8 Agent-Based Simulations

"There's no love in a carbon atom, no hurricane in a water molecule, and no financial collapse in a dollar bill." P. Dodds[1]

The word *complex* comes from the Latin *com*, which means "together" and *plectere*, which means "to weave". The real meaning of the word *complex*, though, is more *complex* than that. Complex systems can partially be understood as collections of agents that interact non-trivially among themselves and their environments producing novel and rich phenomena that, typically, cannot be anticipated from the study of their individual units [223, 224]. Some topics related to complexity were already studied in the previous chapters without an explicit reference to it. For instance, cellular automata was already studied in Sec. 6.3.1 and game theory was seen in Chapter 7.

As stated in the first chapter, economics is a discipline that deals with interacting people that are subject to emotions, conformity, collective motion, and many other complex phenomena. As Wolfram[2] once put it, in order to develop models that capture these complex behaviors, one must look for novel tools beyond the standard mathematical descriptions that we are used to [225].

This chapter deals exactly with the interaction of agents and the emergence that appears in these processes. It begins exploring the intricate connections that agents make and then moves towards some socioeconomic models of opinion dynamics and segregation. The book ends with a study of kinetic models, linking trade and wealth to the Boltzmann[3] equation.

8.1 COMPLEX NETWORKS

If you paid attention to the endnotes so far, you probably noticed that many important authors make networks of collaboration. The same is valid for banks interacting through credit or firms interacting through trade. Depending on the topology of the network, some behaviors may emerge and affect properties such as performance and fragility. In this section we will study these networks, some of their metrics and applications in econophysics.

In order to study networks, we must study *graphs*. These are ordered pairs $G = (V, E)$ where V is a set of points called *vertices* (or *nodes*), and $E \subseteq \{\{x, y\} | x, y \in V,\ x \neq y\}$ is a set of lines called *edges* (or *links*) that connect those nodes. The graph may be *weighted* if there is a function $w : E \rightarrow \mathbb{R}$ associated with all edges of the graph. On the other hand, if the graph is unweighted, then $w : E \rightarrow \mathbb{B}$, where $\mathbb{B}$ is the Boolean[4] domain $\mathbb{B} = \{0, 1\}$. We say that the graph is *undirected* if the edges are comprised of unordered vertices, and *directed* otherwise. All graphs presented here will be unweighted and undirected, unless explicitly stated otherwise.

DOI: 10.1201/9781003127956-8

A graph can be *simple* if it allows only one edge between a pair of nodes or *multigraph* otherwise. A multigraph with loops is known as *pseudograph*. Finally, a graph can be *complete* if all pair of nodes are connected by edges.

A graph can be represented by an *adjacency matrix* whose elements are the number of edges that directly connect two nodes. In the case of a simple graph, the adjacency matrix is a Boolean matrix[a]. We can also define the *neighborhood* $N_G(v)$ of a node v as a subgraph of G induced by the neighbors (adjacent nodes, or nodes connected by an edge to v) of v.

A *walk* of length n is defined as an alternating sequence of nodes and edges $v_0, e_1, v_1, e_2 \ldots, e_k, v_k$ such that edge $e_i = \{v_{i-1}, v_i\}$, for $1 \leq i \leq n$. In Fig. 8.1, a walk could pass sequentially through nodes $abacd$. This walk can be *closed* if the first and last nodes are the same. An example would be a walk through the sequence $abca$. It is considered *open* otherwise. An example would be abc. Also, a *trail* is defined as a walk with no repeated edges such as $abcd$. A *path* is defined as an open trail with no repeated nodes such as $abcd$. Furthermore, a *cycle* is defined as a closed trail with no repeated vertices except the first and last nodes such as $acdfa$. Finally, a *circuit* is a closed trail with no repeated edges that may have repeated nodes. An example would be $abcfdca$.

Observe that the number of 1-walks between nodes is given by the adjacency matrix. The number of 2-walks between two nodes is given by $\sum_n a_{in}a_{nj}$ but this leads to the product AA. Therefore, we find by induction that the elements of the n^{th} power of the adjacency matrix gives the number of n-walks between these nodes. For instance, for the graph in Fig. 8.1 the adjacency matrix is given by Eq. 8.1. The 2-walks between a and a are aba, aca, and afa and this is captured by the element $A^2_{11} = 3$. Also note that the number of triangles in a graph can be found from the diagonal of A^3. Since a triangle has three nodes and every node is counted, we must divide the trace of A^3 by 3. Also, both clockwise and counterclockwise walks are computed. Therefore the number of triangles is given by $Tr\left(A^3\right)/6$.

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}, A^2 = \begin{bmatrix} 3 & 1 & 2 & 2 & 2 & 1 \\ 1 & 3 & 1 & 1 & 0 & 3 \\ 2 & 1 & 4 & 1 & 2 & 2 \\ 2 & 1 & 1 & 2 & 1 & 1 \\ 2 & 0 & 2 & 1 & 2 & 0 \\ 1 & 3 & 2 & 1 & 0 & 4 \end{bmatrix} \tag{8.1}$$

The *degree matrix*, a related concept, is a diagonal matrix whose elements are the node degrees. A *Laplacian matrix* can be constructed as $L = D - A$.

A graph can be *regular* if all nodes have the same number of connections (have the same number of neighbors, or the same *degree*, or even the same *coordination number*, depending the audience) as shown in Fig. 8.2. If the regular graph (discounting its external nodes, or *leaves*, that have degree 1) contains no cycles and is connected, then it is a *Cayley*[5] *tree*. If this tree has an infinite number of nodes (no

[a] A matrix whose elements are in the Boolean domain $\mathbb{B}$.

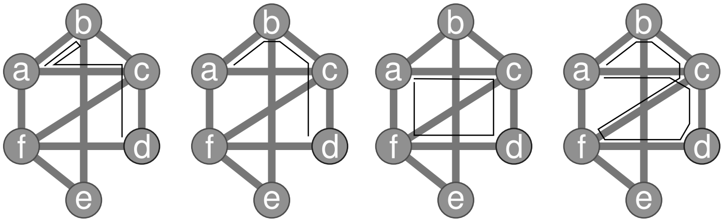

Figure 8.1: A random graph used to illustrate the concepts of (from left to right) walk, trail, cycle, and circuit

leaves), then it is a *Bethe*[6] *lattice*. A *spanning tree* of a graph G is a subgraph of G that is a tree and contains all nodes of G.

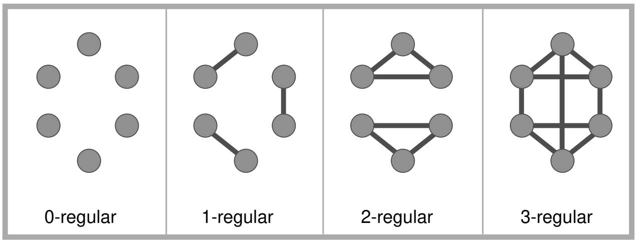

Figure 8.2: Graphs with the same number of nodes but with different regularities

If a graph has non-trivial topological properties, then we call it a *complex network*. A non-trivial topological property could be, for instance, a degree[b] distribution with a long tail. In order to explore these properties we must study some graph metrics.

8.1.1 METRICS

A *metric space* is a tuple (X,d) where X is a set, and $d : X \times X \to \mathbb{R}^+$ is a function (called metric or distance) such that for any $x,y,z \in X$, d satisfies:

1. Leibniz's[7] law of indiscernibility of identicals: $d(x,y) = 0 \iff x = y$,
2. The symmetry axiom: $d(x,y) = d(y,x)$, and
3. The triangle inequality: $d(x,z) \leq d(x,y) + d(x,z)$.

Considering graphs, the *shortest path* (or *geodesic*) between nodes is a metric that can be found in different algorithms such as Dijkstra's[8] [226], Bellman[9]-Ford[10] [227] and Floyd[11]-Warshall[12] [228, 229]. The number of edges in a geodesic is known as *distance* between two nodes. The *eccentricity* of a node is the longest distance between this node and any other in the graph. The *radius* of a graph is the minimum eccentricity of any node. On the other hand, the diameter of a graph is the maximum eccentricity of any node.

[b] Number of connections of a node makes with other nodes.

8.1.1.1 Clustering Coefficient

The node neighborhood of a node v_i is defined as:

$$N_i = \left\{v_j : e_{ij} \in E \vee e_{ji} \in E\right\}. \tag{8.2}$$

The *local clustering coefficient* for a node measures the propensity of the neighborhood of a node to form a clique[c], or cluster together. It is defined as the ratio between the existing number of edges among its neighbors (or similarly, the number of triangles that they form) and the total number of edges that there could be among them:

$$C_i = 2\frac{\left|\left\{e_{jk} : v_j, v_k \in N_i, e_{jk} \in E\right\}\right|}{|N_i|(|N_i|-1)}. \tag{8.3}$$

In terms of the adjacency matrix, it can be computed as:

$$C_i = \frac{\sum_{j,k} A_{ij}A_{jk}A_{ki}}{\sum_j A_{ij}\left(\sum_j A_{ij}-1\right)}. \tag{8.4}$$

The *global clustering coefficient*, on the other hand, is the ratio between the number of closed triplets[d] (or three times the number of triangles) and the number of all triples in the graph. In terms of the adjacency matrix, it is given by:

$$C = \frac{\sum_{i,j,k} A_{ij}A_{jk}A_{ki}}{\sum_i \left[\sum_j A_{ij}\left(\sum_j A_{ij}-1\right)\right]} = \frac{Tr(A^3)}{\sum_i k_i(k_i-1)}, \tag{8.5}$$

where

$$k_i = \sum_j A_{ij}. \tag{8.6}$$

Figure 8.3 shows three graphs with the same number of nodes but with different topologies. The local clustering coefficients for node 1 are $C_1 = 1$, 1/3, and 0 for the graphs a, b, and c, respectively. On the other hand, the global clustering coefficients are $C = 1$, 0.44, and 0, respectively.

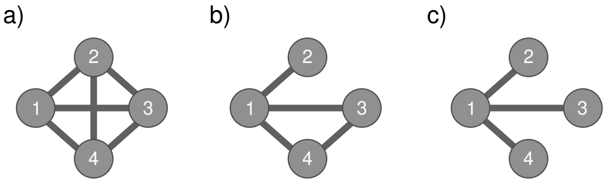

Figure 8.3: Three graphs with the same number of nodes but with distinct topologies

[c] A subgraph where every two distinct nodes are adjacent.

[d] An undirected subgraph consisting of three nodes connected by either two edges (open) or three edges (closed).

A snippet to compute the global clustering coefficient is given bellow.

```
import numpy as np:

def gclust(A):
    N = np.shape(A)[0]

    num = np.trace(np.linalg.matrix_power(A,3))

    k = [0 for j in range(N)]
    for i in range(N):
    for j in range(N):
        k[i] = k[i] + A[i,j]

    den = 0
    for i in range(N):
        den = den + k[i]*(k[i]-1)

    return num/den
```

Some models link the clustering of agents with respect to their demands to many stylized facts such as the fat tails observed in the distribution of returns in the stock market [230].

8.1.1.2 Centrality

How important is a node compared to all others in the network? This is quantified by the *centrality* [231] of a node. The *degree centrality* is simply defined as the ratio between its degree and the total number of nodes subtracted by one. Figure 8.4 shows a *centralized network* where a central node works as a *hub*, a *decentralized network* with multiple hubs, and a *distributed network* with a few or no hubs.

It is also possible to define a *closeness centrality* [232] as the reciprocal of the *farness*, or the sum of the distances d between a specific node and all other nodes of a network:

$$C_c(p) = \frac{|V|-1}{\sum_{(p\neq q)\in V} d(p,q)}. \tag{8.7}$$

There are many other centrality metrics such as *betweenness centrality* [233] and *eigenvector centrality* [234].

The concept of centrality helps us understand some interesting phenomena such as the *friendship paradox*. Consider a network made of symmetrical friendships. The average number of friends a person has is the average degree of the network:

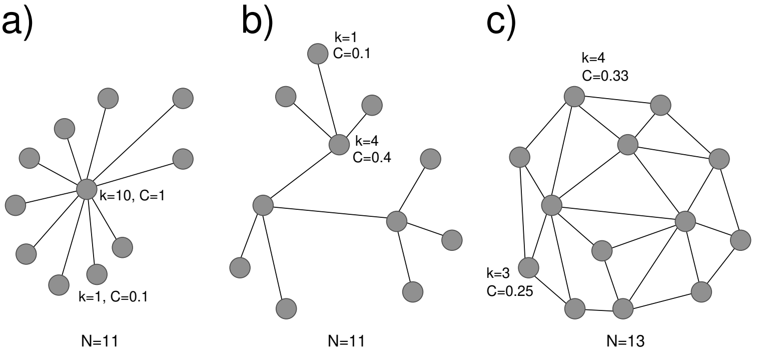

Figure 8.4: a) Centralized network, b) decentralized network, and c) distributed network. *k* indicates the degree of a node, and *C* is its degree centrality

$$\mu = \frac{\sum_{v \in V} d(v)}{|V|}. \tag{8.8}$$

The average number of friends that a friend of a person has, though, can be found by randomly choosing an edge and one of its endpoints. One endpoint is the original person, while the other is a friend. The average degree of the latter node is the value we seek. The probability of selecting a node with a specific degree is:

$$p(v) = \frac{d(v)}{2|E|} = \frac{d(v)}{\sum_{v \in V} d(v)}. \tag{8.9}$$

Hence, the average number of friends of friends is:

$$\begin{aligned} \mu_{ff} &= \sum_{v \in V} p(v) d(v) = \sum_{v \in V} \frac{d^2(v)}{\sum_{v \in V} d(v)} \\ &= \frac{|V|}{|V| \sum_{v \in V} d(v)} \sum_{v \in V} d^2(v). \end{aligned} \tag{8.10}$$

Cauchy-Schwarz Inequality

Consider the polynomial function $f : \mathbb{R} \to \mathbb{R}^+$:

$$\begin{aligned} f(x) &= \sum_i (a_i x - b_i)^2 \\ &= \left(\sum_i a_i^2\right) x^2 - 2\left(\sum_i a_i b_i\right) x + \sum_i b_i^2. \end{aligned} \tag{8.11}$$

Since it is a nonnegative function, its determinant has to be less than or equal to zero:

$$\begin{aligned} \Delta_x = 4\left(\sum_i a_i b_i\right)^2 - 4\left(\sum_i a_i^2\right)\left(\sum_i b_i^2\right) \leq 0 \\ \left(\sum_i a_i^2\right)\left(\sum_i b_i^2\right) \geq \left(\sum_i a_i b_i\right)^2. \end{aligned} \tag{8.12}$$

This is known as the *Cauchy*[a]*-Schwarz*[b]*inequality*.

[a] Augustin-Louis Cauchy (1789–1857) French polymath.

[b] Karl Hermann Amandus Schwarz (1853–1921) Prussian mathematician, advisee of Karl Weierstrass.

According to the Cauchy-Schwarz inequality, we get:

$$\begin{aligned} \mu_{ff} &= \frac{1}{|V|\sum_{v\in V} d(v)}\left(\sum_{v\in V} 1^2\right)\left(\sum_{v\in V} d^2(v)\right) \geq \frac{1}{|V|\left(\sum_{v\in V} d(v)\right)}\left(\sum_{v\in V} d(v)\right)^2 \\ \mu_{ff} &\geq \frac{\sum_{v\in V} d(v)}{|V|} = \mu \\ \mu_{ff} &\geq \mu. \end{aligned} \tag{8.13}$$

This implies that, on average, people tend to make friends with people who already have a number of friends higher than the average centrality. This appears to be a paradox since the original person that we picked could now be a friend's friend. The solution, however, is in the fact that we are talking about averages. This problem, related to structure of the social network, also illustrates the *class size paradox* where a person can experience a much more crowded environment than it really is.

8.1.1.3 Assortativity

In many networks, there is often a tendency of similar nodes to preferentially attach to each other. The *assortativity* (or *homophily*) quantifies this tendency through the correlation of nodes.

Let's define e_{ij} as the probability that an edge connects a node of type i to another node of type j, such that $\sum_{ij} e_{ij} = 1$. Also, let $a_i = \sum_j e_{ij}$ be the probability that an edge comes from a node of type i and $b_j = \sum_i e_{ij}$ be the probability that an edge connects to a node of type j. The assortativity coefficient is then given as:

$$r = \frac{\sum_i e_{ii} - \sum_i a_i b_i}{1 - \sum_i a_i b_i} = \frac{Tr(\mathbf{e}) - \|\mathbf{e}^2\|}{1 - \|\mathbf{e}^2\|}, \tag{8.14}$$

where $\|\mathbf{a}\|$ is the sum of all elements of $\mathbf{a}$. If there is no assortative mixing, then $e_{ij} = a_i b_j$ and, according to Eq. 8.14, $r = 0$. On the other hand, if there is perfect assorativity, then $\sum_i e_{ii} = 1$ and $r = 1$. If the network is perfectly *disassortative* (every node connects to a node of a different type), then $e_{ii} = 0$ and:

$$r = \frac{\sum_i a_i b_i}{\sum_i a_i b_i - 1} = \frac{\|\mathbf{e}^2\|}{\|\mathbf{e}^2\| - 1}. \tag{8.15}$$

The graph in Fig. 8.4a is not assortative, whereas the graphs in Fig. 8.4b and c have assortativity coefficients of -0.6 and -0.2, respectively, implying some level of disassortativeness. A regular graph, on the other hand, is perfectly assortative. The following snippet computes the assortativity.

```
import numpy as np

def assort(A):
    # Degree matrix
    D = np.sum(A,1)

    # Total number of edges
    T = int(np.sum(D)/2)

    # Number of nodes
    N = np.shape(A)[0]

    # Prob. edge from
    e = np.array([[0.0 for i in range(max(D)+1)] for j in range(max(D)+1)])
    for i in range(N-1):
        for j in range(i+1,N):
            e[D[i],D[j]] += A[i,j]/T

    p = np.sum(np.dot(e,e))
    num = np.trace(e)-p
    den = 1-p

    if (np.trace(e) >= 0.9999):
        r = 1
    else:
        r = num/den

return(r)
```

8.1.2 RANDOM NETWORKS

One of the first attempts to understand social networks was given by the random network model proposed by Erdös-Rényi [235][13]. In this model, we start with N unconnected nodes and pick pairs randomly, connecting them by an edge with constant probability p.

The probability of finding a node with degree k is given by the product of three terms: i) the possibility of selecting k links among the total number $N-1$, ii) the probability that k nodes are present, and iii) the probability that the remaining nodes are not chosen. Mathematically, this gives us a binomial distribution:

$$P(k) = \binom{N-1}{k} p^k (1-p)^{N-k}. \tag{8.16}$$

The expected degree of the network can be found expanding the binomial expression:

$$\begin{aligned}
(p+q)^N &= \sum_k \binom{N}{k} p^k q^{N-k-1} \\
\frac{\partial}{\partial p}(p+q)^N &= \sum_k \binom{N}{k} k p^{k-1} q^{N-k-1} \\
N(p+q)^{N-1} &= \frac{1}{p}\sum_k k\binom{N}{k} p^k q^{N-k-1} \\
Np &= \sum_k k\binom{N}{k} p^k (1-p)^{N-k-1}, \text{ where } q = 1-p \\
\langle k \rangle &= Np.
\end{aligned} \tag{8.17}$$

Similarly, we can find the variance of this degree:

$$\begin{aligned}
\frac{\partial^2}{\partial p^2}(p+q)^N &= \sum_k \binom{N}{k} k(k-1) p^{k-2} q^{N-k-1} \\
&= \sum_k \binom{N}{k} k^2 p^{k-2} q^{N-k-1} - \sum_k \binom{N}{k} k p^{k-2} q^{N-k-1} \\
p^2 N(N-1)(p+q)^{N-2} &= \langle k^2 \rangle - \langle k \rangle \\
\langle k^2 \rangle &= p^2 N(N-1) + Np \\
\langle k^2 \rangle - \langle k \rangle^2 &= p^2 N(N-1) + Np - N^2 p^2 \\
\sigma_k^2 &= Np - Np^2 \\
&= Np(1-p).
\end{aligned} \tag{8.18}$$

Therefore, the bigger the network is, the more the distribution shifts and spreads towards larger values. Social networks, though, do not empirically show this behavior [236].

8.1.2.1 Average Path Length

It is fair to say that between two near nodes there can be, on average, $\langle k \rangle$ paths. For second neighbors, there may be $\langle k \rangle^2$ paths, and so on. Therefore, for a distance d, the number of paths is:

$$N(d) = 1 + \langle k \rangle + \langle k^2 \rangle + \ldots + \langle k^d \rangle = \sum_{i=0}^{d} \langle k^i \rangle = \frac{\langle k \rangle^{d+1} - 1}{\langle k \rangle - 1}. \tag{8.19}$$

However, the maximum path length (diameter of the network) cannot be longer than the number of nodes. Therefore, $N(d_{\max}) \approx N$ and we deduce that:

$$\begin{aligned} \langle k \rangle^{d_{\max}} &\approx N \\ d_{\max} \log\left(\langle k \rangle\right) &\approx \log(N) \\ d_{\max} &\approx \frac{\log(N)}{\log\left(\langle N \rangle\right)}. \end{aligned} \tag{8.20}$$

This sublinear relationship between the diameter and the number of nodes is known as *small world phenomenon*. This was initially measured in 1967 by Milgram[14] using letters [237,238]. In short, a recipient would receive a letter addressed to some person and then forward this letter to a friend who was likely to know him or her. After the letter was received by the final contact person, the researcher would count how many steps were necessary for completing this path. The result average path length was about six, which gave rise to the popular expression *six degrees of separation*.

8.1.2.2 Clustering Coefficient

The k_i neighbors of a node i can make a maximum number of connections among themselves:

$$\binom{k_i}{2} = \frac{k_i!}{2(k_i-2)!} = \frac{k_i(k_i-1)}{2}. \tag{8.21}$$

Its neighbors, however, are connected with probability p, creating $p\binom{k_i}{2}$ connections. Hence, the expected number of links among the neighbors of node i is:

$$\langle L_i \rangle = p\frac{k_i(k_i-1)}{2}. \tag{8.22}$$

The clustering coefficient is exactly this probability:

$$C_i = p = \frac{2\langle L_i \rangle}{k_i(k_i-1)} = \frac{\langle k \rangle}{N}, \tag{8.23}$$

as previously derived in Eq. 8.17.

This implies that the clustering coefficient should be inversely proportional to the size of the network. However, this is not observed experimentally either. Rather, social networks show a high clustering coefficient nearly independent on the size of the network.

8.1.3 SCALE-FREE NETWORKS

A network is considered *scale invariant* if its degree distribution follows a power law $P(k) \propto k^{-\gamma}$ with no characteristic scale. Thus, when rescaling its distribution, we get the same distribution (except for a multiplicative factor):

$$P(ak) \propto a^{-\gamma}k^{-\gamma} \propto P(k). \tag{8.24}$$

This distribution can be normalized as:

$$P(k) = Ck^{-\gamma}. \tag{8.25}$$

Since, it is a distribution:

$$\sum_k P(k) = C\sum_k k^{-\gamma} = 1 \rightarrow C = \left(\sum_k k^{-\gamma}\right)^{-1} = \zeta^{-1}(\gamma), \tag{8.26}$$

where $\zeta(\gamma)$ is the Riemann[15] zeta function.

One way of producing scale-free networks is using a *preferential attachment* procedure such as the Yule-Simon's urn process [239, 240] in which the probability of adding a ball to a growing number of urns is linearly proportional to the number of balls already in an urn. In the preferential attachment process in networks proposed by Barabasi[16] and Albert[17] [241, 242] we start with m_0 nodes and a node with $m < m_0$ links is progressively added to the network. Every new node is connected to an existing node with probability proportional to the number of connections it already has:

$$p_i = \frac{k_i}{\sum_n k_n}. \tag{8.27}$$

8.1.3.1 Degree Distribution

The sum in the denominator of the last equation can be computed with the *degree sum formula*[e].

Degree sum formula

Consider a pair (v, e), where v is a node and e is an edge. The number of edges that connect to a node v is simply its degree: $deg(v)$. Therefore, the sum of all degrees is the sum of all incident pairs (v, e). Each edge, though, is connected to two nodes. Therefore, the total number of pairs is twice the number of edges. Since both sums are the same, we conclude that the sum of all degrees of a graph is twice the number of edges:

$$\sum_v deg(v) = 2|E|. \tag{8.28}$$

This implies that the sum of degrees of all nodes (even if they are odd) is always even. Consequently, if we imagine a group of people, the number of those who have shaken hands with people from a subgroup with an odd number of individuals is always even. Hence, this is also known as the *handshaking lemma*.

[e] A nice derivation of the properties of scale-free networks can be found in [243].

If a new node makes m connections at every instant, then the number of edges is mt. If we discount this new node and use the degree sum formula, we find that the denominator gives:

$$\sum_n k_n = 2mt - m. \tag{8.29}$$

The temporal change of the degree of a node has to be propotional to the number of connections that are added and the probability that we find a node with this degree:

$$\begin{aligned} \frac{dk_i}{dt} &= mp_i \approx \frac{mk_i}{2mt} \text{ for large } t \\ \int_{t_i}^{t} \frac{dk_i}{k_i} &= \int_{t_i}^{t} \frac{dt}{2t} \\ \ln\left(\frac{k_i}{m}\right) &= \frac{1}{2}\ln\left(\frac{t}{t_i}\right), \text{ since } k_i(t) = m \\ k_i(t) &= m\left(\frac{t}{t_i}\right)^{1/2}. \end{aligned} \tag{8.30}$$

The probability of finding a node with degree smaller than k is:

$$\begin{aligned} P(k_i(t) < k) &= P\left(m\left(\frac{t}{t_i}\right)^{1/2} < k\right), \text{ from the previous equation.} \\ &= P\left(t_i > \frac{m^2 t}{k^2}\right) \\ &= 1 - P\left(t_i \leq \frac{m^2 t}{k^2}\right). \end{aligned} \tag{8.31}$$

Since we are adding a node at fixed time steps, the number of nodes with degree smaller than k is just $N_< = t\frac{m^2}{k^2}$. On the other hand, the total number of nodes grows linearly as $N_T = m_0 + t \approx t$ for $t \to \infty$. Therefore, the probability of finding a node with a degree smaller than k is:

$$P(k_i(t) < k) = 1 - \frac{m^2}{k^2}. \tag{8.32}$$

Therefore, the probability of finding a node with degree k is:

$$p(k) = \frac{\partial P(k_i(t) < k)}{\partial k} = 2m^2 k^{-3}. \tag{8.33}$$

Most of real social networks show a power-law behavior similar to this.

8.1.3.2 Clustering Coefficient

Let's define the *preferential attachment* of a new node j to an existing node i with degree k_i as:

$$\Pi\left(k_i(j)\right) = \frac{k_i(j)}{\sum_n k_n(j)}. \tag{8.34}$$

If the new node makes m connections, then:

$$p_{ij} = m\Pi\left(k_i(j)\right) = m\frac{k_i(j)}{\sum_l k_l(j)} = \frac{k_i(j)}{2j}. \tag{8.35}$$

Considering that the arrival time of the i^{th} node is i and using the result from Eq. 8.30:

$$p_{ij} = \frac{m\left(\frac{j}{i}\right)^{1/2}}{2j} = \frac{m}{2}(ij)^{-1/2}. \tag{8.36}$$

Assuming now a continuum, the number of connections among neighbors is given by:

$$\begin{aligned} N_\triangle &= \int_{i=1}^{N}\int_{j=1}^{N} P(i,j)P(i,l)P(j,l)\,di\,dj \\ &= \frac{m^3}{8}\int_{i=1}^{N}\int_{j=1}^{N}(ij)^{-1/2}(il)^{-1/2}(jl)^{-1/2}\,di\,dj \\ &= \frac{m^3}{8l}\int_{i=1}^{N}\frac{di}{i}\int_{j=1}^{N}\frac{dj}{j} \\ &= \frac{m^3}{8l}\left(\ln(N)\right)^2. \end{aligned} \tag{8.37}$$

Therefore, the clustering coefficient is given by:

$$\begin{aligned} C_l &= \frac{2N_\triangle}{k_l(k_l-1)} \\ &= \frac{\frac{m^3}{4l}\left(\ln(N)\right)^2}{k_l(k_l-1)}. \end{aligned} \tag{8.38}$$

Using once again the result from Eq. 8.30:

$$\begin{aligned} k_l &= m\left(\frac{N}{l}\right)^{1/2} \\ k_l(k_l-1) &\approx k_l^2 = m^2\frac{N}{l}, \end{aligned} \tag{8.39}$$

we get:

$$C_l \approx \frac{m}{4N}\left(\ln(N)\right)^2. \tag{8.40}$$

Therefore, the Barabasi-Albert network has a higher clustering coefficient when compared to the random network. The clustering behavior found in real social networks, on the other hand, tends to be higher.

8.1.4 SMALL WORLD NETWORKS

The small world property of social networks is well captured by the Watts[18]-Strogatz[19] model [244].

This network can be constructed from a regular ring network composed of N nodes connected to K neighbors symmetrically to each side. A node is picked randomly and $K/2$ neighboring nodes are reconnected with any other node of the ring with probability β[f]. Hence, for $\beta = 0$, we end up with a regular network, whereas for $\beta = 1$ the resulting network is random, as shown in Fig. 8.5. Intermediate values of β generate networks with high clustering coefficients, small diameter, and small world property. The degree distribution for this model, however, does not reflect that of real social networks.

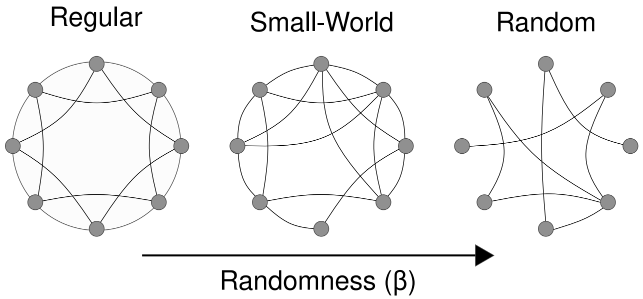

Figure 8.5: Varying parameter β in the Watts-Strogatz model, it is possible to generate regular, small-world, and random networks

8.2 SOCIOECONOMIC MODELS

The Ising[20] model [245, 246] is a popular tool in statistical mechanics used to study ferromagnetic systems. In this model, we have a Hamiltonian[21] given by:

$$\mathscr{H} = -h\sum_i s_i - \sum_{i,j} J_{ij} s_i s_j, \tag{8.41}$$

where h and J are coupling constants, and s is a spin state. Typically J_{ij}, which represents a spin-spin interaction, is such that it is a constant for nearest neighbors and 0 otherwise. The constant h can represents the presence of an external field, and the first sum is related to this field trying to align the spins in a specific direction.

[f] This model is also known as the beta model because of this parameter.

In a typical algorithm to find the equilibrium state of a Ising model, single spin states are randomly created and tested in a Monte Carlo approach (see Appendix B). This is generally a slow process, but alternatives such as the Swendsen[22] [247] and Wolff[23] algorithms [248] are available. In the latter, for example, we create clusters of spins as test states.

In this section we will study how similar ideas using agents instead of spins can be used to model some socioeconomic models. We will start with a model for social segregation and then move to opinion dynamics and will finish the chapter with a simple, yet elegant, model for the formation of prices in a market.

8.2.1 SCHELLING'S MODEL OF SEGREGATION

The Schelling[24] model [249] is an Ising-like agent-based model [250–252] proposed to study social segregation. It is based on an automaton with a $Z \subset \mathbb{Z}^2$ lattice of size N and a neighborhood (originally Moore). The states of the automaton are $S = \{A, B, 0\}$, where A and B are two types of agents that may occupy a grid cell, and 0 indicates an empty one. Only one agent may occupy a grid cell at a time.

The simulation starts with a fraction $\rho = N_0/N^2$ of unoccupied cells. The remaining fraction $1 - \rho$ is randomly occupied by agents of either group with equal probability. This can be created with the following snippet:

```
def createGrid(rho,N):
        return np.array([[np.random.choice([0,-1,1],p=[rho,(1-rho)/2,(1-rho)/2]) \
                for i in range(N)] for j in range(N)])
```

At each round, the agents study their neighborhoods and check the fraction of neighbors that are of the same type:

```
def neighborhood(t):
        p = np.zeros(np.shape(t))

        # Von Neumann neighborhood
        p[:,:-1] = t[:,1:]
        p[:,1:] = p[:,1:] + t[:,:-1]
        p[:-1,:] = p[:-1,:] + t[1:,:]
        p[1:,:] = p[1:,:] + t[:-1,:]

        p = np.where(t != 0, p*t,0)

return p
```

If this fraction is below a certain threshold f, then the agent is unsatisfied and relocates to an empty grid cell[g]. If only unsatisfied agents are allowed to migrate, then we say that is a *constrained* (or *solid*) simulation, whereas if all agents are allowed to migrate (as long as they do not worsen their situations), then we say that it is an *unrestricted* (or *liquid*) simulation. Note that the agents can improve their satisfaction even if their migration may reduce the satisfaction of their neighbors (see Pareto efficiency—Sec. 7.1.4.1). Also, even though the global polarization of the lattice is preserved, the local polarization is not.

We can define two other functions, one for moving an agent to a new destination and another one that finds a new destination:

```
def move(t,frm,to):
    k = t[frm]
    t[frm] = 0
    t[to] = k

def destination(t):
    unoccupied = np.where(t == 0)
    l = len(unoccupied[0])
    p = np.random.randint(l)

    m = unoccupied[0][p]
    n = unoccupied[1][p]

    return(m,n)
```

Given these functions, the simulation itself is performed by:

```
grid = createGrid(0.3,80)

for it in range(50000):
    nbr = neighborhood(grid)
    frm = np.unravel_index(nbr.argmin(),nbr.shape)
    to = destination(grid)
    move(grid,frm,to)
```

[g] See homophily in Sec. 8.1.1.3.

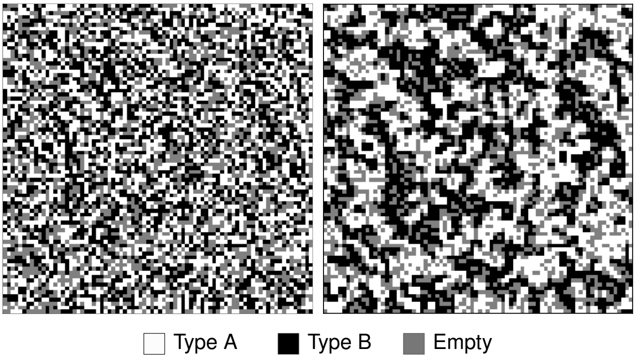

Figure 8.6: Initial grid configuration (left) and steady state grid configuration (right) for the Schelling model with $\rho = 0.3$ on a 80×80 lattice

The result of this simulation is shown in Fig. 8.6. It is clear that the steady state solution exhibits the formation of clusters with agents of different types.

In order to quantify the formation of clusters, we can use the *segregation coefficient* S [253]. This is an order parameter corresponding to the weighted average cluster size:

$$S = \sum_{\{i\}} n_i p_i, \tag{8.42}$$

where $p_i = n_i/M$ is the probability of finding a cluster of mass n_i, and $M = N^2(1-\rho)$ is the total number of agents. A normalized segregation coefficient is often calculated as:

$$s = \frac{S}{M/2} = \frac{2}{M}\sum_{\{i\}} n_i \frac{n_i}{M} = \frac{2}{\left[N^2(1-\rho)\right]^2}\sum_{\{i\}} n_i^2, \tag{8.43}$$

where we have used the fact that the biggest cluster can only be $M/2$. In the extreme situation where there are only two clusters, the normalized segregation coefficient is 1, whereas it is $1/M$ if there is no cluster formation.

Clusters can be identified with the following flood fill algorithm:

```
def cluster(t,y,x,c):
    L = np.shape(t)[0]
    mass = 0

    candidates = [(y,x)]
    while(len(candidates)>0):
        y,x = candidates.pop()
        if (t[y,x] == c):
            if (y > 0):
                candidates.append((y-1,x))
            if (y < L-1):
                candidates.append((y+1,x))
            if (x > 0):
                candidates.append((y,x-1))
            if (x < L-1):
                candidates.append((y,x+1))

            mass = mass + 1
            t[y,x] = 3

    return mass
```

The segregation coefficient can be found with:

```
def segregation(t):
    L = np.shape(t)[0]
    n = []

    for j in range(L):
        for i in range(L):
            mass = cluster(t,j,i,1)
            if (mass > 0):
                n = np.append(n,mass)
            mass = cluster(t,j,i,-1)
            if (mass > 0):
                n = np.append(n,mass)

    return 2*np.sum(n**2)/np.sum(t!=0)**2
```

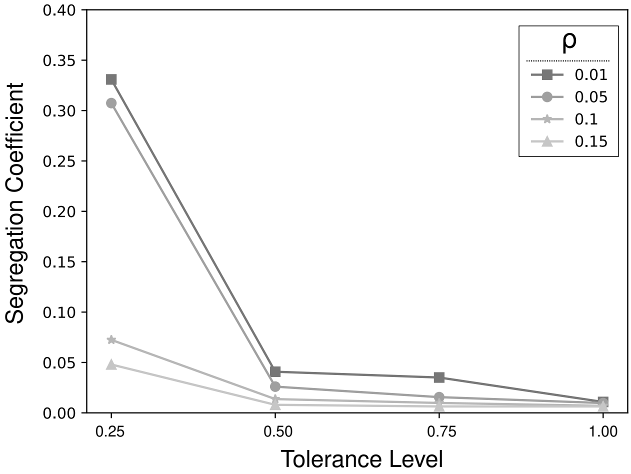

Figure 8.7: Single shot simulation of the segregation coefficient as a function of the tolerance level for the Schelling model on a 80×80 grid and different values of ρ

A single shot simulation of the segregation coefficient as a function of the *tolerance level*[h] is shown in Fig. 8.7. The tolerance level can only assume four values when using von Neumann neighborhood: 1/4, 2/4, 3/4, and 4/4. It is interesting to note that a tolerance level higher than 1/4 is enough to cause a significant lowering of the segregation coefficient.

Although Schelling model does not capture many restrictions such as financial barriers, it is supported by many empirical evidence (see [254, 255], for instance).

8.2.2 OPINION DYNAMICS

In 1951 Asch[25] proposed the following experiment [256]: given a card A with a single line drawn on it and another card B with three lines, a college student would have to answer which of the three lines in card B had the same length as the one in card A. The participant would be in a group with seven confederates that would purposefully choose a wrong answer. Asch also had a control condition where the participant would be tested alone. The experiment showed that approximately 75 % of the participants conformed at least once with the group even if the answer was completely wrong, whereas less than 1 % of the participants gave wrong answers in the control condition. This is a classic experiment that shows the tendency of humans to conform with a group.

When the coordination of behaviors occurs spontaneously without a central authority, we call it *herd behavior*. This is widely seen in nature as a form of *collective behavior*[i]. This happens, for example, during an *information cascade* when investors

[h] The fraction of neighbors of a different type one tolerates before moving.

[i] One of the most popular models for collective motion is the Vicsek[26] model.

tend to follow the investment strategies of other agents rather then their own [257]. This partially explains the dot-com bubble of 2001, for example. After seeing the commercial potential of the internet, investors put aside their personal believes and conformed with a common tendency of investing in e-commerce start-ups. Between 1995 and 2000, the Nasdaq index rose more than 400 %, but after this *irrational exuberance*[27], the index lost all its gains leading to the liquidation of a vast number of companies and a general glut in the job market for programmers.

The Hegselmann[28]-Krause[29] (HK) is an agent-based model [258] that tries to capture this behavior. The simulation starts with an opinion profile $x_i \in [0;1]$, $i = 1,\ldots,N$, where N is the number of agents:

```
import numpy as np
import matplotlib.pyplot as pl

NA = 50
NI = 10

x = np.zeros((NA,NI))
x[:,0] = [np.random.uniform() for i in range(NA)]
```

At each simulation step, a neighborhood for each individual is formed with agents that have similar opinions:

$$\mathcal{N}_t(n) = \{m : |x_t(m) - x_t(n)| \leq \varepsilon\}, \tag{8.44}$$

where ε is a *confidence level*. Therefore, the neighborhood is bounded by this level and the model is often known as a *bounded confidence model*. The following snippet finds the neighborhood and calculate the average opinion:

```
def neighborhood(S,el):
    ac = 0
    z = 0.0
    for y in S:
        if abs(y-el) <= 0.05:
            z = z + y
            ac = ac + 1

    return z/ac
```

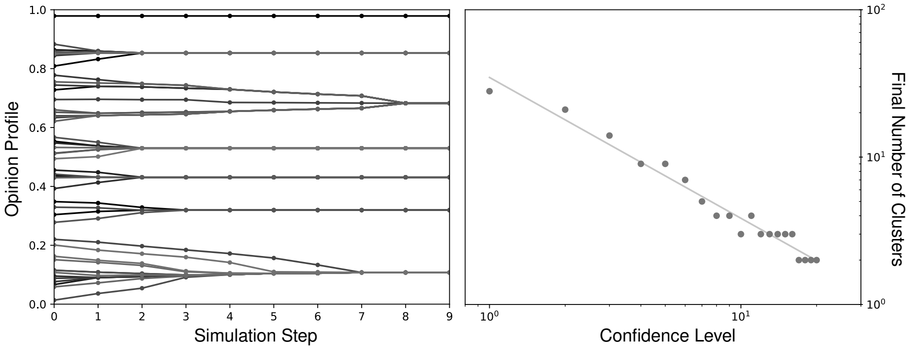

Figure 8.8: The evolution of the opinion profile in the HK model (left) and the final number of clusters as a function of the confidence level

The opinion profile is updated by the average opinion of the neighborhood of each agent:

$$x_n^{t+1} = \frac{1}{|\mathscr{N}_t(n)|} \sum_{m \in \mathscr{N}_t(n)} x_m^t. \tag{8.45}$$

```
for t in range(NI-1):
    for i in range(NA):
        x[i,t+1] = neighborhood(x[:,t],x[i,t])
```

The result for a simulation with 50 agents in shown in Fig. 8.8. Regardless of the initial distribution, clusters of agents tend to be formed and opinions tend to a small set. The final number of clusters approximately depends on the confidence level as $N_{final} \sim \varepsilon^{-1}$. In other words, more groups are formed as the agents restrict their neighborhood of individuals with similar opinions.

A variant of the HK is the Deffuant[30] model [259]. There, pairs of agents are picked randomly and adjust their opinions if they are relatively close. Otherwise, communication is believed not to be possible and they keep their old believes. The update rule is given by:

$$\begin{aligned} x_i(t+\Delta) &= x_i(t) - \mu\left[x_i(t) - x_j(t)\right] \\ x_j(t+\Delta) &= x_j(t) - \mu\left[x_j(t) - x_i(t)\right], \end{aligned} \tag{8.46}$$

where $\mu \in [0, 1/2]$ is a *convergence parameter*.

The initialization for this simulation is identical to that of the HK model, but the dynamics is now given by:

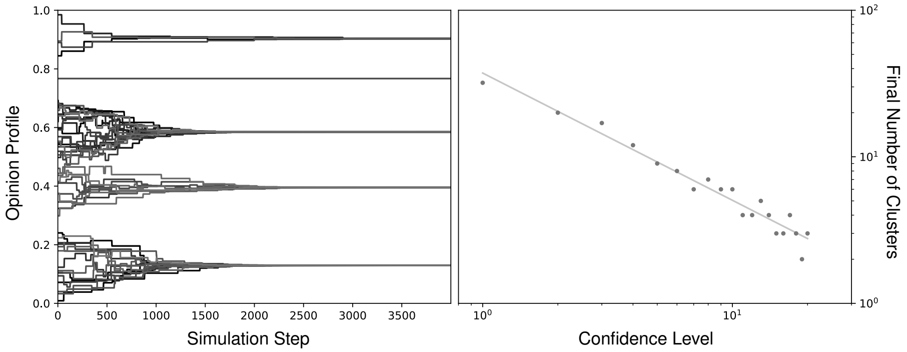

Figure 8.9: The evolution of the opinion profile in the Deffuant model (left) and the final number of clusters as a function of the confidence level

```
for t in range(NI-1):
        i = np.random.randint(NA)
        j = np.random.randint(NA)

        x[:,t+1] = x[:,t]
        if abs(x[i,t]-x[j,t]) < 0.1:
                x[i,t+1] = x[i,t] - mu*(x[i,t]-x[j,t])
                x[j,t+1] = x[j,t] - mu*(x[j,t]-x[i,t])
```

The result of a simulation with 50 agents and $\mu = 0.35$ is shown in Fig. 8.9.

As in the HK model, the final number of clusters approximately depends on the confidence level as $N_{final} \sim \varepsilon^{-0.9}$.

8.2.2.1 Kirman Model

Imagine two nearly identical securities. It is not uncommon that a large majority of investors end up choosing one rather than the other. The same happens, for instance, with ants presented with two sources of similar foods. Instead of the ants exploring both sources equally, one source is consumed first. The Kirman[31] model [260] is a Markov chain agent-based model created to answer this kind of problem.

Let's start with two distinct sources A and B and a total population of N agents. At each simulation step, two random agents meet and one is converted to the other's opinion with a chance $1 - \delta$. There is also a probability ε that an agent changes its opinion independently of meeting another agent. This could happen, for instance, as a reaction to an exogenous information.

The state of the system $k \in (0,1,\ldots,N)$ is defined as the number of agents that prefer source A. Therefore, the probability p_1 that k increases by one agent is given by:

$$p_1 = p(k,k+1) = \left(1 - \frac{k}{N}\right)\left(\varepsilon + (1-\delta)\frac{k}{N-1}\right), \tag{8.47}$$

whereas the probability that k is decreased by one agent is given by:

$$p_2 = p(k,k-1) = \frac{k}{N}\left(\varepsilon + (1-\delta)\frac{N-k}{N-1}\right). \tag{8.48}$$

There is also a probability $p_3 = 1 - p_1 - p_2$ that k remains unchanged. Note that this model resembles a Polya urn process (see Sec. 3.1.2). If $\varepsilon = 1/2$ and $\delta = 1$ then it is just an Ehrenfest[32] urn process where the agents change opinions without any interaction. Also, when $\varepsilon = \delta = 0$, the expected value of k is:

$$\langle k_{n+1} | \mathscr{F}_n \rangle = (k_n + 1)p_0 + (k_n - 1)p_0 + k_n(1 - 2p_0) = k_n, \tag{8.49}$$

where

$$p_0 = \left(1 - \frac{k}{N}\right)\frac{k}{N-1} = \frac{k}{N}\frac{N-k}{N-1}. \tag{8.50}$$

Therefore, under these parameters, the process becomes a martingale (see Sec. 3.1).

The Kirman model can easily be simulated with the following snippet:

```
N = 100
eps = 0.15
delta = 0.3
k = N/2
x = []

for it in range(10000):
        p1 = (1-float(k)/N)*(eps+(1-delta)*float(k)/(N-1))
        p2 = (float(k)/N)*(eps+(1-delta)*(N-k)/(N-1))

        r = np.random.rand()

        if (r <= p1):
                k = k + 1
        if (r > p1 and r <= p1+p2):
                k = k - 1

x.append(k)
```

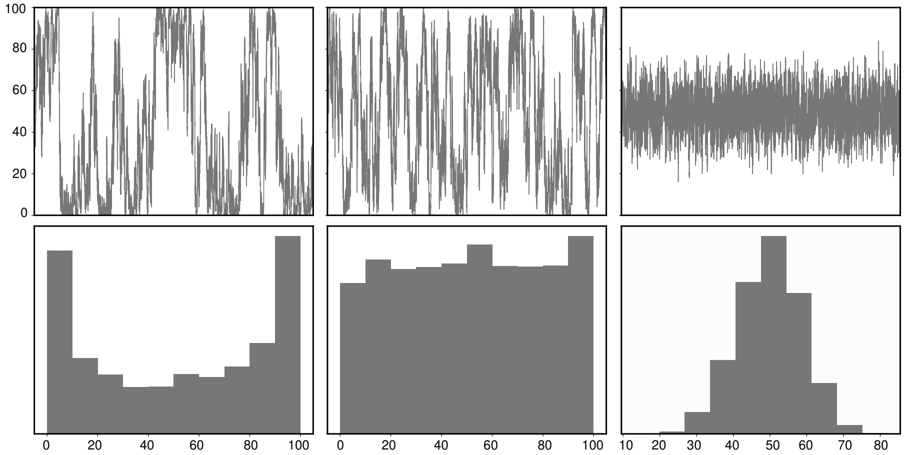

Figure 8.10: The number of agents k that prefer source A as a function of the simulation step (top) and their respective histograms for $\varepsilon = 5 \times 10^{-3}$, $\delta = 10^{-2}$ (left), $\varepsilon = 10^{-2}$, $\delta = 2 \times 10^{-2}$ (center), and $\varepsilon = 0.15$, $\delta = 0.3$ (right)

Some results of this simulation are shown in Fig. 8.10. It is interesting to note that it is possible to adjust the type of distribution of k by changing the values of ε and δ.

8.2.3 MARKET SPIN MODELS

The Bornholdt model[33] [261] is another Ising-like automaton on a $Z \subset \mathbb{Z}^2$ lattice of size N and a Von Neumann neighborhood. The states of this automaton are $S = \{+1, -1\}$, where $+1$ corresponds to a buyer, and -1 corresponds to a seller state.

The simulation starts with a random field and the dynamics is given by a stochastic transition function that assigns the state $+1$ to a cell with probability p_i, and the state -1 with a probability $1 - p_i$. The probability p_i is given by:

$$p_i = \left[1 + \exp\left(-2\beta h_i(t)\right)\right]^{-1}, \tag{8.51}$$

where β is the inverse temperature. $h_i(t)$ is a local field representing the influence to conform with the majority of nearest neighbors in accordance with earlier models [262, 263]:

$$h_i(t) = J \sum_{j \in N^1_{VN}(i)} S_j(t) - \alpha C_i(t) \langle S \rangle, \tag{8.52}$$

where J is a *disagreement* constant that indicates the tendency for the agent to conform, and $\alpha > 0$ is a *demagnetizing* constant that indicates the tendency for the agent to seek an anti-ferromagnetic order. This latter term represents the preference towards the minority group (see Sec. 7.1.1.1). C_i is a second spin available to each cell that relates to the strategy of the agent with respect to the global magnetization.

$C_i = 1$, for instance, designates a desire for the agent to join the global minority group that is interested in future returns, a *fundamentalist* behavior. The other situation $C_i = -1$ points to the desire to follow the majority group, a *chartist* behavior. Thus, the dynamics of the *strategy spin* is given by:

$$C_i(t+1) = \begin{cases} -C_i(t) & \text{if } \alpha S_i(t) C_i(t) \sum_j S_j(t) < 0, \\ C_i(t) & \text{otherwise.} \end{cases} \tag{8.53}$$

If, however, the strategy spin is allowed to change instantaneously, then Eq. 8.52 becomes:

$$h_i(t) = J \sum_{j \in N^1_{VN}(i)} S_j(t) - \alpha S_i(t) \left| \langle S(t) \rangle \right|. \tag{8.54}$$

The magnetization of the system $M(t) = \langle S(t) \rangle$ is identified as the price, from which it is possible to obtain the logarithmic returns.

To simulate the Bornholdt model we start with the following snippet:

```
import numpy as np
import random as rd
import matplotlib.pyplot as pl

N = 32
J = 1
beta = 1.0/1.5
alpha = 4

S = np.array([rd.choices([1,-1],k=N) for i in range(N)])
Sn = np.zeros((N,N))

r = []
M = 1
```

The simulation itself is an update loop:

```
for it in range(2000):
    Ml = M
    M = np.average(S)
    for i in range(N):
        for j in range(N):
            sm = 0
            for x in neig(i,j,N):
                sm = sm + S[x[0],x[1]]
            h = J*sm - alpha*S[i,j]*abs(M)
            p = 1.0/(1+np.exp(-2*beta*h))

            if (rd.random() < p):
                Sn[i,j] = 1
            else:
                Sn[i,j] = -1

        S = Sn
        r = np.append(r,np.log(abs(M))-np.log(abs(Ml)))
```

In the snippet, *neig* is a function that returns the Von Neumann neighborhood:

```
def neig(i,j,N):
    z = []
    if (i > 0):
        z.append([i-1,j])
    if(i < N-1):
        z.append([i+1,j])
    if (j > 0):
        z.append([i,j-1])
    if (j < N-1):
        z.append([i,j+1])

    return np.array(z)
```

The result of the simulation shows metastable phases as shown in Fig. 8.11. Moreover, the log-returns in Fig. 8.12 show fat tails as indicated by the CCDF. Bornholdt also showed that his model also shows some stylized facts such as volatility clustering.

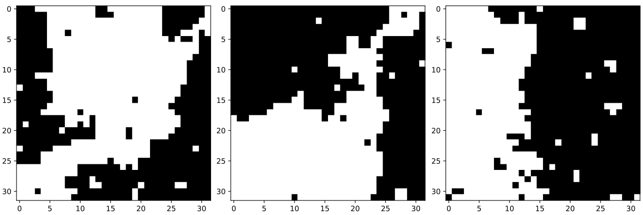

Figure 8.11: Grid configuration on a 32 × 32 lattice at undercritical temperature after t=100, 200, and 300 simulation steps (from left to right)

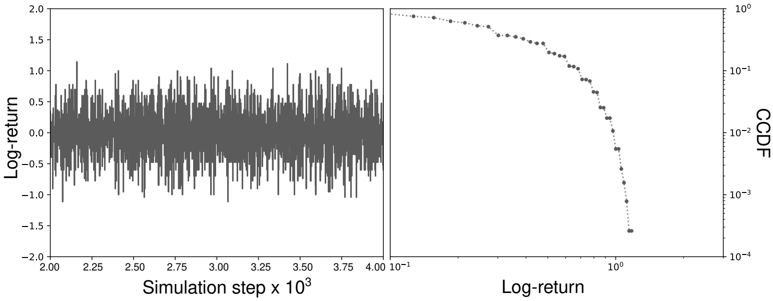

Figure 8.12: Left: Log-returns for the Bornholdt model with $\alpha = 4.0$, $J = 1.0$, and $\beta = 2/3$. Right: The corresponding complementary cumulative distribution function (CCDF)

8.3 KINETIC MODELS FOR WEALTH DISTRIBUTION

In the case of socioeconomic problems, we can use *ad hoc* models [264] where particles maps to agents, energy maps to wealth[j], and the binary collisions map to trade interactions. Under this scheme, we can invoke the famous Boltzmann equation to model the interaction among agents (see Appendix F).

In simulations of wealth distribution, for instance, two traders with initial possessions $v_1, v_2 \in \Omega \subseteq \mathbb{R}$ meet randomly and are distributed as:

$$\begin{bmatrix} v_1'(t) \\ v_2'(t) \end{bmatrix} = \begin{bmatrix} p_1 & q_1 \\ p_2 & q_2 \end{bmatrix} \begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix}, \tag{8.55}$$

where $v_1', v_2' \in \Omega$ as well.

In order to work with the framework of random processes, we can adopt associated independent random variables X and Y that are distributed in accordance with:

$$P(X \in S) = P(Y \in S) = \int_S f(v,t)dv, \ \forall S \subseteq \Omega. \tag{8.56}$$

Their interaction rules are:

$$\begin{bmatrix} X'(t) \\ Y'(t) \end{bmatrix} = \begin{bmatrix} p_1 & q_1 \\ p_2 & q_2 \end{bmatrix} \begin{bmatrix} X(t) \\ Y(t) \end{bmatrix}, \tag{8.57}$$

After a small period Δ, the random variables are updated to $X(t+\Delta) = X'(t)$ if there was a binary interaction and $X(t+\Delta) = X(t)$ otherwise. The probability of interaction can be assigned to a Bernoulli distributed random variable T such that $P(T=1) = 1 - P(T=0) = \mu\Delta$, where μ is an *interaction kernel*.

Therefore, we can write:

$$\begin{aligned} X(t+\Delta) &= TX'(t) + (1-T)X(t) \\ Y(t+\Delta) &= TY'(t) + (1-T)Y(t). \end{aligned} \tag{8.58}$$

For any linear observable φ we can calculate expected values:

$$\begin{aligned} \langle \varphi(X(t+\Delta)) \rangle &= \langle T\varphi(X'(t)) \rangle + \langle (1-T)\varphi(X(t)) \rangle \\ \langle \varphi(Y(t+\Delta)) \rangle &= \langle T\varphi(Y'(t)) \rangle + \langle (1-T)\varphi(Y(t)) \rangle \end{aligned}. \tag{8.59}$$

This can be rewritten as:

$$\begin{aligned} \langle \varphi(X(t+\Delta)) - \varphi(X(t)) \rangle &= \mu\Delta \left[\langle \varphi(X'(t)) \rangle - \langle \varphi(X(t)) \rangle \right] \\ \langle \varphi(Y(t+\Delta)) - \varphi(Y(t)) \rangle &= \mu\Delta \left[\langle \varphi(Y'(t)) \rangle - \langle \varphi(Y(t)) \rangle \right] \end{aligned}. \tag{8.60}$$

Taking the limit when $\Delta \to 0$:

$$\begin{aligned} \frac{\partial}{\partial t} \langle \varphi(X(t)) + \varphi(Y(t)) \rangle = \mu \big[&\langle \varphi(X'(t)) \rangle + \langle \varphi(Y'(t)) \rangle \\ &- \langle \varphi(X(t)) \rangle - \langle \varphi(Y(t)) \rangle \big]. \end{aligned} \tag{8.61}$$

[j] Not to be confused with income, which is an inflow of resources.

We are assuming that the possessions are uncorrelated (*Stosszahlansatz*). Therefore, we take the joint probability distribution as the product of the individual ones (see Appendix G). Thus,

$$
\begin{aligned}
&2\partial_t \left\langle \int_\Omega \varphi(v) f(v,t) dv \right\rangle = \\
&\quad = \mu \left\langle \iint_{\Omega\times\Omega} \left[\varphi(v_1') + \varphi(v_2') - \varphi(v_1) - \varphi(v_2)\right] f(v_1,t) f(v_2,t) dv_1 dv_2 \right\rangle_{\substack{p_1,q_1\\ p_2,q_2}}.
\end{aligned}
\tag{8.62}
$$

It is possible to obtain a more physics-friendly equation considering a constant interaction kernel and a Dirac[34] delta observable $\varphi(\star) = \delta(v-\star)$ in this weak form of the Boltzmann equation:

$$
\partial_t f(v,t) = \frac{1}{2} \left\langle \iint_{\Omega\times\Omega} \left[\delta(v-v_1) + \delta(v-v_2)\right] f(v_1,t) f(v_2,t) dv_1 dv_2 \right\rangle - f(v,t). \tag{8.63}
$$

This equation can be written as:

$$
\partial_t f + f = Q_+(f,f), \tag{8.64}
$$

where $Q_+(f,f)$ is the *collision operator* given by:

$$
Q_+(f,f) = \frac{1}{2} \left\langle \iint_{\Omega\times\Omega} \left[\delta(v-v_1) + \delta(v-v_2)\right] f(v_1,t) f(v_2,t) dv_1 dv_2 \right\rangle. \tag{8.65}
$$

8.3.1 CONSERVATIVE MARKET MODEL

Some models assume that trade is conservative [265, 266], implying that the wealth is preserved in a transaction. This considers wealth as having an objective nature quantifiable by the stock of some scarce resource such as gold. In this case we have:

$$
\int_\Omega f(v,t) dv = m, \tag{8.66}
$$

where m is a finite and constant wealth. Also, we have that the collisions are perfectly elastic: $\langle p_1 + p_2 \rangle = \langle q_1 + q_2 \rangle = 1$. In the Chakraborti-Chakrabarti[35] model [265], for example, the distribution of wealth is given by:

$$
\begin{bmatrix} v_1' \\ v_2' \end{bmatrix} = {}^1\!/_2 \begin{bmatrix} 1+\lambda & 1-\lambda \\ 1-\lambda & 1+\lambda \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}, \tag{8.67}
$$

where $\lambda \in [0,1]$ is a parameter that relates to the saving tendency of the agents. According to this rule, we have conservation since $v_1' + v_2' = v_1 + v_2$.

The variance of the wealth in this model can be found using the observable $\varphi(v) = (v-m)^2$ in the weak form (Eq. 8.62):

$$\begin{aligned}
&\partial_t \left\langle \int_\Omega (v-m)^2 f(v,t)dv \right\rangle = \\
&= \frac{1}{2} \left\langle \iint_{\Omega\times\Omega} \left[(v_1'-m)^2 + (v_2'-m)^2 - (v_1-m)^2 - (v_2-m)^2\right] \right. \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \left. f(v_1,t)f(v_2,t)dv_1dv_2 \right\rangle \\
&= -\frac{(1-\lambda^2)}{4} \left\langle \iint_{\Omega\times\Omega} (v_1-v_2)^2 f(v_1,t)f(v_2,t)dv_1dv_2 \right\rangle \\
&= -\frac{(1-\lambda^2)}{4} \left\langle \iint_{\Omega\times\Omega} \left[(v_1-m)-(v_2-m)\right]^2 f(v_1,t)f(v_2,t)dv_1dv_2 \right\rangle \\
&= -\frac{(1-\lambda^2)}{4} \left\langle \iint_{\Omega\times\Omega} \left[(v_1-m)^2 + (v_2-m)^2 - 2(v_1-m)(v_2-m)\right] \right. \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \left. f(v_1,t)f(v_2,t)dv_1dv_2 \right\rangle .
\end{aligned}$$

Since the two processes are uncorrelated, we end up with:

$$\partial_t \int_\Omega (v-m)^2 f(v,t)dv = -\frac{(1-\lambda^2)}{2} \int_\Omega (v-m)^2 f(v,t)dv. \tag{8.68}$$

Hence, the variance of the distribution approaches zero at an exponential rate $-(1-\lambda^2)/2$. This implies that every agent ends up with the same wealth, which is not what is observed in real scenarios.

8.3.2 NON-CONSERVATIVE MARKET MODEL

If wealth is preserved in a transaction, why would anyone engage in trade? People trade because they give more value to the good they are receiving rather than the one that is being given. This imposes big limitations to the use of conservative models (see, for instance: [267]).

Value is subjective[k] [268]. Thus, making a cardinal measurement of wealth troublesome. Rather, one usually resorts to ordinal descriptions, or in a wider context, wealth could be defined as the ability to have one's desires fulfilled. Therefore, trade can only occur if there is an increase in wealth for both players, producing a positive sum game. The quantification of marginal utility and possession, nonetheless, has to be accepted and some non-conservative models that try to incorporate an increase of wealth in economic transactions have been devised [269, 270].

Slalina's[36] model, for instance, is a model inspired by dissipative gases where the exchange rule is given by:

[k] Consider this famous paradox attributed to Adam Smith: What would the values of water and diamond be for a person who is dying of dehydration in a desert?

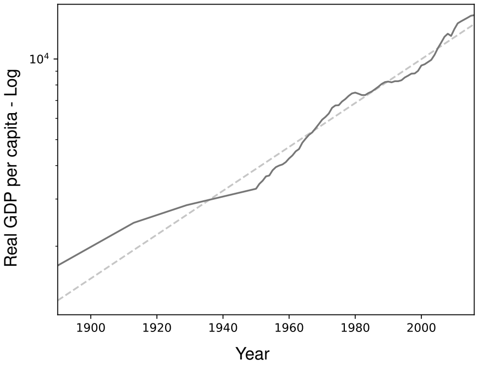

Figure 8.13: Real global GDP *per capita* adjusted for the value of money in 2011-US$ (created with data from [1] through ourworldindata.org; the dashed line is an exponential fit)

$$\begin{bmatrix} v_1'(t) \\ v_2'(t) \end{bmatrix} = \begin{bmatrix} 1-\beta+\varepsilon & \beta \\ \beta & 1-\beta+\varepsilon \end{bmatrix} \begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix}, \tag{8.69}$$

where ε is a positive growth rate parameter, and $\beta \in [0,1]$ plays the role of the saving propensity.

The evolution of the average wealth can be found using $\varphi(v) = v$ in Eq. 8.62:

$$\begin{aligned} 2\partial_t \left\langle \int_\Omega v f(v,t) dv \right\rangle &= \\ & \left\langle \iint_{\Omega\times\Omega} \left[(1-\beta+\varepsilon)v_1 + \beta v_2 + \beta v_1 + (1-\beta+\varepsilon)v_2 - v_1 - v_2\right] \right. \\ & \qquad \left. f(v_1,t) f(v_2,t) dv_1 dv_2 \right\rangle \\ \frac{\partial \bar{v}}{\partial t} &= \frac{1}{2} \left\langle \iint_{\Omega\times\Omega} \varepsilon (v_1 + v_2) f(v_1,t) f(v_2,t) dv_1 dv_2 \right\rangle \\ \frac{\partial \bar{v}}{\partial t} &= \varepsilon \left\langle \int_\Omega v f(v,t) dv \right\rangle = \varepsilon \bar{v}. \end{aligned} \tag{8.70}$$

Consequently, the average wealth grows as $\bar{v} = \bar{v}_0 e^{\varepsilon t}$. The GDP[1] *per capita* indeed shows an exponential growth as illustrated in Fig. 8.13.

Since the wealth grows exponentially, there is no steady state solution. Nonetheless, it is possible to seek self-similar solutions rescaling the wealth distribution as:

$$f(v,t) = \frac{1}{\bar{v}(t)} g\left(\frac{v}{\bar{v}(t)}, t\right). \tag{8.71}$$

[1] Gross domestic product, a measure of all goods and services produced during a specific period in an economy.

The temporal derivative of this equation is given by:

$$\begin{aligned}\frac{df}{dt} &= -\frac{1}{\bar{v}^2}\frac{d\bar{v}}{dt}g + \frac{1}{\bar{v}}\left(\frac{\partial g}{\partial t} + \frac{\partial g}{\partial x}\frac{\partial x}{\partial \bar{v}}\frac{\partial \bar{v}}{\partial t}\right),\ x = \frac{v}{\bar{v}} \\ &= -\frac{\varepsilon}{\bar{v}}g + \frac{1}{\bar{v}}\left[\frac{\partial g}{\partial t} + \bar{v}\frac{\partial g}{\partial v}\left(-\frac{v}{\bar{v}^2}\right)\varepsilon\bar{v}\right] \\ &= \frac{\varepsilon}{\bar{v}}g + \frac{1}{\bar{v}}\left(\frac{\partial g}{\partial t} - \varepsilon v\frac{\partial g}{\partial v}\right) \\ &= \frac{1}{\bar{v}}\left[\frac{\partial g}{\partial t} - \varepsilon\left(g + v\frac{\partial g}{\partial v}\right)\right] \\ &= \frac{1}{\bar{v}}\left[\frac{\partial g}{\partial t} - \varepsilon\frac{\partial}{\partial v}(vg)\right].\end{aligned} \tag{8.72}$$

Using this result in Eq. 8.64 we get:

$$\begin{aligned}&\frac{1}{\bar{v}}\left[\frac{\partial g}{t} - \varepsilon\frac{\partial}{\partial v}(vg)\right] + f = Q_+(f,f) \\ &\frac{\partial g}{\partial t} = Q_+(g,g) - g + \varepsilon\frac{\partial}{\partial v}(vg),\end{aligned} \tag{8.73}$$

which now shows a drift term related to the growth of wealth.

Following the steps showed at the beginning of this section backwards, we get the equation:

$$\begin{aligned}&\left\langle\frac{\partial}{\partial t}\int_\Omega \varphi(v)g(v,t)dv\right\rangle - \varepsilon\left\langle\int_\Omega \varphi(v)\frac{\partial}{\partial v}(vg)dv\right\rangle = \\ &\quad = \frac{1}{2}\left\langle\int_{\Omega\times\Omega}\left[\varphi(v_1') + \varphi(v_2') - \varphi(v_1) - \varphi(v_2)\right]g(v_1,t)g(v_2,t)dv_1dv_2\right\rangle.\end{aligned} \tag{8.74}$$

Making $\varphi(v) = e^{-sv/\bar{v}}\Theta(v)$[m], the integrals become Laplace transforms and we get, according to the transition rule (Eq. 8.69):

$$\frac{\partial}{\partial t}G(s) + \varepsilon s\frac{\partial}{\partial s}G(s) = G\left([1-\beta+\varepsilon]s\right)G(\beta s) - G(s). \tag{8.75}$$

In steady state:

$$\varepsilon s\frac{\partial}{\partial s}G(s) = G\left([1-\beta+\varepsilon]s\right)G(\beta s) - G(s). \tag{8.76}$$

By expanding the parameters β and ε in Taylor series and taking the inverse Laplace transform, Slalina showed that the asymptotic wealth distribution for the tails displays a Pareto power-law behavior [264, 269]. This is observed in real data (see for instance [271]).

[m] Θ is the Heaviside[37] function defined as $\Theta(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{otherwise.} \end{cases}$

Notes

[1] Peter Sheridan Dodds, Australian mathematician.

[2] Stephen Wolfram (1959–) British physicist.

[3] Ludwig Eduard Boltzmann (1844–1906) advisee of Josef Stefan, Gustav Kirchhoff and Herman von Helmholtz among others. Boltzmann advised Paul Ehrenfest among others.

[4] George Boole (1815–1864) British philosopher.

[5] Arthur Cayley (1821–1895) British mathematician winner of many awards, including the De Morgan Medal in 1884.

[6] Hans Albrecht Bethe (1906–2005) German physicist winner of many awards, including the Nobel Prize in Physics in 1967. Bethe was advised by Arnold Sommerfeld and advised many notable students including Jun John Sakurai, David James Thouless, and Freeman Dyson.

[7] Gottfried Wilhem von Leibniz (1646–1716) German polymath advised by Christian Huygens (among others) and adviser of Jacob Bernoulli.

[8] Edsger Wybe Dijkstra (1930–2002) Dutch computer scientist.

[9] Richard Ernest Bellman (1920–1984) American mathematician winner of many prizes including the John vou Neumann Theory Prize in 1976.

[10] Lester Randolph Ford Jr. (1927–2017) American mathematician.

[11] Robert W. Floyd (1936–2001) American computer scientist, winner of the Turing Award in 1978.

[12] Stephen Warshall (1935–2006) American computer scientist.

[13] Paul Erdös (1913–1996) and Alfréd Rényi (1921–1970) Hungarian mathematicians.

[14] Stanley Milgram (1933–1984) American social psychologist.

[15] Georg Friedrich Bernhard Riemann (1826–1866) German mathematician, advisee of Carl Friedrich Gauss.

[16] Albert-László Barabási (1967–) Romanian physicist advisee of Eugene Stanley and Tamás Vicsek.

[17] Réka Albert (1972–) Romanian physicist.

[18] Duncan James Watts (1971–) Canadian physicist, advisee of Steven Strogatz.

[19] Steven Henry Strogatz (1959–) American mathematician, adviser of Duncan Watts.

[20] Ernst Ising (1900–1998) German physicist.

[21] William Rowan Hamilton (1805–1865) Irish mathematician.

[22] Robert Swendsen, American physicist.

[23] Ulli Wolff, German physicist.

[24] Thomas Crombie Schelling (1921–2016) American economist winner of the Nobel Memorial Prize in Economic Sciences in 2005.

[25] Solomon Eliot Asch (1907–1996) Polish social psychologist, adviser of Stanley Milgram.

[26] Tamás Vicsek (1948–) Hungarian physicist, adviser of Albert-László Barabási.

[27] Phrase firstly used by Alan Greenspan (1926–) American economist, chair of the Federal Reserve of the United States between 1987 and 2006.

[28] Rainer Hegselmann (1950–) German philosopher.

[29] Ulrich Krause, German mathematician and economist.

[30] Guillaume Deffuant, French complexity scientist.

[31] Alan Kirman (1939–) British economist.

[32] Paul Ehrenfest (1880–1933) Austrian physicist, advisee of Ludwig Boltzmann and adviser of George Uhlenbeck among others.

[33] Stefan Bornholdt, German physicist.

[34] Paul Adrien Maurice Dirac (1902–1984) English physicist, adivisee of Ralph Fowler. Dirac was the recipient of the Nobel Prize in Physics in 1933.

[35] Anirban Chakraborti and Bikas Kanta Chakrabarti (1952–), Indian physicists.

[36] František Slanina, Czech physicist.

[37] Oliver Heaviside (1850–1925) English polymath.