Predicting Phase Transitions: Investigating the Quality of Mean Field Approximations in a Random Graph, Small-World Graph and a Scale-Free Graph

Jolanda J. Kossakowski · Lourens J. Waldorp

Received: July 17, 2015

Abstract A phase transition is defined as a qualitative change resulting from a small perturbation; a concept thought to exist in constructs like depression or in learning cognitive abilities. Phase transitions can be located with a Mean Field Approximation (MFA), which assumes that the network is a toroidal grid. In this study, we investigated the quality of the MFA when the network is not a toroidal grid, but a Random Graph (RG), Small-World Graph (SWG), or a Scale-Free Graph (SFG). We simulated one million time points for various configurations for each network. First-order phase transitions were found in RGs, and second-order phase transitions in the SFG. Furthermore, chaos was found in the SWG and SFG. Results show that the MFA can accurately detect and predict phase transitions in all networks, even though phase transitions were not observed. The MFA enables educators or clinicians to anticipate phase transitions and adapt their strategy.

Keywords Mean Field Approximation · Probabilistic Cellular Automata · Network Analysis

1 Introduction

Phase transitions or bifurcations (Kuznetsov, 2013) are qualitative changes that result from a small perturbation to a system of mutually interacting variables that together form a network (Cramer, 2013; Olami et al., 1992; Sethna, 2006). Phase transitions can be divided into two types: first-order phase transitions, which are phase transitions that involve a discontinuity and thus a jump between phases, and second-order phase transitions: phase transitions in which a system steadily transitions from one stable phase to the other stable phase (Jaeger, 1998). Figure 1 depicts simplified examples of these types of phase transitions, as well as an example of an unstable system, where the network is at a critical point at which a first-order phase transition may occur at any moment (Scheffer et al., 2009), and an example of a stable system in which no phase transitions occur. Next to these possibilities, networks can also be chaotic and behave randomly. These networks are sensitive to small perturbations (Tabor and Weiss, 1981) and will not transition to a stable phase. The evaporation of water, the onset of major depressive disorder, or learning a cognitive developmental task may be seen as examples of phase transitions.

Constructs like major depressive disorder can be seen as a network of interacting symptoms. Where traditional approaches assume that depression is a latent variable causing variation in observable symptoms of depression (Kossakowski et al., 2015), the network approach does not make this assumption, but instead allows symptoms to interact directly, without the interference of a latent variable (Cramer et al., 2010). Using dynamic networks in predicting phase transitions, networks in which the state of nodes or connections between nodes change over time (Gulyás et al., 2013), we gain not only more insight into constructs like major depressive disorder, but we also gain insight into which combination of variables drives the onset of a first-order phase transition.

Furthermore, predicting first-order phase transitions may not only aid psychiatrists and psychologists in their treatment of patients with psychological disorders like major depressive disorder, but may also contribute in the anticipation of an epidemic (van Borkulo et al., 2015), or help in adjusting one’s educational approach in anticipation of a de-
developmental transition (Hartelman et al., 1998). Developing a method that predicts the possibility of phase transitions is of great societal and scientific importance, as predicting first-order phase transitions provides important insights in the self-organising behaviour that specific network structures may have. The goal of the present paper is to make a first step into the development of a method with which we can describe first-order phase transitions.

First-order phase transitions may be explained with percolation theory (Stauffer and Aharony, 2003). Percolation theory states that, in the long run, a system will be completely in one phase once the system has surpassed some critical value (van Borkulo et al., 2015). In other words, when some network parameter, exceeds a critical value, we can expect all the nodes in the network to become active, with the result that the network jumps to the other phase and experiences a first-order phase transition.

Network structures that may contain these first-order phase transitions can be seen as Cellular Automata; complex systems of networks, where nodes lay on a finite Toroidal Grid (torus), and in which the state of individual nodes at a time point is a function of the states of that node’s neighbourhood (Markov Blanket; MB) at the previous time point (Wolfram, 1984). A cellular automaton becomes probabilistic (PCA) when the state of a node at time point \( t + 1 \) has a certain probability that depends on that node’s MB at the previous time point \( t \) (Balister et al., 2006). The state of nodes at \( t + 1 \) can be determined with a Mean Field Approximation (MFA), an approximation method developed for tori, where it is assumed that each node’s MB is similar in size (Balister et al., 2006). In other words, the network has to be similar at all locations and be spatially invariant. Because of this assumption, the probability for a node to have a certain state in a PCA therefore only depends on the number of active nodes in its MB.

The MFA was originally developed for a torus. Let \( G = (V, E) \) be a network with \( V \) being the set of nodes \( \{1, 2, \ldots, p\} \) and \( E \) the set of edges that connect two nodes with each other (Koller and Friedman, 2009). Edges denote a reciprocal relation between two nodes, making the network a Markov Random Field (MRF). A network becomes a torus when each node is connected to exactly four other nodes. Moreover, a node’s MB lay in a diamond around the node itself, as can be seen in Figure 2, where the left figure represent half a torus in three-dimensional space (Rickert, 2014), and the right a two-dimensional close-up. The grey nodes and edges represent the diamond that is the MB of the middle node. Note that the outer nodes are connected to their opposite nodes (e.g., the node in the upper left corner is also connected to the node in the upper right corner), but that this is not visible due to the two-dimensionality of Figure 2.

Tori like Figure 2, follow the Ising Model (Ising, 1925). The Ising model arose in statistical physics and was developed to model ferromagnetic materials, like the configuration of atoms and their corresponding spins. In an Ising Model, all nodes are binary and in either of two states (e.g., \{+1, -1\} or \{active, inactive\}). An Ising model consists of

![Fig. 2: Visualization of half a torus (left figure), and a close-up that shows the grid structure (right figure).](image-url)
cliques (a subset of nodes in a graph that is fully connected) that have solely one or two nodes in the case of a torus (Waldorp, 2015b). Each node in an Ising model has a probability to be in a certain state, and by having weighted connections with nodes that are in the other state, a node can switch states (Kindermann and Snell, 1980).

Kozma et al. (2005) showed that the MFA works well in tori that contain long-range connections; an adaption that is appropriate for neural data. It remains questionable whether real life phenomena like the onset of major depressive disorder or a developmental transition can be captured in a torus like Figure 2. More realistic network structures that better represent these real-life phenomena include a Random Graph (RG; Erdös and Rényi, 1959), a Small-World Graph (SWG; Watts and Strogatz, 1998) and a Scale-Free Graph (SFG; Barabási and Albert, 1999). Examples of these network structures are found in Figure 3. The RG, SWG and SFG each represent a class of network structures (Kolaczyk, 2009), in which the RG, SWG and SFG themselves are the most generic network structures. The RG, SWG and SFG have been chosen for the current study as these network structures are among the most commonly used network structures that have also been investigated in detail over the past few years.

The RG is a network model in which all edges between nodes are formed with an independent, constant probability $p_{\text{edge}}$ (Gilbert, 1959; Erdös and Rényi, 1960). An advantage of the RG is that it is a simple network model that has been thoroughly investigated with regard to various properties (e.g., Erdös and Rényi, 1964; Callaway et al., 2000). Although the RG may not be suitable for the representation of real-world networks (Newman et al., 2002), RGs are often used as a baseline against which observed networks can be compared. With respect to first-order phase transitions, analytical results showed that the critical point where the network is unstable and where first-order phase transitions may occur at any moment shifts to a higher value of $p_{\text{active|MB}}$ as the probability for an edge to form increases (Waldorp, 2015a).

The SWG is a network model in which the average distance between two nodes resembles that of an RG, and where the averaged ratio between the amount of existing edges and possible edges is high compared to an RG (Watts and Strogatz, 1998). In contrast to the RG, the SWG is found in real-world networks, like social networks (Milgram, 1967) and co-author networks (Jackson, 2008). With respect to first-order phase transitions, analytical results showed that the critical point up until which first-order phase transitions can occur, shifts to a lower value of $p_{\text{active|MB}}$ as the probability for an edge to be rewired increases (Waldorp, 2015a).

Lastly, the SFG is a network model in which the connectivity distribution has a power-law form (Barabási and Albert, 1999). This results in a network in which a lot of nodes have few edges, and a few nodes have a lot of edges. Real-world examples of SFGs include the internet (Pastor-Satorras and Vespignani, 2001; Barabási et al., 2000), the cell (Barabási, 2009) and even the brain (Eguíluz et al., 2005). An advantage of the SFG is that it not only can describe the birth and death of nodes and edges (Barabási et al., 2000; Barabási, 2009), but that it is robust against the death of random nodes.

The goal of the present study is to determine the quality of the MFA’s prediction of first-order phase transitions in dynamic networks. By using the MFA, we aim to draw con-
clusions about whether a network may experience first-order phase transitions.

We identify and map the parameter space at which first-order phase transitions are imminent for network structures other than tori. Furthermore, we expect that the area in which first-order phase transitions may occur is a function of either the size of a node’s MB (torus), or the parameter that identifies the network structure (RG, SWG, SFG).

This paper is organised as follows. First, we give a description of the simulation study that has been performed, after which we describe the network structures in more detail. We then proceed with the results of the simulation for a torus, an RG, SWG and an SFG, respectively.

2 Methods

In this study, we investigate the robustness of the MFA when network structures other than a torus are under investigation. We first describe how the data are simulated. Second, we assess the quality of the MFA when non-standard network structures are used by investigating the density, that is, the amount of active neighbours proportional to the total amount of neighbours a node has. Third, we define the network structures used to simulate data in relation to the MFA.

2.1 Characteristics

To simulate data, we used either an RG (Erdős and Rényi, 1959), an SWG (Watts and Strogatz, 1998), or an SFG (Barabási and Albert, 1999) to simulate data of an Ising model. We also simulated data of an Ising model on a torus to compare the results of the non-standard network structures with the results of the standard network structure that is typically used in an MFA. The probability for a node to become active given its MB \( p_{MB} \) (all graphs), the probability for an edge to be rewired \( p_{rewire} \) (SWG) and power-law parameter \( \gamma \) (SFG) were varied in order to obtain a parameter space for each network structure where we map the area in which phase transitions occur. To determine whether first-order phase transitions occur, we measure the network’s density at each time point; the ratio of active nodes in a network compared to the total amount of nodes in a network.

Parameters \( p_{activeMB} \), \( p_{rewire} \) and \( p_{edge} \) are set to \{0.10, 0.15, 0.20, 0.25, 0.30\}, and the \( \gamma \) parameter is set to \{1.00, 1.25, 1.50, 1.75, 2.00\}. The amount of nodes \( n \) is set to 64; an amount that is both found in realistic networks and is a quadratic term, which enables the construction of tori. We ran the simulations with \( T = 10,000 \) time points and \( T = 1,000,000 \) time points. In this study, we only simulated MRFs, which are unweighted, binary networks: nodes can either be active (’1’) or inactive (’0’). Moreover, edges in networks depict a reciprocal relation between two nodes, without taking the strength of the relation into account.

2.2 Design

A total of 16 networks were constructed: one torus, and five RGs, SWGs and SFGs, respectively. Each of these network structures was defined according to either \( p_{edge} \) (RG), \( p_{rewire} \) (SWG) or \( \gamma \) (SFG). For each network, we determined the MB of individual nodes. To initialise the MFA, a random number of nodes were activated and the resulting density was calculated. Nodes had a probability to become active at time point \( t + 1 \) according to the majority rule

\[
p_{activeMB} = \begin{cases} 
    p + h & r < |\Gamma|/2 \\
    1 - p + h & r > |\Gamma|/2
  \end{cases}
\]

where \( p \) is the probability that is a priori fixed, and \( h = 0 \), which is also a priori fixed. A node’s active MB is given by \( r \) and \(|\Gamma|/2 \) denotes the total MB of a given node divided by two, hence the name ‘majority rule’: more than half of a node’s MB must be active in order for that specific node to become active with probability \( p \) at \( t + 1 \). The result of equation 1 is \( p_{activeMB} \), a vector of probabilities that is used to set up a new network with active nodes based on \( p_{activeMB} \), after which the density of that network is calculated. This sequence of steps is then repeated for each time point. Appendix I displays the code that was used to perform the simulation study.

2.3 Toroidal Grid

As previously stated, tori are network structures that display a regular pattern: each node is connected to exactly four neighbouring nodes, which lay in a diamond around a specific node. Equation 1 shows the majority rule that is used to determine \( p_{activeMB} \) at \( t + 1 \). In tori, this probability is homogeneous, and therefore we obtain a binomial distribution for amount of active nodes in a node’s MB \( N = 1 \)). This means that the probability \( \phi^{-1}(1) \) being equal to \( r \) can be represented with \(|\Gamma|\) Bernoulli trails that each have a success probability of \( \rho \), which is equal to the network’s density (Waldorp, 2015a). We can combine this knowledge with equation 1, and define the probability of the network’s density given the density at the previous time step \( \phi(p) \) as:

\[
p_{\phi}(p) = p \sum_{r=0}^{t-1/2} \binom{|\Gamma|}{r} p^r (1-p)^{|\Gamma|-r} + (1-p) \left( 1 - \sum_{r=0}^{t-1/2} \binom{|\Gamma|}{r} \phi^{-1}(1) p^r (1-p)^{|\Gamma|-r} \right)
\]
which is also used in Kozma et al. (2005).

2.4 Random Graph

Consider graph $G_{ER}(V,E)$, with $n$ nodes and $m$ edges. A total of $E$ edges is randomly chosen out of $\binom{n}{2}$ possible edges (Erdős and Rényi, 1960). An RG can be formulated as

$$G_{ER} = p^m q^{(\binom{n}{2})-m}$$

where $p = p_{edge}, q = 1 - p_{edge}$ (Bollobás, 2001). The probability that a random graph $G_{ER}$ is connected is $\exp(-\exp(-\lambda))$, where $p_{edge} = (\log n + \lambda + o(1))/n$ with $n$ being the amount of nodes in $V$, $\lambda$ fixed and $o(1)$ that is assumed to be 0 (Theorem 7.3; Bollobás, 2001).

Similar to the torus, RGs contain Binomial processes for the presence or absence of an edge. Because this process is independent for all edges, we can express the probability of the network’s density given the density at the previous time step $p_{\Phi}(p_t)$ as

$$p_{\Phi, RG}(p_t) = \sum_{r=0}^{n-1} p_{r,1} \binom{n-1}{r} (p_t p_{edge})^r (1-p_t p_{edge})^{n-r-1}$$

where $p_{r,1}$ denotes the probability of a node being active given that $r$ neighbours are also active and $n$ the amount of nodes. It can be seen that equation 4 contains an extra random variable in comparison to equation 2: in an RG, not only the density at $t$ determines the density at $t+1$, but $p_{edge}$ as well.

2.5 Small-World Graph

Consider graph $G_{SWG}(V,E)$, where we start out with $G$ being a torus that has $n$ nodes and $k$ edges per node. Each has a certain ‘rewiring probability’ $p_{\text{rewire}}$. When $p_{\text{rewire}} = 1$, the network structure becomes an RG (Watts, 1999). In the current study, we chose to create SWGs both by replacing edges by rewired edges, and by adding rewired edges (Monasson, 1999; Newman and Watts, 1999). Results of the two approaches will be compared to see whether results are a function of the chosen method.

The adjustment that has to be made to equation 2 in order for it to represent the probability of the network’s density given the density at the previous time step in an SWG is relatively small: next to the binomial process that is present in tori, a binomial process is added for the addition of edges as a result of the rewiring process:

$$p_{\Phi, SWG}(p_t) = (1-p) \sum_{r=0}^{\left\lfloor \frac{|G|}{2} \right\rfloor} \binom{|G|}{r} p_r^r (1-p_r)^{|G| - r}$$

$$+ p \left( 1 - \sum_{r=0}^{\left\lfloor \frac{|G|}{2} \right\rfloor} \binom{|G|}{r} p_r^r (1-p_r)^{|G| - r} \right)$$

$$+ \sum_{k=0}^{n} \sum_{r=0}^{k} p_{r,1} \binom{k}{r} (p_{p_{\text{rewire}}}^r (1-p_{p_{\text{rewire}}})^{k-r})$$

where $k$ denotes the size of a node’s MB. Because a node’s MB differs for each node, we need to take all possibilities into account and sum over all possible sizes of a node’s MB, which is denoted by the two summation signs in equation 5.

2.6 Scale-Free Graph

Consider graph $G_{SFG}(V,E)$, where at first only one node and no edges exist. Each node is then subsequently added to the graph, and with the birth of each node, edges between this node and the older nodes are created with the following probability:

$$p[i] \sim k[i]^{\gamma + 1},$$

where $i$ denotes an older node, and $k[i]$ the amount of edges of that older node that were not initiated by the birth of that specific node (Csardi and Nepusz, 2006).

The probability’s decrease of the connectivity distribution follows $p(m) \sim m^{-\gamma}$, with $1 < \gamma < 2$. When $\gamma$ becomes larger than 2, the mean degree, the average amount of neighbours that a node has, becomes infinite (Li et al., 2006), which makes it impossible to map the parameter space in which first-order phase transitions occur.

Adjusting equation 2 so that it is appropriate for the SFG is somewhat more complicated that it is for the RG and the SWG, because of the $\gamma$ parameter. The probability of the network’s density given the density at the previous time step is

$$p_{\Phi, SFG}(p_t) = \sum_{k=0}^{\left\lfloor \frac{n}{2} \right\rfloor} \sum_{r=0}^{k} p_{r,1} \binom{k}{r} p_r^r (1-p_r)^{k-r} k^{-\gamma}/Z_\gamma$$

where $k-rk^{-\gamma}/Z_\gamma$ denotes the probability for a node to be connected to another node with $k$ being the amount of edges. The $Z_\gamma$ parameter is the partition function and turns equation 7 into a probability function. Like equation 5, equation 7 also holds two summation functions for the very same reason: we need to account for all possible neighbourhood sizes.

All simulations and analyses are performed using the R statistical software 3.2.0 (R Development Core Team, 2014).
Initial structures of the RG and the SFG were constructed using the R-package igraph version 0.7.1 (Csardi and Nepusz, 2006) and the R-package qgraph version 1.3.1 (Epskamp et al., 2012). All code files can be requested with the first author.

3 Results

For clarity of presentation, results from the simulation study with $T = 1.000.000$ time points are shown only for selected configurations. We ran the simulation for two types of SWGs: one in which a random sample of edges were removed, rewired and added to the initial network structure, and one in which the rewired edges were added to the initial network structure. Results from the first method were omitted from the current study, as results were highly similar to results from SWGs constructed with the second method. Figures of configuration that are not presented in the current paper can be requested from the first author.

Figures 5 and 6 (left column) show the evolution of the density throughout the simulation for each network structure. Fragments of $T = 10.000$ are displayed for selected configurations. It can be seen in Figure 6a and 6d that first-order phase transitions were found in RGs with $p_{\text{edge}} = 0.25$ and $p_{\text{edge}} = 0.75$. Unfortunately, we did not observe first-order phase transitions in other network structures: stable network structures were observed for the torus (Figure 5a), for the RG with $p_{\text{edge}} \neq 0.25$ and $p_{\text{edge}} \neq 0.75$ (Figure 6a and 6d), for the SWG with $p_{\text{active}|MB} = \{0.10, 0.15, 0.20, 0.25, 0.30\}$ (Figure 5d) and for the SFG with $\gamma = 1$ (Figure 6j). We observed chaotic network structures for the SWG with $p_{\text{active}|MB} = \{0.80, 0.85, 0.90, 0.95, 0.97\}$ (Figure 5g) and for the SFG with $\gamma = \{1.25, 1.50, 1.75, 2.00\}$ (Figure 6g).

Interestingly, not solely chaos was found in the SFG simulation: second-order phase transitions were also found in the SFG. A close-up of the evolution of the density in an SFG is presented in Figure 4, where a fragment of Figure 6g of size $T = 1000$ is visualised. It can be seen that the density, on a few occasions in between stages of chaos, slowly increases or decreases; which is a second-order phase transition (Jaeger, 1998). Thus, the MFA is not only able to predict first-order phase transitions, it can also predict second-order phase transitions, which is unexpected.

Histograms in Figures 5 and 6 (middle column) show the distribution of the density estimates ($T = 1.000.000$) of each network structure and for selected configurations. These histograms illustrate that, when we are dealing with a stable system with no visible first-order phase transitions, the distribution of densities has one mode, as can be seen in Figures 5e and 6k. In the case of a stable system with visible first-order phase transitions, the distribution tends to have two (unequal sized) modes, as can be seen in Figure 6b and 6e. There is also the possibility that there is chaos in a network: Figure 5g and 6g are examples of this situation. In those cases, we see that the histogram has two equal sized modes, which indicates that the system constantly switches between a low and a high density, which is a characteristic of a chaotic network.

Means and standard errors (SEs) calculated for the last hundred density estimates individually in each network structure for selected configurations are shown in Figure 5 and Figure 6 (right column), where the grey lines represent the mean density +/- one SE and where $SE = \sqrt{\frac{\text{density} \cdot (1 - \text{density})}{n}}$. These figures show that, in the case of first-order phase transitions (Figure 6d and 6f), second-order phase transitions (Figure 6i), and no visible phase transitions (Figure 5c and 5f), the spread of the densities is relatively narrow. The same thing cannot be said about the spread of the densities in networks that show chaos (Figure 5i and Figure 6i): as the density switches between extremely high and low values, the upper and lower bound of the density estimates behaves in the same manner, resulting in uninterpretable figures.

Figure 7 shows bifurcation diagrams for each network structure and corresponding 3-D bifurcation diagrams. Bifurcations are drawn based on analytical results (Waldorp, 2015a), and a random sample of 1000 density estimates in each simulation is shown for selected configurations. Stable phases, if present, are shown in the left part of each bifurcation diagram: the two prongs of the fork indicate the location of these phases for values $p_{\text{active}|MB}$. Unstable parts or chaos in the network are shown in the right part of bifurcation diagram. A single line indicates a critical point at which the network is unstable, and a rapid division into multiple prongs indicate chaos. 3-D bifurcation diagrams could not be drawn for the SFG, as the density can theoretically exceed the bounds of $\{0, 1\}$, which is in practice impossible by definition.

Confidence intervals were constructed based on the last hundred density estimates of each configuration, by calculating the mean density of these hundred estimates and adding or subtracting the SE. We inspected the histograms of the density estimates (see Figure 5 and 6) to justify this method.
for constructing confidence intervals, which is only appropriate for normal distributions. In the case of two modes (e.g. Figure 5h), we fitted two normal distributions with the R-package mixtools version 1.0.3 (Benaglia et al., 2009), which all resulted in a good fit. When two modes were fitted, confidence intervals were calculated for each mode, with a split at density 0.50.

It can be seen that, for low values of $p_{\text{active|MB}}$, the density estimates align with the bifurcation diagrams. This indicates that the MFA can accurately predict the density of a network structure at time point $t+1$. Moreover, confidence intervals show that the spread in the density estimates is small, which is also an indication of an accurate performance of the MFA. These results show that the MFA is not only appropriate for predicting first-order phase transitions in a torus, but is also appropriate for predicting first-order phase transitions in an RG and SWG.

In the right part of Figure 7g, we see that the SWG becomes chaotic. The current study confirmed this: when we exploratory simulated with $p_{\text{active|MB}} = \{0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.97\}$, we found a chaotic SWG from $p_{\text{active|MB}} = 0.80$ and higher, as is shown in Figure 5g. Figure 7e shows a critical point at higher values of $p_{\text{active|MB}}$. Results show that the network behaves in an unstable manner, which is expected at a critical point. It shows in Figure 7e that the density estimates cover a wider range of values and even seems to behave as if it is chaotic; the sample of density estimates create the same shape as in Figure 7g.

In Figure 7i, it can be seen that the bifurcation for the SFG quickly converges to two phases, after which the bifurcation stops. This means that, in the parameter space between $p_{\text{active|MB}} = 0$ to $p_{\text{active|MB}} \approx 0.38$, the system does not show stable phases. The current study confirmed this, as is shown in Figure 6. The SFG became less chaotic when the $\gamma$ parameter was decreased, but we could not prove this analytically, as analytical results could not draw a bifurcation diagram.

The 3-D bifurcation diagrams in Figure 7 show that the predicted densities generally align with the analytically derived surface. These results indicate that the MFA behaves as would be expected based on analytical results, not only in a 2-D parameter space, but also in a 3-D parameter space.
In Figure 7d, it can be seen that the density estimates form a cloud at high density values at \( t = 1 \) and \( t \), and that a cloud is starting to emerge at low density values \( t = 1 \) and \( t \). Since we observed a first-order phase transition at \( p_{\text{active}}|\text{MB} = 0.2 \), this is to be expected. Figure 7f shows only one cloud of density estimates, but since this cloud is fairly small in diameter with respect to the density at \( t \) axis (\( \approx 0.30 \)), it suggests that the density estimates are accurately predicted. Lastly, Figure 7h again shows the chaos that is present in the SWG at high values for \( p_{\text{active}}|\text{MB} \). As in Figure 7d, two clouds of density estimates exists, albeit it more strongly than in 7d. However, where the lower cloud aligns with the 3-D surface, the upper is not. As these clouds are parallel to one another on the \( \text{Prem} \) axis, this confirms the chaos and the extreme values between which the density fluctuates.

Figure 8 shows networks at individual time points for each network structure and selected configurations. Although the presented networks denote a tiny portion of all the possible networks that could be selected, this selection lets us take a peak at how the networks are behaving during the MFA. For a torus, we not only show the network, but also the spatial pattern, which is often shown in studies that investigate...
Fig. 7: Bifurcation diagrams in a 2-dimensional space (left column) and in a 3-dimensional space (right column) with a random sample ($T = 1000$) of density estimates plotted against the diagrams. Confidence intervals in the 2-D bifurcation diagrams are constructed using the last hundred estimates.
Fig. 8: Networks of each network structure in selected configurations. BLACK = active node. WHITE = inactive node.
the MFA. Black nodes in the networks denote active nodes, and white nodes denote inactive nodes. These networks not only show the structure of the networks that were simulated, but also its activity pattern at a certain time point. For example, Figure 8k shows an SFG with one inactive node at the center of a lot of active nodes. One would expect that, at the next time point, this node becomes active as a result of its active neighbours, but as can be seen in Figure 8l, a lot of the nodes that are around this centered node change from active to inactive at the next time point. Another interesting sequence is shown in Figure 8o and 8p, where it can be seen that the SWG in Figure 8 has only two active nodes, but at the next time point it has only three inactive nodes. These figures show the chaos that is present in the SWG at high values for $p_{\text{active}}$.

4 Discussion

In this study we investigated the performance of the MFA when network structures other than a torus are under investigation. Even though we only observed first-order phase transitions in the RG, we have shown that the MFA accurately predicts the density at $r+1$ in all network structures, as shown by the confidence intervals in Figure 7. This indicates that, even though we did not observe first-order phase transitions, we are able to detect first-order phase transitions when they do occur.

As mentioned, we did not observe first-order phase transitions in a torus, SWG or SFG. A possible explanation for this is the fact that we only performed one simulation block for each configuration, which consisted of $T = 1,000,000$ time points. In order to properly draw any conclusions with regard to the performance of the MFA in network structures other than a torus, and to investigate whether first-order phase transition occur under the investigated circumstances, we would have to run more simulation blocks. Future research may expand the present study in order to replicate and generalise our results.

In contrast to what was expected, we observed second-order phase transitions in the SFG at certain values of the $\gamma$ parameter. This shows that the MFA is not only able to detect first-order phase transitions, but also second-order phase transitions, which are more subtle in nature. The fact that we found second-order phase transitions in one type of network structure, the SFG, suggests that second-order phase transitions are also possible to detect and predict in other network structures. On the other hand, it is quite possible that the SFG has unique properties which make the detection of second-order phase transition possible. Future research may therefore focus itself on the further investigation of second-order phase transitions, under what circumstances they occur, if they can occur in other types of networks structures, and how well the MFA is able to detect and predict these.

Our simulations did not encompass the entire parameter space of possible values, as is shown in the Figure 7. As analytical results were available before conducting the current study, we deliberately chose to simulate data within the area in which first-order phase transitions were theoretically possible. We cannot draw any conclusions with the regard to the MFA's performance in areas that were not under investigation. That being said, it is not possible to perform the simulation for each parameter value. To illustrate, we can only perform the simulation for the SFG for $1.00 < \gamma < 2.00$, as the mean degree becomes infinite at $\gamma > 2.00$ (Li et al., 2006). In Figure 6, it is shown that the density is already in the range $\{0.00; 1.00\}$ when $\gamma = 2.00$, and that this range narrows to $\{0.22; 0.86\}$ when $\gamma = 1.00$. This suggests that, when we increase the $\gamma$ density values may exceed the range $\{0.00; 1.00\}$, which is impossible by the definition of the density. Future research may therefore focus on expanding the current study by performing the simulation at values with with we can draw conclusions on the MFA's performance in the entire parameter space.

In the current study, it is difficult to discriminate between unstable networks and chaotic networks. As shown in Figures 5, 6 and 7, unstable networks and chaotic networks (Figure 7e and 7g) tend to have the same pattern of density estimates. However, the interpretation of an unstable network and a chaotic network is vastly different: an unstable network is a network that may be on the verge of transitioning between two phases and is thus on a critical point (Scheffer et al., 2009), whereas a chaotic network is a network that is very sensitive to subtle changes and that behaves almost randomly (Tabor and Weiss, 1981). In terms of major depressive disorder, a person whose network is unstable may feel already down, and will only need a small, but significant event to occur in order for that person to transition from healthy to depressed. A person whose network is chaotic is neither healthy nor depressed, nor will they ever reach a stable phase. We may wonder if a person with a chaotic network is experiencing depression at all. It is therefore important for future research to investigate the properties of unstable and chaotic networks and how these differ, in order to improve the discrimination between them.

Lastly, the confidence intervals that were constructed were based on data that were not independent and identically distributed (iid), since the density estimates depend on the estimate at the previous time point. It is quite possible that the intervals shown in Figure 7 are biased, which may result in smaller or bigger intervals than they would have been if there was no bias. We should therefore interpret these intervals with caution, but since we have presented results in various ways and drawn our conclusions on several aspects of the MFA, we believe that there is enough ground to base our conclusions on.
To our knowledge, this is the first study that investigated the performance of the MFA in network structures other than a torus. The present study provides a starting point for a whole range of studies with respect to the MFA and the prediction of first-order phase transitions. One possible next step is the adjustment to the procedure so that we can estimate the parameter from the data, in contrast to fixing it a priori. A proposed method for this is Maximum Likelihood Estimation. By taking the logarithm of the function that defines the conditional probability for the density, given the density at \( t - 1 \), deriving that function and solving it for the parameter one wishes to find the Maximum Likelihood Estimator, we can formulate an equation with which we can find the parameter values for a specific data set. This expansion enables us to parameterise the MFA to fit everyone’s need, which is of great importance once we want to use the MFA to predict first-order phase transitions in major depressive disorder.

Another possible continuation of the current study is improving the MFA so that less time points are needed to make accurate predictions with regard to first-order phase transitions. In our simulation, we simulated a million time points in order to draw conclusions on performance of the MFA. However, it is both impossible and inhumane to expose participants to that amount of questionnaires when we want to apply the MFA to major depressive disorder, for example. Improving the MFA so that less time points are needed for accurate predictions will enable us to apply the method to empirical data and investigate whether the MFA can predict phase transitions in real life participants.

A last possible next step is the usage of a different network parameter in the MFA. In the current study, we used the density to detect first-order phase transitions. As we have shown, the MFA performs quite well when we use the density, but it is possible that other network parameters increase the quality of the MFA even more. Options are the closeness, a measure of a node’s predictive quality, the degree, a measure of a node’s connectivity, or Shannon’s entropy, a measure of how much information cannot be explained by a node (Quax et al., 2013). Together with the mutual information, a measure of variables’ mutual dependence, Shannon’s entropy can determine the reachability of information in the network. It is important here that the parameters that is used in the MFA is stable: a small change in the network must also result in a small change in the parameter value (Segarra and Ribeiro, 2014). Studying the MFA’s performance under different network parameters, and comparing these performances will give use more insight into the quality of the MFA, which network parameters are stable and which network parameters can be best used in the prediction of first-order phase transitions.

In conclusion, this study has shown that the MFA is able to accurately detect and predict first-order phase transition in not only tori, but other network structures as well. The MFA can be used in several applications, like detecting whether a child will transition from failing the conservation task to passing the conservation task (Elkind, 1967), or whether a person transitions from healthy to depressed after experiencing a divorce. The MFA enables researchers, therapists, educators or other clinicians to draw a conclusion as to whether a person will experience a first-order phase transition of some sort, and use this information to adapt their strategy or approach in order to establish the phase transition or prevent the phase transition from occurring.

References


5 Appendix I

sim <- sim # amount of simulations
Nvar <- Nvar # dimensions grid
n <- Nvar^2 # total amount of nodes
t <- t # amount of time steps
# probability for a node to become active
prob <- probs

network <- grid(Nvar, FALSE) # generate lattice
adj <- getWmat(network) # adjacency matrix
# array that hold p for each node
probabilities <- array(0,
         dim = c(t, n, length(prob), sim))
# array that holds the activity pattern
activity <- array(0,
         dim = c(n, t, length(prob), sim))
# array that hold number of active neighbours
r <- array(0, dim = c(n, t, length(prob), sim))
# sum activated neighbours per timestep
sum_activation <- array(0,
         dim = c(t, length(prob), sim))
density <- array(0,
         dim = c(t, length(prob), sim))

# paste all edges
edgelist <- as.matrix(data.frame(
    "Node" = c(network$Edgelist$from,
      network$Edgelist$to),
    "Neighbour" = c(network$Edgelist$to,
      network$Edgelist$from)))

# order edgelist
edgelist <- edgelist[order(edgelist[,1]),]

# save neighbours in data frame
neighbours <- as.data.frame(table(edgelist[,1]))

# maximum no. active neighbours
sum <- sum(neighbours[,2])

for(s in 1:sim)
{
  for(p in 1:length(prob))
  {
    print(c(s, p))
    
    # save diagonal
    activity[, 1, p, s] <- random_diag_active(Nvar^2)
    sum_activation[1, p, s] <- sum(activity[, 1, p, s])
    density[1, p, s] <- sum_activation[1, p, s] / n
    for(n in 1:(Nvar^2))
    {
      if(r[n, 1, p, s] < (neighbours[n,2] / 2))
      {
        probabilities[1, n, p, s] = prob[p]
      } else
      {
        probabilities[1, n, p, s] = 1 - prob[p]
      }
      
      # Majority Rule
      for(time in 2:t)
      {
        r[, time, p, s] = adj \%*\% activity[, time - 1, p, s]
        for(n in 1:(Nvar^2))
        {
          if(r[n, time, p, s] < (neighbours[n,2] / 2))
          {
            probabilities[time, n, p, s] = prob[p]
          } else
          {
            probabilities[time, n, p, s] = 1 - prob[p]
          }
          
          activity[, time, p, s] <-
          as.numeric(probabilities[, p, s] >
          runif(n, 0, 1))
      }
        sum_activation[time, p, s] <-
        sum(activity[, time, p, s])
        density[time, p, s] <-
        sum_activation[time, p, s] / n
      }
  }
}

# # # Calculations timestep t = 0 # # #

# save diagonal