the x-axis range is $\sigma_p = 0.5$ to $\sigma_p = 2$ and the y-axis range is $AC(1) = 0.25$ to $AC(1) = 1$.

---

[1] The Fortran 90 and Python 3 codes are deposited in github (https://github.com/DanielAlejandroMartin/Apparently-Similar-Neuronal).

[2] Consider $f_S^*(t+1) = gh$ where $g = 1 - (1-p)^{N \cdot f_S}$ and $h = 1 - f_S$. We will denote h' the derivative of h with respect to f_S. It is straightforward to find $h' < 0$, $h'' = 0$, $g' = -(1 - p \cdot W)^{N f_S} \times ln(1 - p \cdot W) > 0$, then, $g'' = -(1 - p \cdot W)^{N \cdot f_S} \times [ln(1 - p \cdot W)]^2 < 0$, and finally $f_S^{*\prime\prime} = g''h - 2g' < 0$