Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Lecture with Computer Exercises: Modelling and Simulating Social Systems with MATLAB

Project Report

Opinion Convergence

Lukas Böke Melanie Gut Anne Lüscher

Zürich
Dezember 2013

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of Originality

This sheet must be signed and enclosed with every piece of written work submitted at ETH.

I hereby declare that the written work I have submitted entitled

Opinion Convergence

is original work which I alone have authored and which is written in my own words.*

Author(s)

Last name	First name
Böke	Lukas
Gut	Melanie
Lüscher	Anne

Supervising lecturer

Last name	First name
Woolley	Olivia
Kuhn	Tobias

With the signature I declare that I have been informed regarding normal academic citation rules and that I have read and understood the information on 'Citation etiquette' (http://www.ethz.ch/students/exams/plagiarism_s_en.pdf). The citation conventions usual to the discipline in question here have been respected.

The above written work may be tested electronically for plagiarism.

Zürich, 9.12.13

Place and date

Signature

*Co-authored work: The signatures of all authors are required. Each signature attests to the originality of the entire piece of written work in its final form.

Print form

Agreement for free-download

We hereby agree to make our source code of this project freely available for download from the web pages of the SOMS chair. Furthermore, we assure that all source code is written by ourselves and is not violating any copyright restrictions.

Lukas Böke Melanie Gut Anne Lüscher

Table of Content

1. Abstract

A model for opinion convergence in a simple social system was developed and implemented as a simulation using `MATLAB`. The code was based on an existing model as suggested by Laguna, Abramson and Zanette.[1] Additional parameters were introduced in order to make the model more realistic and dynamic. The parameters were then modified and simulations were conducted for each modification with the goal of finding the qualitative impact of each parameter on the system.
Other papers and models were consulted: Holme, Newman: Nonequilibrium phase transition in the coevolution of networks and opinions arXiv:physics/0603023v3
Helbing: A Mathematical Model for the Behavior of Individuals in a Social Field arXiv:cond-mat/9805194v1

2. Individual contributions

The project was done in a collaborative manner by all three group members. In the beginning, ideas were collected and discussed in the group and the mathematical model was established. Then, a first version of the code was written and tested. From that point onwards, the work was roughly subdivided to the follow main responsibilities:
Lukas Böke: Implementation in `MATLAB`, improving and altering the code
Anne Lüscher: Data analysis/results, writing of report
Melanie Gut: Presentation
However, we always counter-read each other's work and were always informed, what the others were doing in order to keep an efficient workflow. Obviously, the code had to be understood by everyone to complete the work.

3. Introduction and Motivations

Opinion formation is a point of interest in various fields; especially in topics such as politics and public affairs it is of great importance. Therefore, it is not surprising that the realistic modelling of the dynamic, time-dependent process of opinion formation and development has become an interesting tool to predict trends. This is made possible by using simulation methods based on mathematical models, which make the necessary assumptions.
But how can such a complex process be packed into a simple formula and then be converted into code, until it finally leads to a result? This was one of the questions we aspired to investigate out of personal interest. Furthermore, we wanted to better understand how opinion groups form and how peer influence can be included in a mathematical model.
While studying related research papers we came across various approaches and noticed two things: Firstly, the models were often rather complex and thus not easy to implement. On the other hand, we found simpler ideas, which seemed to lead to valuable qualitative results none the less. We then found ourselves motivated to strike a balance between simplicity and complexity. In concrete terms: The idea came up to use a simple existing model and conduct some simulations in order to see how it works. In a second step, adding some factors would refine our approach.

Our main research goal was to find out, if our refinements to the model would turn out be a valuable supplement. In concrete terms, we wanted to investigate whether the altered approach would lead to different results, and if so, to what extent and for what reasons.
The first step was converting the model/formula used by the authors of our main reference[1] into code and thus implementing it in `MATLAB`. In order to test the programme, we tried to reproduce the data described in [1] and see if our simulations would lead to comparable results. The aim was to better understand how each of the parameters influences the convergence of opinions in our simplified social system.

4. Description of the Model

4.1. Basic Model

Our model is based on the approach used in [1]: We take N agents with a random opinion, which is represented by a number from [0, 1]. Now we repeatedly choose pairs of agents. Two of these agents interact, if the difference between their opinions is smaller than a previously defined threshold u. After one interaction between a pair of agents, the opinions of two agents a and b are described as follows (adopted from [1]):

$$x_a(t+1) = x_a(t) + c * \left(x_b(t) - x_a(t)\right)$$
$$x_b(t+1) = x_b(t) + c * \left(x_a(t) - x_b(t)\right)$$

x_a and x_b describe the opinion of agents a and b, respectively, as a function of time. The opinion at time t+1 differs from the opinion at time t if – and only if – an interaction between the agents takes place, which is the case if abs ($x_a(t)$ - $x_b(t)$) < u is true. If this condition is met, the agents will adjust their opinions as much as the convergence parameter c allows. This parameter takes on values from [0,0.5] and controls, how much the opinions of two agents approach each other after an interaction. In the extreme case of c = 0.5, the agents' opinions will meet in between the two initial values, and, consequently, converge completely, so that only one opinion prevails. If c=0, there will be no adaption at all, and both opinions remain unchanged. Summarily, c is of great importance to the model, as it has a substantial influence on the outcome of the simulation, as will be discussed later on. Therefore, choosing its value carefully is vital.

4.2. Refined Model 1: Introduction of parameter "Charisma"

We first added one new factor to our model, which we called "The Charisma" of an agent, for which we use the letter γ. This parameter, too, is assigned randomly. Our refined version blends into our basic model if two agents have the same γ. If two agents have different γ_i, meaning $\gamma_a \neq \gamma_b$, the opinion of the agent with the smaller γ will change more than the opinion of the agent with the bigger γ.

$$x_a(t+1) = x_a(t) + \frac{2 * c * \gamma_b}{\gamma_a + \gamma_b} * \left(x_b(t) - x_a(t)\right)$$
$$x_b(t+1) = x_b(t) + \frac{2 * c * \gamma_a}{\gamma_a + \gamma_b} * \left(x_a(t) - x_b(t)\right)$$

This was an important step towards a more realistic model, as the agent with greater charisma changes the opinion of his interaction partner more, than he adapts his own opinion. The exact value of how much the opinions change still depends on the convergence parameter, but is also determined by the two charisma values. Furthermore, a factor of two is included, which is required for the formula to be consistent with the basic model in the case where $\gamma_a = \gamma_b$.

4.3. Refined Model 2: Introduction of "Circle of Trust"

As soon as the charisma factor was included in the code, yet another parameter was added in order to account for the influence of familiarity between agents. The thought behind this was that peer influence depends on how well the agents know each other. It is intuitively clear that any agent will have a tendency to be more susceptible to a friend's or family member's influence than to a stranger's. This problem was not easy to convert into a formula and we finally solved it using a physical approach by creating a two-dimensional "trust space".

In the new model, every agent is assigned a random point in $[0,1] \times [0,1]$. The (mathematical) distance between two agents in this square corresponds to how well two agents know each other. Thus, we can define "circles of trust" around one agent with different radiuses, where the change of opinion is now dependent on "how far away" the two agents are from each other within the social system. We assume that the closer two agents are in space, the better they know each other, which then determines the convergence of their opinions.

5. Implementation

5.1. Implementation of the Basic Model

The implementation in MATLAB is quite simple and straightforward: For the opinions of the agents we have a matrix M, where the rows correspond to the opinions of one agent at different times, and the columns correspond to all opinions of all agents at a given time t. To randomly choose the pairs of interacting agents, we first shuffle the rows of the matrix M. After that we split the matrix in two halves. Now the i-th agent in the first half is paired with the i-th agent in the second half. We then check if the difference of two agents opinions are smaller than our threshold u. We then use the formulas above to compute the new opinions of the agents.

5.2. Implementation of the new Parameter "Charisma"

For the charisma of an agent we use a vector, with the rows corresponding to one agent. Here, the rows of this vector correspond to the rows in our matrix M. As one can easily see, the charisma of one agent does not change over the time. To shuffle the rows of the matrix/vector, we create a vector with a randomly computed order, we then arrange the rows of the matrix according to this vector.

5.3. Implementation of the “Circles of Trust”

We first implemented the position by assigning a random position to every agent. For this, we just added another matrix, where every row corresponds to the coordinates of one agent. The rows of this matrix are shuffled in the same way as we did that with the charisma.
We then implemented this model slightly different to get a homogenous distribution of the agents in $[0,1] \times [0,1]$: The agents are positioned on a grid represented by a matrix, here the entries of the matrix where the indices of the agents opinion in the matrix M at the beginning. We therefore used another vector that at the beginning contained the not yet shuffled indices used in the grid. We then shuffle the rows of this vector in the same way as we shuffle the matrix of opinions M, so that we know what permutation we have to use to get the original order of the rows of M. These plots however were not used in this paper, we uploaded the code to our Github repository [2] anyhow.

6. Simulation Results and Discussion

6.1. Conducting the Simulations

In the different runs we tried to determine the impact of each factor on the outcome of the simulation. In order to test the reliability each combination was tried several times to verify whether the outcome was reliable or only a coincidence. The plots were then analysed with focus on the following aspects, relating to our initial research questions:

1. How quickly (after how many iterations) is the equilibrium reached?
2. How many opinions prevail? Is there complete convergence observed? Is there any cluster formation?
3. Are the results consistent with our predictions, as well as compatible with the findings of our reference[1]?
4. Are the outcomes both reproducible and reliable?

The programme was run with different combinations for the parameters. The influence of u and c is equal in the two different versions, that is, they have the same influence in combination with the charisma parameter and the circle of trust.
It has to be mentioned, that in order to completely understand the connection and correlation between all the parameters, a lot more simulation runs would have had to be conducted; it is due to time shortage that we could only find superficial results. However, we do believe that they illustrate the abilities as well as the limits of our model well in a qualitative matter.

6.2. Influence of the Number of Agents

Unfortunately, the number of agents had to be strongly restricted due to limited workspace. The simulation was run with 100, 500 and 1000 agents, while in our reference[1], the number of agents was significantly higher (by several orders of magnitude). With only 100 agents, the results were not very reliable. The distributions often seemed to be random and the results were not reproducible. This is obviously due to the randomly assigned opinions and values for γ. As could also be seen in the plots, the distribution of the initial opinions between 0 and 1 is sometimes more uniform and sometimes less so. With growing number of agents, the distribution becomes more reliable and so do the results. It became clear, that 500 agents are a good compromise. The initial opinions are equally enough distributed, but the time for a single calculation is still within a reasonable range.
This is illustrated by Figs 1.-4. In Fig.1 the simulation was conducted with 1000 agents. The assignment of initial opinions is perfectly uniform and in the end only one opinion prevails (converges to 0.5). With 500 agents (Fig.2) the simulation still looks similar, so does the initial distribution of opinions. This is different if just 100 agents are used. As a consequence of the small number of agents, there is a statistically induced “pre-clustering" of opinions when the initial values are assigned. 100 agents are simply not enough to reach an initial situation, where the random opinion values are perfectly regularly distributed. From Fig. 3, Fig.4 it can be deducted that if those pre-clusters are in the peripheral opinion space (either close to 0 or to 1), two main opinions will prevail, while complete convergence can be observed if the pre-clusters are closer to 0.5. The other parameters obviously also have an influence on the process. In other terms, with the same

pre-clusters formed in the beginning, due to the random assignment of opinions there are still different results to be expected, depending on the threshold and convergence parameter, but even more so depending on the also randomly assigned values for gamma. Summarily, with 100 agents, the simulation results were not reliable and our explanations could not be verified to a hundred percent. This illustrates, that even our very simple and straightforward model already has some sort of unexpected complexity. However, this could also support our model to be a better depiction of reality, since in real social systems there are actually even more such unforeseeable parameters. Even though, the results were less predictable, running the programme with a smaller amount of agents proved itself to be an interesting tool, as that way the opinion behaviour of single agents is perfectly traceable.

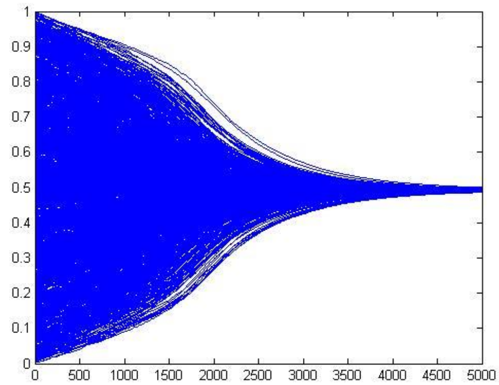

Figure 1: agents: 1000, with c: 0.002; u: 0.35, 5000 iterations

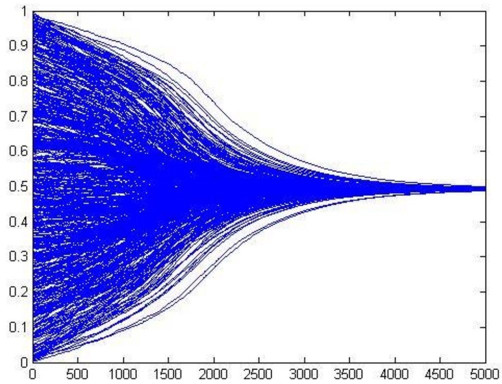

Figure 2: agents: 500, with c: 0.002; u: 0.35, 5000 iterations

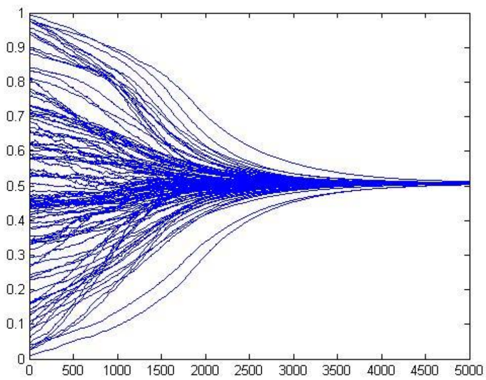

Figure 3: agents: 100, with c: 0.002; u: 0.35, 5000 iterations

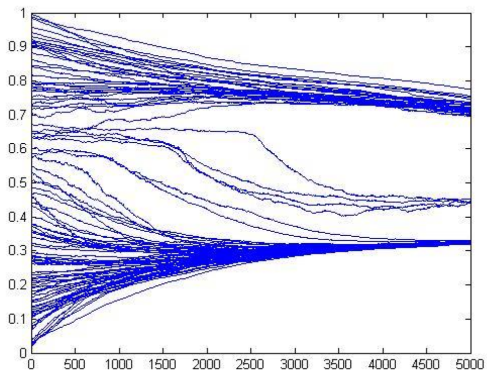

Figure 4: agents: 100, with c: 0.002; u: 0.35, 5000 iterations

6.3. Influence of convergence parameter c

The convergence parameter is chosen arbitrarily, which may be one weakness of our model. It's only purpose is to control the rate of two opinions merging - in other terms, by how much two agents approach their opinions after an interaction. This obviously influences, after how much time the equilibrium is reached. This becomes interesting, if the time (or the number of interactions) is set to a certain value. Depending on c, the number of opinions at that point will be very different. For large c, meaning values close to 0.5, interesting anomalies occur. As the convergence happens so quickly, sometimes several opinions prevail, while the main cluster goes fast towards equilibrium. There is one main opinion plus a few minorities consisting of only a very small number of agents. Our explanation is that there is a stochastic chance – however small – that some agents will within the first few interactions only meet agents, with which the threshold u is not reached or from whose opinion they do not differ much. Consequently, they keep their opinion or only change it slightly. Because of the large value of c, the main part of convergence has already happened after relatively few interactions, and at that point, there are no agents left within the threshold – the logical consequence of which is the prevailing of all the opinions left at this point. This was already found by Laguna, Abramson and Zanette in their version of the model. Our assumption is supported by the fact, that minorities become less frequent if either the threshold is made larger or c made smaller.

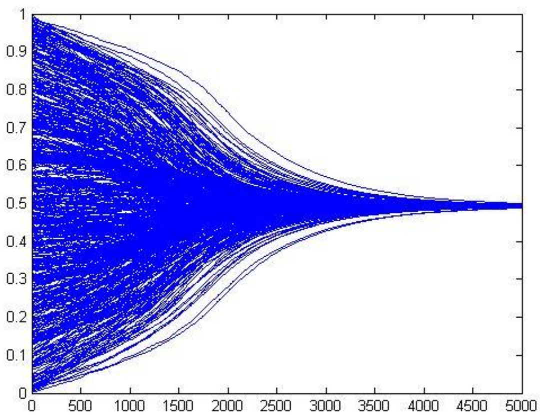

Figure 5: c: 0.002 with u: 0.35; 5000 iterations; 500 agents

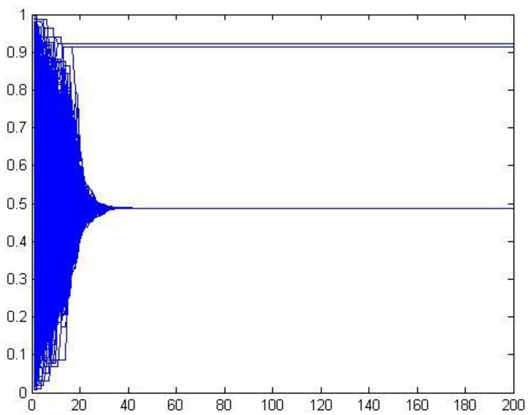

Figure 6: c: 0.5 with u: 0.35; 5000 iterations; 500 agents

6.4. Influence of threshold u

The threshold's value is surprisingly influential. While the convergence parameter has only an impact if a rather extreme value is chosen, the threshold changes the result already with moderate modifications. In the figures below, the influence of the parameter u is shown to be dramatical. All of the graphs show the stand of opinions after 5000 iterations and with different values of u. All the other parameters were kept constant. These results confirm what could already be deducted from the model in advance: The higher the interaction threshold – or the smaller u – the less convergence will be observed, or, specifically, the longer it takes until an equilibrium is reached. This makes sense, since the chance of two agents with an opinion difference of less than, say, 0.1, is a lot lower than the one of an interaction between two agents within a difference of 0.5. For a value of

u=1.0, each interaction will inevitably lead to an adaption of the agents' opinion. Thus, maximal convergence is achieved.
To determine the impact of the model's other parameters, u was set to a value of 0.35. This is again an arbitrary choice, but it seemed legit on a common sense basis and led to reasonable results within the usual 5000 iterations.

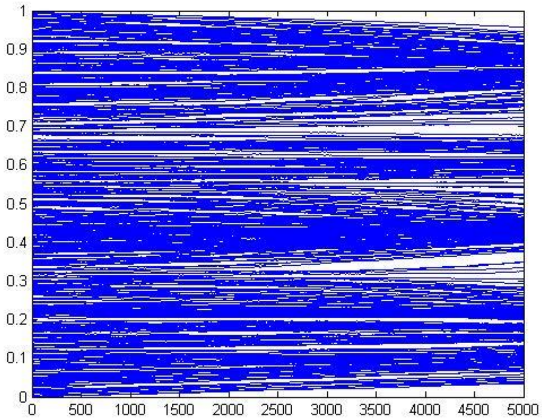

Figure 7: u: 0.1, with c: 0.002; 5000 iterations; 500 agents

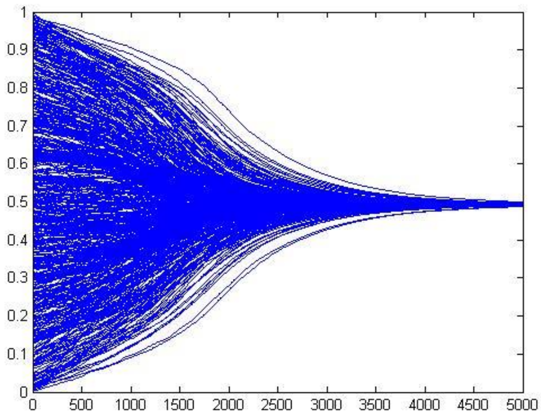

Figure 8: u: 0.35, with c: 0.002; 5000 iterations; 500 agents

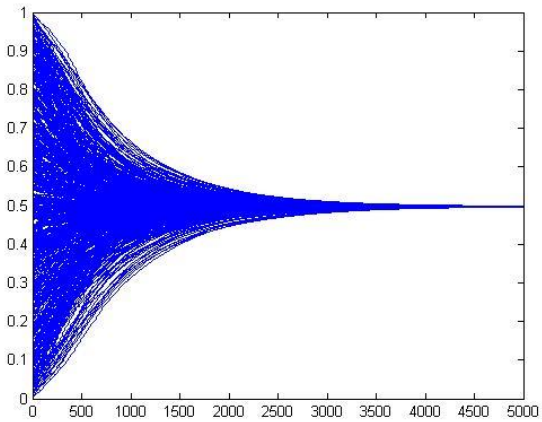

Figure 9: u: 0.6, with c: 0.002; 5000 iterations; 500 agents

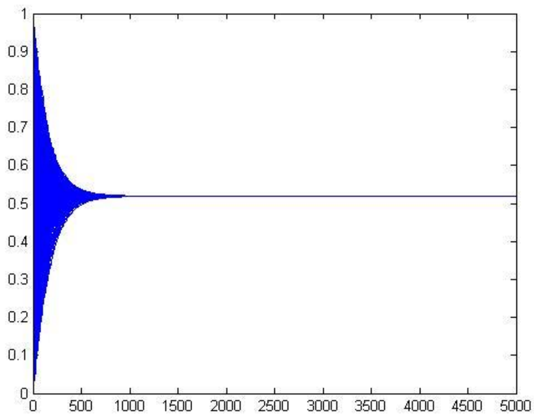

Figure 10: u: 2.0, with c: 0.002; 5000 iterations; 500 agents

6.5. Influence of γ

The charisma parameter γ does not seem to have an impact on the final result. Compared to the original model, there is no increased cluster formation and there is still only one final opinion – complete convergence can be observed over time. This time, however, is longer than in the scenario without the charisma parameter. The reason for this can be found in the underlying formula. Gamma does not replace the convergence parameter, but is an additional factor, by which the opinion adaption is multiplied. Only if the value for γ is 1 for both of the interacting agents the convergence rate will be the same; a scenario which is not very probable. The logical consequence is that the process of convergence slows down.
The result, that γ does not affect any factors besides the speed, could have been foreseen all along when looking at the used approach. The values for γ, just as the number representing the agents' opinion, are assigned randomly using the function rand(). The distribution will be uniform, caused by the nature of this very function. Statistically, any effect γ might have, will be compensated by a counteracting effect. Unfortunately, we did not think of this in advance. However, γ can partly compensate large values for c, thus inhibit minority formation caused by fast convergence (see point 6.3). But since this can be achieved by simply minimising the value for c, it is an interesting, yet useless fact.

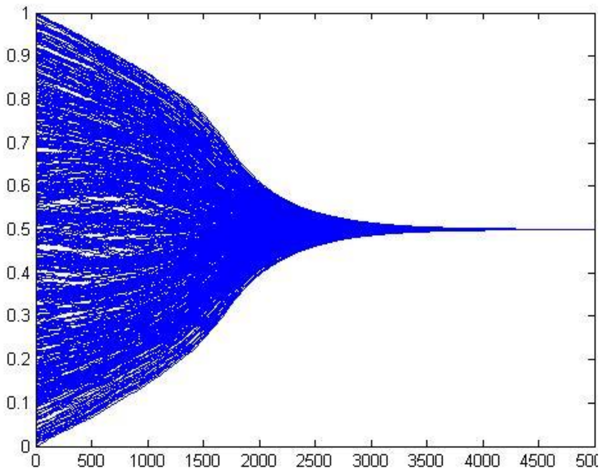

Figure 11: u: 0.35, with c: 0.002; 500 agents

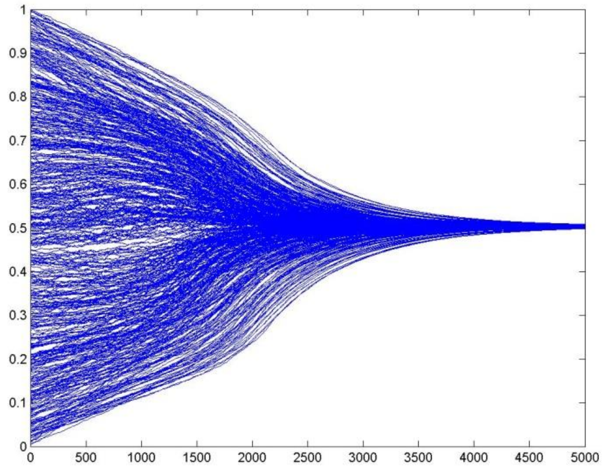

Figure 12: same parameters as in Scheme 11, with alteration

6.6. Influence of circles of trust

The influence of the "circles of trust" depends strongly on the arbitrarily defined values for f1, f2 and f in the code. The general rule is: The higher the values, the faster the convergence. There is a lot of space for testing. The radiuses can be changed as well as the factors, thus having an impact, too. They can also counteract each other, a small radius with a large factor will have a similar impact on convergence speed as a large radius in combination with a smaller factor. "Nonsense combinations" are equally possible, where closeness of the agents has a negative impact on their opinion convergence compared to a further distance. In a real social system, this would be an unlikely result. This happens if

f1 < f2 < f. In the case of f1 = f2 = f = 1 no difference from the original model will result.
Summarily, this addition is not of big use without any numerical data to justify specific values for the factors.

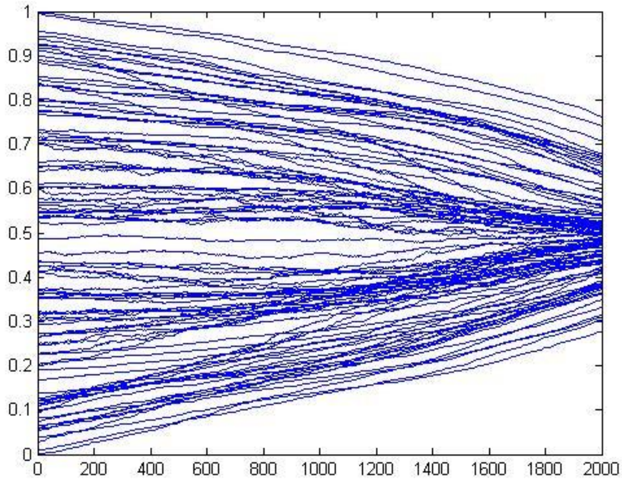

Figure 13: u: 0.35, with c: 0.002; 2000 iterations; 100 agents; f2=3, f1=1, f=0.75

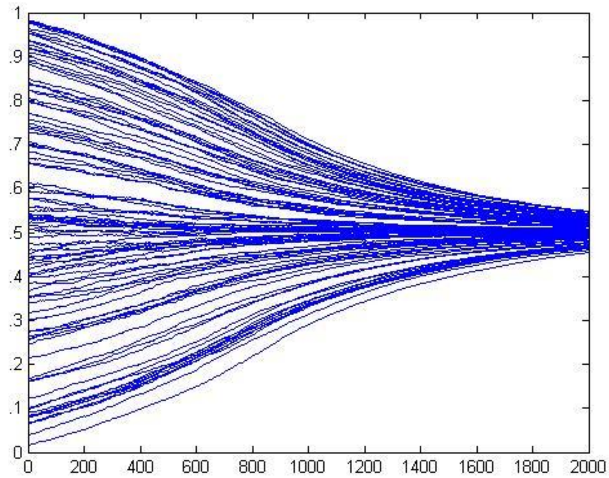

Figure 14: u: 0.35, with c: 0.002; 2000 iterations; 100 agents; without circles of trust

7. Summary and Outlook

The formula given in our reference[1] as well as our additional ideas were succesfully translated into a model and then into code, so that we could reproduce the results with our programme. We came to the same qualitative conclusions about the influence of the threshold u and the convergence parameter c. This was one of our set goals in the beginning and we successfully accomplished that part.
What resulted from our alterations to the first version of the code was somewhat disappointing, but we did find the flaws of the model.
The final source code could none the less be a valuable basis for further modifications of the model. The parameters could easily be connected and made dependent on each other. If there were to be a scientifically proven correlation between charisma and, for instance, the extremeness of an agent's opinion, this could be incorporated into the model. Such a change would probably have a more significant impact on the outcome of the simulations, since there would be a force counteracting the convergence. Certainly, scenarios worth investigating would emerge.
Another – very obvious – addition would be making the interaction probability a function of the distance between two agents. As a living individual, one is a lot more likely to exchange opinions with a friend or relative than within a stranger. It is exactly this familiarity between agents that we already tried to represent by introducing an areal distance between them. However, only the grade of opinion adaption was made dependent on that distance, not the probability of the interaction itself.
Another modification could be to not use a uniform distribution of the initial opinions, but a different approach, e.g. a Gaussian distribution. In politics, it is often observable that moderate positions have the most majority appeal, because more people can relate to them. This would support a different, non-uniform distribution of opinions in the first place.
The same principle could be adopted for the second parameter where `rand()` was used, which was the charisma (γ). This would make more sense with respect to real social systems, as there will be more people with average than with outstandingly high (or low) charisma.
Those are only a few examples, the model could be extended a lot further and then be experimented with.

8. References

[1] M.F.Laguna, Guillermo Abramson, Damiàn H. Zanette: “Minorities in a Model for Opinion Formation”, San Carlos de Bariloche, 2003
[2] Our group´s Github repository: https://github.com/boekel/annel-boekel-mgut