Einführung ins wissenschaftliche Arbeiten: Biophysik/Physik komplexer Systeme

Student:	Robin Cedric Danek
E-Mail:	robincerdic.danek@stud.uni-goettingen.de
Betreuer 1:	Dr. Joao Pinheiro Neto
E-Mail:	joaopn@ds.mpg.de
Betreuer 2:	Riccardo Carlucci
E-Mail:	riccardo.carlucci@ds.mpg.de
Gutachter:	Prof. Dr. Stephan Herminghaus
E-Mail:	stephan.herminghaus@ds.mpg.de
Matrikelnummer:	21978121

Contents

1 Introduction

In this report I am going to elaborate the tools I need to work on my bachelor thesis, for which I´m going to look into opinion polarization and possible mechanisms of depolarization. I will start with covering some basic network theory. After that I am going to introduce the model proposed by Fabian Baumann[1] which I´m going to use. At the end, I will elaborate some important code I need and I will cover two methods of detecting whether polarization is present in a multidimensional opinion network.

2 Basics of Network Theory

A network is a set of nodes, which can be linked by edges.[2] Both nodes and edges can have attributes, an edge for example can be weighted by a real number. Edges can be either directed or undirected, meaning that they either only point from node i to node j in a one-way like manner or that they link both nodes meaning that node j also points at node i.
For an undirected network, meaning that all edges in the network are undirected, a network density d can be defined as

$$d = \frac{L}{L_{max}} = \frac{2L}{N(N-1)},$$

where L is the number of edges and N the number of nodes included in the network[3]. In the bachelor thesis only sparse networks will be of importance, meaning that $d \ll 1$.
Another important definition is the one of subnetworks. A subnetwork is a chosen subgroup of nodes and all their links among eachother from the network. One example that is useful when looking into opinion polarization is dividing the network into subnetworks or communities including nodes having the same stance on a topic, so one could then count the number of nodes having a certain opinion and then look into the opinion distribution of the whole network.
For understanding social networks the definition of the degree of a node is needed. In an undirected network the degree k_i is the number of links a node i has. For a directed network one has to distinguish between in-degree and out-degree, being the number of incoming and

[1] Fabian Baumann, Philipp Lorenz-Spreen, Igor M. Sokolov, and Michele Starnini: Emergence of Polarized Ideological Opinions in Multidimensional Topic Spaces. Berlin, 2021.

[2] Filippo Menczer, Santo Fortunato and Clayton A. Davis: A First Course in Network Science. Cambridge, 2020. p.15

[3] Menczer et. al. p.20

outgoing links of a node.[4] The degree plays a big role in social networks because it can resemble the relation two people represented as nodes in the network have. A common feature when looking at the degree distribution in a social network is that the distribution follows a power-law $F(k) \sim k^{-\alpha}$ with $\alpha > 0$. The few nodes having a high degree are called hubs.[5]

3 The model

3.1 Opinion Dynamics

The model by Fabian Baumann et. al. is a reinforcement model[6] which means that when two agents with different opinions interact their opinions get closer to each other rather than pushing themselves away from each other. Each agent can hold a set of opinions on T topics. These topics can be related to each other because of including similar arguments which results in the correlation of the opinions of an agent on these topics. The opinions of each agent i are represented by an opinion vector

$$\boldsymbol{x}_i = (x_i^{(1)}, x_i^{(1)}, ..., x_i^{(T-1)}, x_i^{(T)}),$$

in which the components take values from $x_i^{(\nu)} \in (-\infty, +\infty)$. The sign of each component represents the agent's stance towards the topic and $\left|x_i^{(\nu)}\right|$ denotes the strength of his conviction. For the opinion of agent i to change, the agent has to interact with other agents j with $j \neq i$. All connections between agents are described by the adjacency matrix A_{ij}, where $A_{ij} = A_{ji} = 1$ means that agents i and j are connected and $A_{ij} = 0$ means that they are not. The change of the opinion on topic ν of agent i is described by the ordinary differential equation

$$\dot{x}_i^{(\nu)} = -x_i^{(\nu)} + K \sum_j A_{ij} \tanh\left(\alpha [\boldsymbol{\Phi} \boldsymbol{x}_j]^{(\nu)}\right) \tag{1}$$

which leads to N times T ordinary differential equations describing the dynamics of a network of N nodes each having a set of T opinions. In equation (1) the influence of the opinions of agent j on agent i is calculated by using the $\tanh(x)$ function which is used for lessening the impact of extreme opinions. The matrix $\boldsymbol{\Phi}$ expresses the topic overlap of all T topics. Its

4 Menczer et. al. p.23

5 Menczer et. al. p.72

6 Andreas Flache, Michael Mäs, Thomas Feliciani, Edmund Chattoe-Brown, Guillaume Deffuant, Sylvie Huet, Jan Lorenz: Models of Social Influence: Towards the Next Frontiers. Groningen, 2017. §2.9

entries are $\Phi_{u\nu} = \cos(\delta_{u\nu})$ where $\cos(\delta_{u\nu})$ is calculated by the scalar product of the basis vectors of topics u and ν, $\cos(\delta_{u\nu}) = \boldsymbol{e}^{(u)} \cdot \boldsymbol{e}^{(\nu)}$. This results in agent j's opinion on topic u influencing agent i's opinion on topic ν if topics u and ν are correlated which corresponds to their basis vectors not being orthogonal to each other.
The overall strength of social influence on each agent in equation (1) is controlled by parameter K. α is to be interpreted as the controversialness of the topics, meaning that when a topic is highly controversial $\alpha \gg 1$ in which case even moderate opinions have a big impact.

3.2 Network Dynamics

The network on which the model works is not static, but instead temporal. This means that after each iteration, all edges between agents are cut and new links are formed. Because of this the adjacency matrix is time-dependent, $A_{ij} = A_{ij}(t)$. A formation of a connection between agents i and j is either established by i contacting j or i being contacted by j.
Each agent has a certain activity a_i which corresponds to his probability to become active in each iteration. The activity distribution follows a power-law distribution with exponent $-\gamma$. If an agent is activated he contacts m other agents. The probability of agent i contacting agent j is dependent on their opinion distance

$$p_{ij} = \frac{d(x_i,x_j)^{-\beta}}{\Sigma_j d(x_i,x_j)^{-\beta}} \tag{2}$$

where β controls the homophily[7] of the network meaning that for large β agents mostly connect to agents that are similar to themselves (strong homophily) and for small β agents are more likely to connect to others that are not similar to themselves (weak homophily). The distance d between two agents is induced by the scalar product

$$\boldsymbol{x}_i \cdot \boldsymbol{x}_j = \boldsymbol{x}_i^T \boldsymbol{\Phi} \boldsymbol{x}_j = \sum_{u,\nu} x_i^{(u)} x_j^{(\nu)} \cos(\delta_{u\nu}) \tag{3}$$

resulting in the distance of two agents being

$$d(x_i,x_j) = \sqrt{(\boldsymbol{x}_j - \boldsymbol{x}_i) \cdot (\boldsymbol{x}_j - \boldsymbol{x}_i)}. \tag{4}$$

[7] McPherson, M., Smith-Lovin, L., Cook, J. M. (2001). Birds of a Feather: Homophily in Social Networks. Annual review of sociology, 27, 2001. p.415–444.

When a connection between agents i and j is established at time t the adjacency matrix is updated to $A_{ij}(t) = A_{ji}(t) = 1$.

4 Implementation

In the following the implementation of the model in Python 3.7 is presented for the two dimensional case. At first an empty array with N rows and 3 columns is created. Each entry of the array represents an agent, where the first two entries of the agent represent her opinion on two distinct topics and the third entry is her probability to become active. Now each agent‘s entries are initialized. The opinion on each topic is drawn from a Gaussian with a mean of 0 and a standard deviation of $\sqrt{2.5}$ and the activity is drawn from a power-law distribution with exponent $-\gamma$. Since a power-law distribution with a negative exponent is undefined at zero the activities are not sampled from $[0,1]$ but rather from $[\epsilon_1,1]$ where $0 < \epsilon_1 < 1$. The activities and the starting opinions of the agents are saved.

Now the network dynamics are going to be simulated. This happens in a while-loop that goes on until a pre-determined number of iterations is reached.

As a first step the network for the current iteration is created. Here an adjacency list A of dimensions $N \times N$ is initialized which is filled with the placeholder value $N + 1$. Also a counter-array C of dimension N is created to access the adjacency list. Now a for-loop iterates over each agent and an equally distributed random number on $[0,1]$ is drawn for each agent. If this random number is smaller than the agent's activity the agent is activated and connects to m randomly picked agents. These agents are picked with a probability determined by equation (2) which is made possible by an array that includes the normalized probabilities for drawing each pair p_{ij} with $i \neq j$ that is passed on to numpy.random.choice. The connected agents are picked without replacement so that agent i can only connect to each agent once. The connected agents are now listed in the adjacency list by going over all of the picked agents. If agent i connected to agent j the list is updated as $A[i][C[i]] = j$ and $A[j][C[j]] = i$ and the counter array is updated as $C[i] = C[i] + 1$ and $C[j] = C[j] + 1$. This way the adjacency list doesn't only save which agents an agent actively connected to but also by which agents an agent was contacted by.

After the network is formed the opinion dynamics take place by numerically integrating equation (1) via the classic Runge-Kutta method[8]. Since here an adjacency list is used instead of

[8] Geeta Arona, Varun Joshi and Isa Sani Garki: Developments in Runge–Kutta Method to Solve Ordinary Differential Equations. Phagwāra, 2020. p. 197.

an adjacency matrix the sum in the differential equation doesn't summarize over all agents but rather over all entries of the adjacency list A that are not $N+1$.
For community detection one needs to retrieve a time integrated network from the temporal network. This can be done by forming an integrated adjacency matrix over a certain last number of iteration steps of the network. In order to do this one has to create a matrix B with dimensions $N \times N$ that is filled with zeros. When reaching the last iterations of the network B can be updated by going over the iteration's adjacency list A and updating B with $B_{ij} = 1$ if agents i and j were connected in the network. Thus B saves all connections established within the last iterations of the activity-driven network so that now a Graph can be created from B on which community detection can be performed on.
After completing the network dynamics the last opinions of all agents are saved to a csv-file together with the activities and starting-opinions of the agents. Optionally B can be exported to a csv-file as well.
In figure 1 final opinion distributions for the cases of consensus, polarization and ideology can be seen.

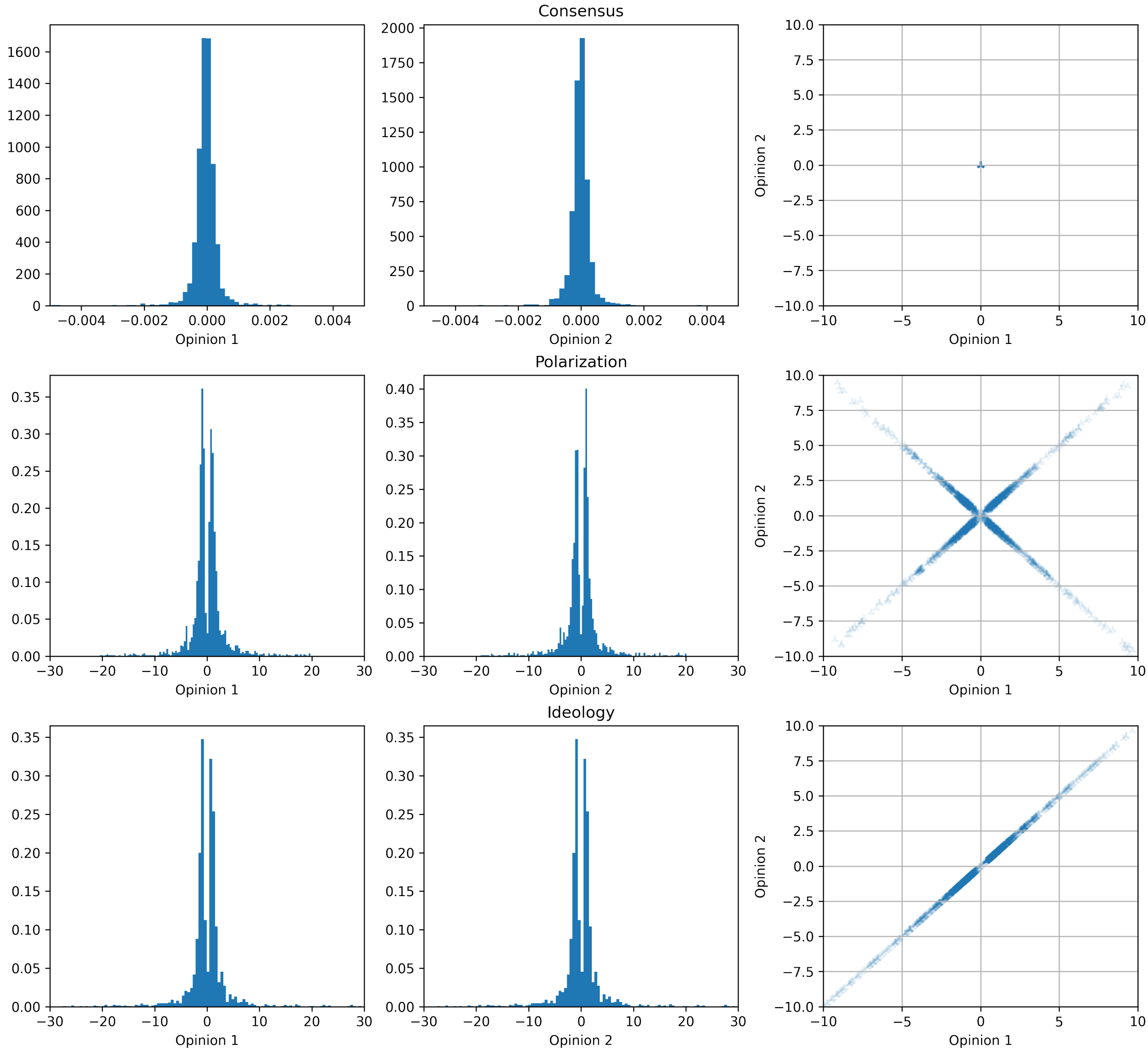

Figure 1: In this figure the distributions of agent's opinions towards two topics and their places in opinion-space are shown for the cases of consensus ($\alpha = 0.05$, $\delta = \pi/2$), polarization ($\alpha = 3.0$, $\delta = \pi/2$) and ideology ($\alpha = 3.0$, $\delta = \pi/4$). The first two columns of figures show the agent's opinion distributions towards topics one and two. In the third column the place of agents in opinion-space can be seen. In the plots of the opinion-space the nodes have a certain transparency so one can see where agents are concentrated. Simulations were performed for $\gamma = 2.1$, $K = 3$, $\epsilon_1 = 0.01$, $N = 2500$, $m = 10$ $\beta = 3.0$.

5 Detection of Polarization

The detection of whether consensus, polarization or ideology is present in a final opinion distribution is of importance since looking at what different manipulations done to the model do to the parameter-space of its mean-field approximation[9] helps to understand the effect of the manipulation. In the following two classification-methods will be discussed.

5.1 Community Classification

The first way of classifying distributions is by looking at the communities formed in the network and their position in the opinion-space. Since the network of the model is a temporal one and since only active nodes connect to a small number of other agents community detection on the network formed in the last iteration would only lead to some small communities and many nodes forming a community on their own. In order to find meaningful communities the approach of Baumann et. al.[10] was followed. Here an integrated network was formed by saving all connections made over the last 70 time-steps of a simulation, as was already described in the former section. On this integrated network communities can be detected by using the Louvain algorithm[11].

Once communities were detected the ones that were too small, namely communities with a number of nodes smaller than $N/1000$, were filtered out. The mean angle of nodes being in a community was then calculated for each community. Then it was checked whether these mean angles took values of $\pi/4$, $3\pi/4$, $-3\pi/4$ or $-\pi/4$, each with a tolerance of $\pm 30°$. If the mean angles were distributed over all 4 angles the distribution was classified as polarization. If the mean angles were found to be distributed only around $\pi/4$ and $-3\pi/4$ or $3\pi/4$ and $-\pi/4$ the opinion distribution was classified as ideology. In the case of not all mean angles being able to be categorized as being close to one of the four angles the distribution was classified as consensus.

For reproducing the phase-space of the mean-field approximation with this classification method three simulations were classified per parameter pair for a parameter range of $\alpha \in \{0,0.1,...,3.9,4.0\}$ and $\cos(\delta) \in \{0,0.1,...,0.9,1.0\}$. The results can be seen in figure 2. While there is some resemblance between the two phase-space plots the numerically calculated one has a lot of

[9] Baumann et. al. p.6-7

[10] Baumann et. al. p.8-9

[11] Andrea Lancichinetti1, and Santo Fortunato: Community detection algorithms: a comparative analysis. Torino, 2010. p.3-4

fluctuations and wrongly classifies many polarized states as consensus and many consensus states as polarized or ideological.

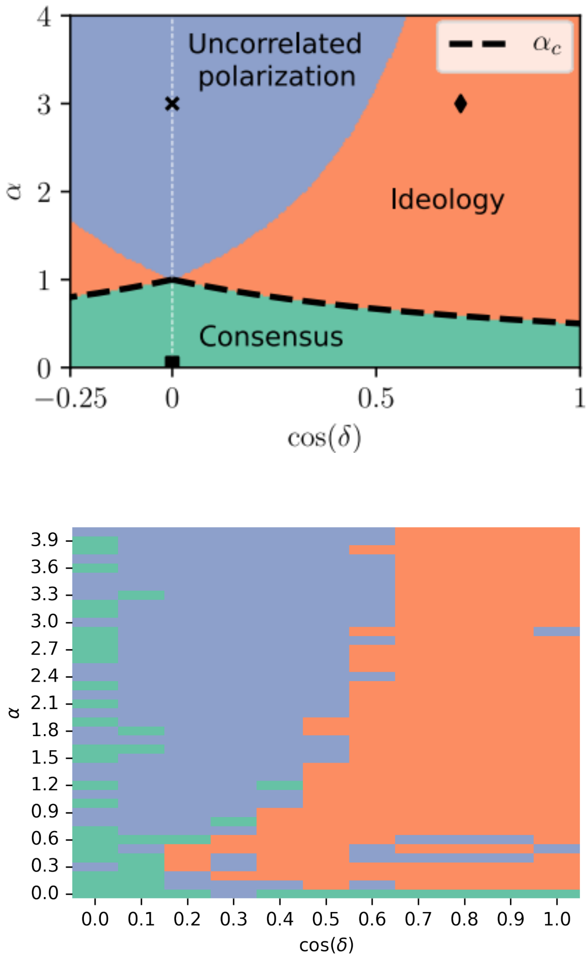

Figure 2: In the top figure the analytically calculated phase-space of the model can be seen [Baumann et. al. p.7]. Below is the numerically calculated phase-space which was created using community classification. Simulations were performed for $N = 2500$, $m = 10$, $\beta = 5.0$, $\gamma = 2.1$, $K = 3.0$ and $\epsilon = 0.01$.

5.2 Agglomerative Classifier

An improvement to the Community Classification is a classifier based on an agglomerative algorithm designed to find clusters of nodes. This algorithm is based on the idea of merging agents to one agent if their opinion difference is below a certain threshold and then creating a

new agent between the former agents with a bigger weight, meaning that the new node doesn't move as far when merging with another a single node.
Again for a two-dimensional topic-space an array of the size $N \times 3$ is created. The first two entries are the opinions of an agent which are drawn from the final opinion distribution that is to be analyzed. The third entry is initialized with 1. This entry resembles the number of nodes that are already merged within a node. The algorithm begins with a while-loop that runs until a certain runtime is reached. At the beginning of each iteration, an array I containing all normal numbers from 0 to $N_t - 1$ is created, with N_t being the number of agents at iteration t. Now a for-loop loops over all agents and it is checked whether the current agent is included in I. If it isn't the agent is skipped. For each agent another agent is picked randomly where now the probability of picking an agent is uniform for all agents and it is checked whether their opinion distance, which is the Euclidean distance in the opinion space, is below a threshold η. This threshold should be roughly equal to the absolute position of a peak of opinions in a polarized distribution. If the distance is smaller than η the nodes are merged and a new node is created. Its position in the opinion-space is calculated by

$$\vec{o} = \frac{n_i \cdot \vec{o_i} + n_j \cdot \vec{o_j}}{n_i + n_j} \tag{5}$$

where n_i is the number of nodes that were merged within node i. Because of this equation nodes that consist of many nodes don´t move much anymore when they merge with single nodes but rather draw them towards their position resulting in the "heavy" nodes sitting roughly where the peaks of the opinion distribution are if η is chosen correctly. After creating the new node the old two nodes are deleted and their numbers are deleted from I. When the for-loop ends the remaining nodes that were not merged and the newly created ones are put together in a new array. For the new array at time-step t $N_t <= N_{t-1}$ is always the case. Thus N_t converges until only nodes are left that can't be merged since their distance is bigger than η.
With the final nodes produced by the agglomerative algorithm one can classify the distributions into consensus, polarization and ideology. For this two things are important: The angle at which the final nodes lie in the opinion space and how far they are from the center. The angle here is the angle between an opinion vector and the x-axis.
At first each final node is categorized into either having an angle of $\pi/4$, $3\pi/4$, $-3\pi/4$, $-\pi/4$ all with a tolerance of $\pm 30°$. After that the final nodes are also categorized into indicating polarization or consensus by putting them into the consensus-category if all their opinion's

absolute is below a threshold θ and into the polarization-category else. Now it is checked whether more nodes are in the consensus-category or in the polarization-category. This is done by summing up all nodes merged within the final nodes in each of the two categories and comparing the amounts. If more nodes are within the consensus category the distribution is classified as consensus. If that isn´t the case, the angles are checked. If there are final nodes only in either $\pi/4$ and $-3\pi/4$ or in $3\pi/4$ and $-\pi/4$ the distribution is classified as ideology. If there are nodes in each angle-area the distribution is classified as polarization.

Since the final nodes are randomly formed and since the positions and node distributions of the final nodes fluctuate multiple runs of the agglomerative algorithm are done for each data set and the state that was classified most often is chosen as the state of the data set. If two states were classified equally often consensus is chosen over polarization and ideology and polarization is chosen over ideology.

At the top of figure 3 an example of the final nodes, that include at least 200 agents (for $N = 2500$), produced by the agglomerative algorithm for a polarized state are presented. The zones that are used to determine where a final node lies in the opinion-space are colored in. The presented example is classified as polarization since at least one final node is present in each of the four zones and since none of the final nodes are within the zone for consensus which can be seen at the bottom of of figure 3. The example on the bottom is one that is close to transitioning to consensus which is why final nodes emerge not only in the four corners but also in the middle of opinion-space. Since a final node is found in the zone around the center it is counted whether the nodes outside of the zone include more agents than the node in the middle. Since this is the case the distribution is classified as polarized. If more agents would be included in the center final node the distribution would be classified as consensus.

In figure 4 a comparison of the analytical phase-space and the one found by using the agglomerative classifier can be seen. The agglomerative classifier is much less prone to fluctuations than the community classifier. What is important is that the agglomerative classifier finds much more consensus than what is expected from the analytical result. This discrepancy can be explained by polarization and consensus not being well defined. Especially at the phase-transition from consensus to polarization or ideology it is up to discussion whether a distribution having two peaks but many nodes between them is called consensus or polarization[12].

To further elaborate why the agglomerative classifier finds much more consensus than the analytical result different distributions around the phase-transitions and their classification were

[12] Namkje Koudenburg, Henk A. L. Kiers and Yoshihisa Kashima: A New Opinion Polarization Index Developed by Integrating Expert Judgements. Groningen/Melbourne, 2021

plotted in figure 5. The according opinion-spaces can be found in figure 6. As can be seen in the figures consensus is reached for at least $\alpha \leq 1.5$ and $\cos(\delta) = 0.0$. For $\alpha = 1.8$ two peaks are visible but there are still many nodes between them. Thus the distribution can be seen as both polarization and consensus, depending on what criterion one uses. When increasing α or $\cos(\delta)$ the peaks grow more elaborate which makes the classification into polarization or ideology more clear.

The difference between the analytical and numerical results can also be traced back to $N \to \infty$ and $\beta \gg 1$ and thus the conditions of the mean-field approximation not being fulfilled perfectly since simulations were performed for $N = 2500$ and $\beta = 5$. It is possible that for bigger N the peaks near the phase-transition would be formed more clearly which would result in the classifier classifying polarization for lower values of α than it currently does.

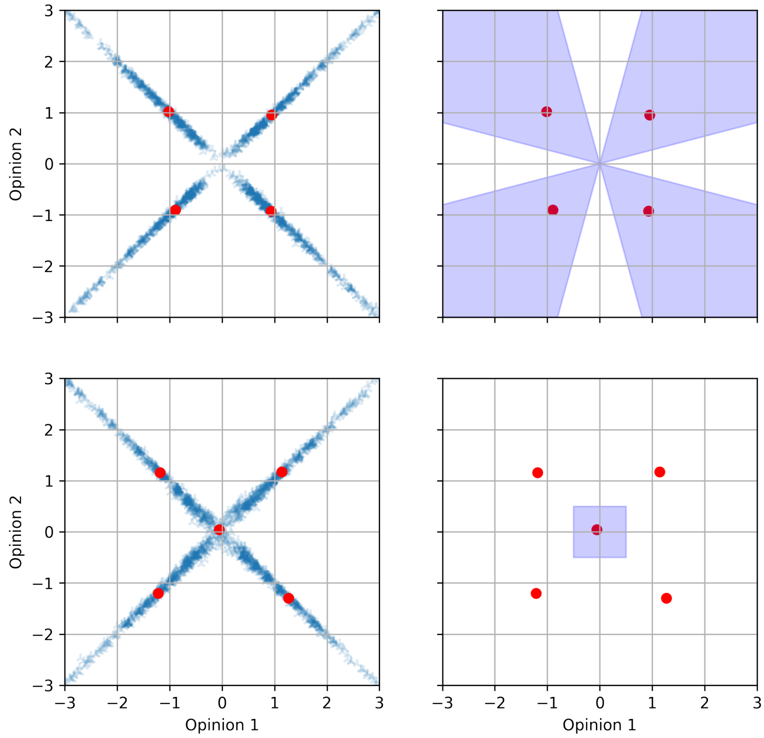

Figure 3: The top two figures show the final nodes found by the agglomerative algorithm for a polarized state ($\alpha = 3.0$). On the left the original agent distribution with the final nodes in red can be seen while on the right the final nodes and the zones used for classifying in which area of the opinion-space they are in are depicted. The bottom two figures show the final nodes for a state with $\alpha = 1.8$. On the left again the agent's original distribution and the final nodes found by the algorithm can be seen while on the right only the final nodes and the zone in blue used for classifying whether nodes contribute to a classification of consensus or not are depicted.

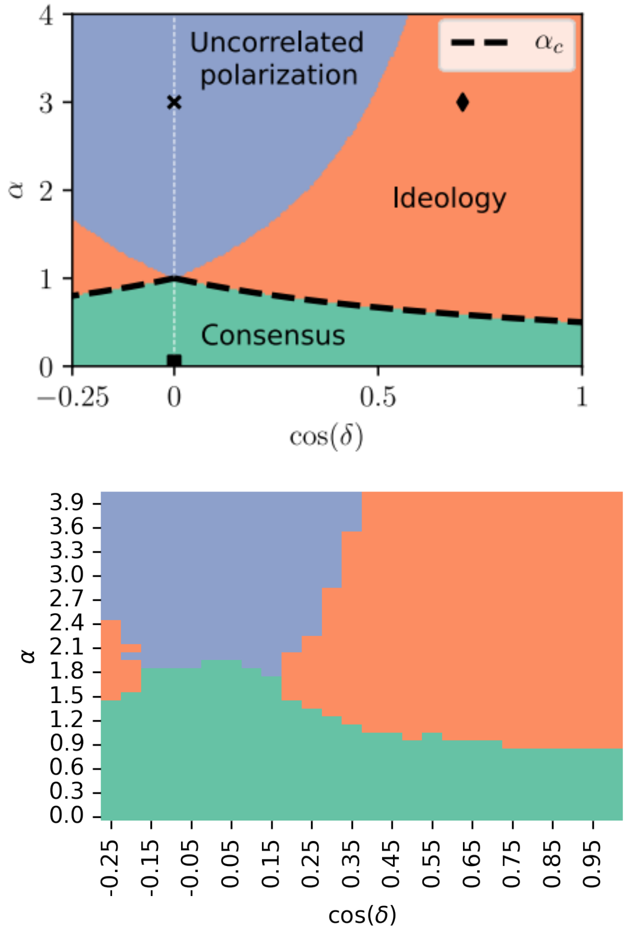

Figure 4: In the top figure the analytically calculated phase-space of the model can be seen [Baumann et. al. p.7]. Below there is the numerically calculated phase-space which was created using agglomerative classification. Simulations were performed for $N = 2500$, $m = 10$, $\beta = 5.0$, $\gamma = 2.1$, $K = 3.0$ and $\epsilon = 0.01$. For each pair of parameters 3 simulations with 5 agglomerative algorithm runs per simulation were done.

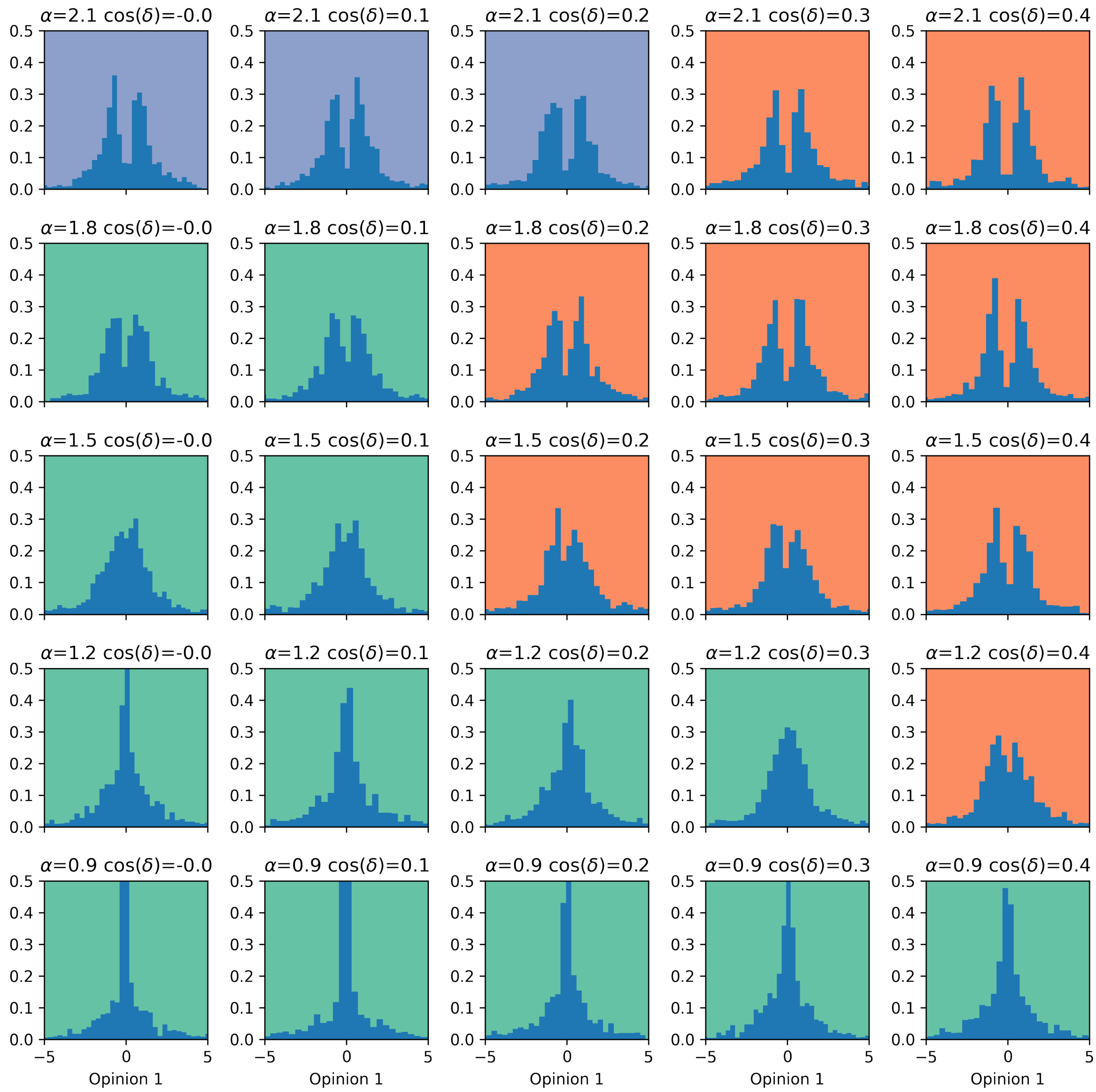

Figure 5: Different final opinion distributions for diverse values of α and $\cos(\delta)$ can be seen. Their background is colored according to how they were classified by the agglomerative classifier, namely green meaning consensus, blue representing polarization and orange standing for ideology.

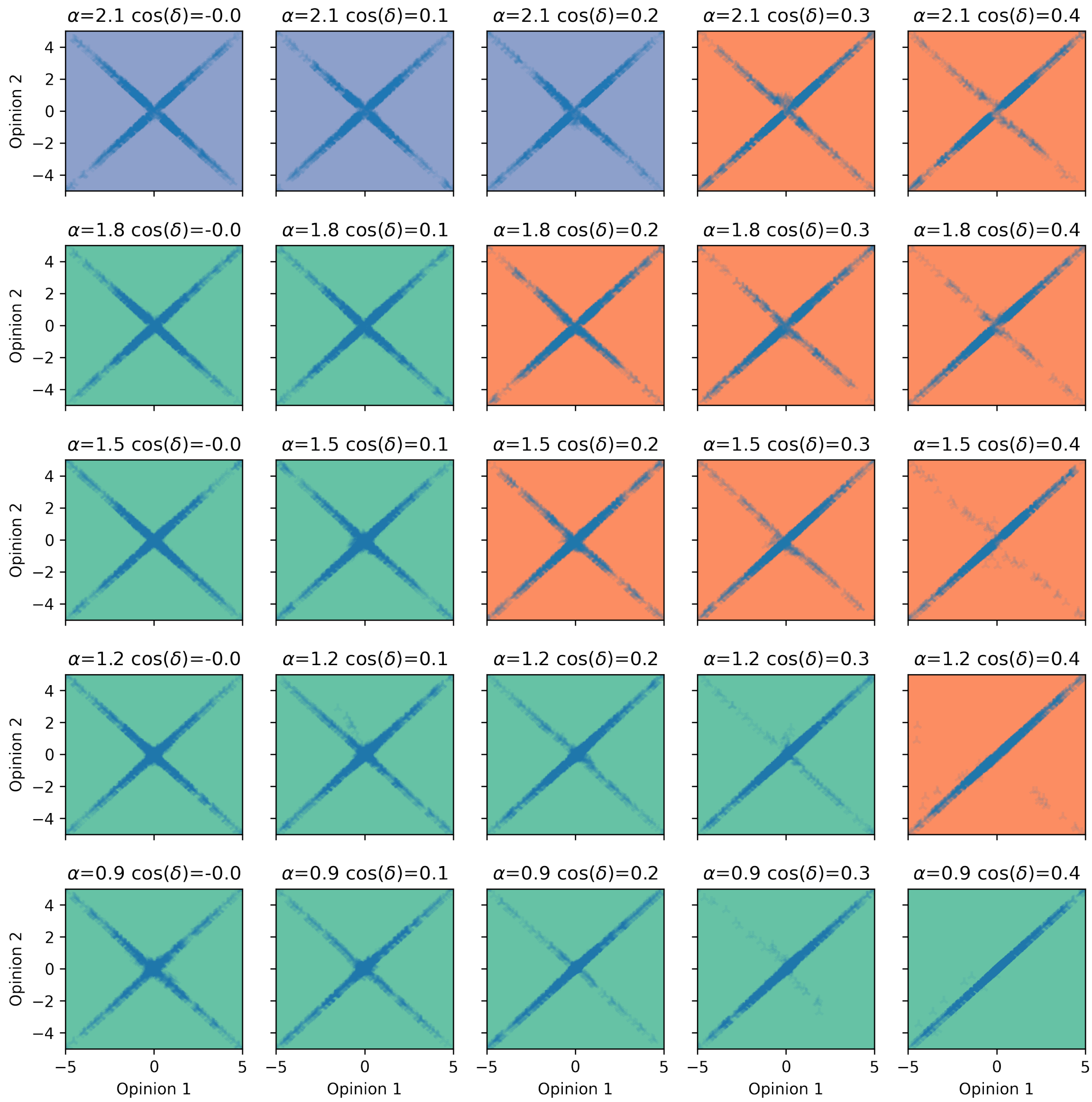

Figure 6: Different final opinion-spaces for diverse values of α and $\cos(\delta)$ can be seen. Their background is colored according to how they were classified by the agglomerative classifier, namely green meaning consensus, blue representing polarization and orange standing for ideology.

References

[1] Andrea Lancichinetti and Santo Fortunato: Community detection algorithms: a comparative analysis. Torino, 2010.

[2] Andreas Flache, Michael Mäs, Thomas Feliciani, Edmund Chattoe-Brown, Guillaume Deffuant, Sylvie Huet and Jan Lorenz: Models of Social Influence: Towards the Next

[3] Fabian Baumann, Philipp Lorenz-Spreen, Igor M. Sokolov and Michele Starnini: Emergence of Polarized Ideological Opinions in Multidimensional Topic Spaces. Berlin, 2021.

[4] Filippo Menczer, Santo Fortunato and Clayton A. Davis: A First Course in Network Science. Cambridge, 2020.

[5] Geeta Arona, Varun Joshi and Isa Sani Garki: Developments in Runge–Kutta Method to Solve Ordinary Differential Equations. Phagwāra, 2020. Frontiers. Groningen, 2017.

[6] McPherson, M., Smith-Lovin, L., Cook, J. M. (2001): Birds of a Feather: Homophily in Social Networks. Annual review of sociology, 27, 2001.

[7] Namkje Koudenburg, Henk A. L. Kiers and Yoshihisa Kashima: A New Opinion Polarization Index Developed by Integrating Expert Judgements. Groningen/Melbourne, 2021.