Analysis of Integrity and Segregation Seeking Models with a Product Adoption Application

by
Utku Arık

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fullfillment of
the requirements for the degree of
Master of Science

Sabancı University
July 2018

ANALYSIS OF INTEGRATION AND SEGREGATION MODELS WITH A PRODUCT ADOPTION APPLICATON

APPROVED BY:

Prof. Dr. Ali Rana Atılgan
(Thesis Supervisor) .....

Assoc. Prof. Dr. Güvenç Şahin .....

Asst. Prof. Dr. Ahmet Onur Durahim .....

DATE OF APPROVAL: 31/07/2018

Analysis of Integrity and Segregation Seeking Models with a Product Adoption Application

İbrahim Utku Arık

Master of Science in Industrial Engineering, 2018

Thesis Supervisor: Prof. Dr. Ali Rana Atilgan

Abstract

In this thesis, we study a kind of Ising-Schelling model with different lower and upper bound thresholds limits. The model consists of a regular lattice network with agents on it. Additionally, periodic boundary conditions are hold for the model. An agent is classified as unhappy if the proportion of the total number of neighbors which are the same type as this agent over the total number of neighbors is not greater than or equal to the lower bound of the agent. Then these unhappy agents start to convert themselves to other type. The algorithm works until there is no unhappy agent left in the network. The model is analyzed with different initial configurations, lower and upper bounds and the size of monochromatic clusters is studied. Our simulation results suggest that it is possible to reach ground state by manipulating initial numbers of different agent types and also with assigning different lower bounds for different agent types. The results also say that the size of monochromatic regions is independent of network size but highly depends on the lower bound and neighborhood radius. Furthermore, the models with upper bounds end up with interesting combined lines. Therefore, we named them as maze states and under different configurations, the appearances and distortion of these kind of shapes has been analyzed. Finally, we also implement a marketing application model about new product adoption behavior of customers. Our simulation results suggest that the chance of new product to survive in the community highly depends on the lower bound preferences of the individuals.

Bütünleşme ve Ayrışma Arayan Ağ Modellerinin Analizi: Yeni Ürün Adaptasyon Modeli

İbrahim Utku Arık

Endüstri Mühendisliği Yüksek Lisansı, 2018

Tez Danışmanı: Prof. Dr. Ali Rana Atılgan

Özet

Bu tez çalışmasında, farklı alt ve üst sınır eşik limitleri olan bir çeşit Ising-Schelling modeli üzerinde çalıştık. Model, üzerinde ajanlar bulunan düzenli bir kafes ağından oluşmaktadır. Ek olarak, model için periyodik sınır koşulları da geçerlidir. Bir ajan, eğer bu ajan ile aynı tipte olan komşularının toplam sayısının tüm komuşularına oranı, o ajanın alt sınırından büyük veya ona eşit olmaması durumunda mutsuz olarak sınıflandırılır. Sonra bu mutsuz ajanlar kendilerini diğer türe dönüştürmeye başlar. Algoritma, ağda mutsuz bir ajan kalmayıncaya kadar çalışır. Model farklı başlangıç konfigürasyonları, alt ve üst sınırlarla analiz edildi ve monokromatik kümelerin boyutu incelendi. Simülasyon sonuçlarımız, farklı ajan tiplerinin başlangıç sayılarını manipüle ederek ve farklı ajan tipleri için farklı alt sınırlar atayarak, minimum enerji sevşyesine ulaşmanın mümkün olduğunu göstermektedir. Sonuçlar ayrıca, monokromatik bölgelerin büyüklüğünün ağ boyutundan bağımsız olduğunu, ancak alt sınır ve komşuluk yarıçapına bağlı olduğunu belirtmektedir. Ayrıca, üst sınırları olan modeller ilginç birleşik çizgilerle sonuçlanmakta ve bu nedenle, bunları labirent halleri olarak adlandırdık ve farklı konfigürasyonlar altında, bu tür şekillerin görünüşleri ve çarpıklıkları analiz edildi. Son olarak, müşterilerin yeni ürün adaptasyon davranışları hakkında bir pazarlama uygulaması modeli uyguladık. Simülasyon sonuçlarımız, toplumda yeni bir ürünün hayatta kalma şansının büyük ölçüde bireylerin alt sınır tercihlerine bağlı olduğunu göstermektedir.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Agent based simulations on regular lattice networks have been widely used for research studies in different areas of science. Ising[11] and Schelling[17][16] models are one of the famous examples of the agents based simulations and in this thesis, we mainly focused on a similar model to these models which simply includes neighbor interactions in different size of neighborhoods and thresholds in regular lattice network.

Mathematically similar simulation models which are dealing with neighborhood interactions with certain thresholds or individual preferences of agents can be seen with different names in the litearature, such as Ising models, spin glass models, cellular automata, Hopfield networks, error correcting codes and Schelling model [19] [8] [9][14] [17]. These all models basically consist of some agents on a network and the interactions of these agents between each other. Each agent wants that some function of its predefined neighbors count to be greater than or equal to its predefined threshold. If this happiness condition does not hold, then that agent will need to take some actions regarding to algorithm dynamic such as converting itself to other type, moving to another vacant agent in the network or swapping positions with another unhappy agent and so on.

In 1969, an American economist Thomas Schelling had published a pioneer paper about segragation in cities and created a representative model for it. Despite, Schelling simply made his simulations by just flipping metal coins on a board, later, it has been proven with the help of advanced computer systems that his calculations are correct. He discovered that for certain level of thresholds, the movement of unhappy agents to vacant places creates a cascading affect which cause network to segregate to different monochromatic regions.[16] Moreover, this paper resulted in different studies about agent based simulations in both 1D and 2D networks by various research groups. Despite the huge amount of work about the similar topics,

almost all the results are the only outcomes of the simulation results. Then, early 2010s, the first mathematical proofs have shown up. Brandt et al. [4] analyzed 1D Schelling model. Their study is the first detailed analysis of Schelling model. The main result of this work is when the simulation reaches one of its possible equilibrium states, the average size of monochromatic regions is independent of n which is the length of the ring network and the average size of these regions is polynomial in w which is the size of the neighborhoods. Immorlica et al. [10] have done similar analysis for 2D models and found that the average size of monochromatic regions is exponential in w^2. Further, for the lower bound threshold 1/2, the size of monochromatic region is bounded by $\Theta(w^2)$. Barmpalias et al. analyzed Schelling model with swapping algorithm for unequal threshold levels for two different agent types. [3] In addition to that, Barmpalias et al. [2] showed that for one dimensional Schelling segregation model, when the lower bound marginally decreases from 1/2 to some point, the size of monochromatic regions dramatically changes from bounded in linear w to bounded in exponential w. Young [20] described the stochastically stable states for perturbed segregation process which cause the separation of the network to two different monochromatic regions. Additionally they described the pairwise swapping algorithm on the simulations, and this algorithm called as Kawasaki dynamics and also Young extended the Schelling model by simply adding noise which makes temperature greater than zero ($T > 0$). Upon the works of Young, Zhang [21] furthered this study to 2 dimensional network and again analyzed the stable states that divide network to separated groups. Furthermore, there are different but similar versions of these kind of Ising-Schelling models. For instance, Krapivsky and Redner[13] analyzed the spin interactions on two-state spin systems with majority rule. Then, Chen and Redner[7] analyzed the this majority rule dynamic on finite dimension networks. By this dynamic, contrary to previous Ising - Schelling dynamics, not a single agent but the all the neighborhood converts themselves to the type of the majority class.

If we define neighborhood radius of agent i as w_i and the spin of this agent as s_i then hamiltonian equation becomes:

Hamiltonian:

$$\mathcal{H} = -\sum_{j \epsilon w_i} J s_i s_j \tag{1.1}$$

where J is the strength of the bound between spins i and j.

The main purpose of the dynamic algorithms for those different models is the minimize this energy or in other words hamiltonian function, reaching the global minima in the energy landscape. Since the algorithms that are used are mainly heuristic algorithms, in some cases, the simulations might stuck at local minimas and cannot reach to ground state. In the literature, the term ground state can be used for the minimum energy level which is possible to reach during the simulation. However, in this thesis, we use this term when the network only consists of one type of agents. In other words, the term is only used for global minima energy level not for local minimas.

Chapter 2

Model

The model is 2 dimensional Ising-Schelling model with Glauber dynamics with zero temperature. The network is torus and it consists of n x n agents with periodic boundary conditions. Unlike classical Ising models with 4 neighbors we considered $(2w+1)^2-1$ neighbors within Moore neighborhood and for each agent, that w is the neighborhood radius of a agent and each agent is Bernoulli distributed as a green or a red agent with probability 1/2. Depending on its type(color), each agent has lower bound and upper bound which are represented as $\tau^l_{g(r)}$ and $\tau^u_{g(r)}$ respectively. These bounds represent acceptable limits for neighbor type proportions. agents prefer same type neighbor proportions greater than or equal to their lower bound and less than or equal to their upper bound. Zero temperature means that agents can not behave against their will and preferences, in other words type changing(flipping) decision of an agent does not depend on a probability if the agent is already chosen for flipping process. In addition to these features, periodic boundary conditions provide connection between the agents which are located at the boundaries of the network. This condition represented in the below figure.

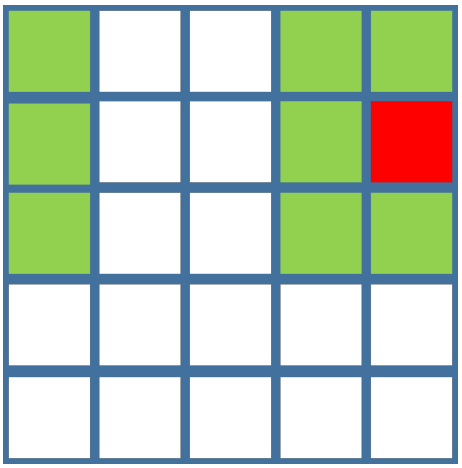


Figure 2.1: Representation of periodic boundary conditions, the green agents are the neighbors of the red agent. Therefore, the agents that are at the boundaries of the network connected to each other. If you move to the left and leave the network you enter again from the right side. Again in a same way, going to upward ends up with reaching the bottom of the network.

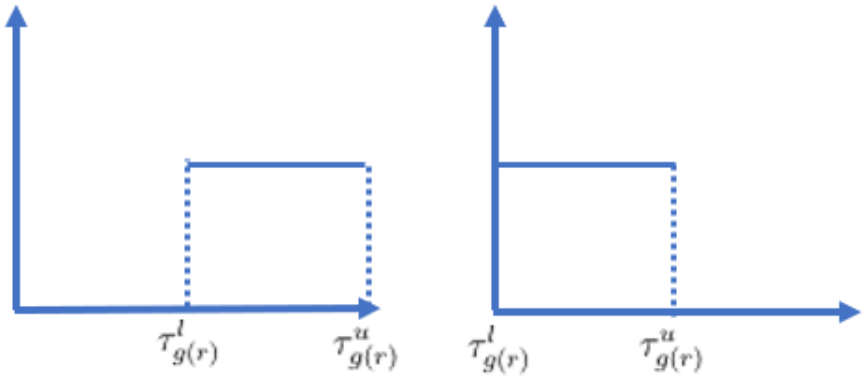


Figure 2.2: Representation of $\tau^l_{g(r)}$ and $\tau^u_{g(r)}$. The first graph shows the lower bound $\tau^l_{g(r)} = 1/2$ and the upper bound $\tau^u_{g(r)} = 1$ case and the second graph shows $\tau^l_{g(r)} = 0$ and $\tau^u_{g(r)} = 1/2$ case. In other words, in the first figure the agents prefer a least half of the its neighbors from its type, and in the second case, the agents prefer at most half of the its neighbors from its own type.

Chapter 3

Dynamics

The process dynamic in the model that has been investigated in this thesis is executed as follows; at each time step t the algorithm creates a list of unhappy agents. Then let us define Ω specifically for each agent as the proportion of the total number of agents with the same type neighbors over the total number of neighbors in the neighborhood with radius w. We will define then a agent is unhappy if one of these two inequalities holds:

$$number\ of\ same\ type\ neighbors = \Omega > \tau^{u}_{g(r)}((2w+1)^2 - 1) \tag{3.1}$$

or

$$number\ of\ same\ type\ neighbors = \Omega < \tau^{l}_{g(r)}((2w+1)^2 - 1) \tag{3.2}$$

This means in words that an agent is identified as unhappy if the proportion of the total number of neighbors which are the same type as this agent over the total number of neighbors is not greater than or equal to the lower bound of the agent, and at the same time is not less than or equal to the upper bound of the agent. In a result of this case, this agent will be categorized as unhappy. After the list is completely created which means all of the unhappy agents in the network have been added to this list, a agent is chosen uniformly at random one by one in that time step and if this chosen agent is unhappy when it is picked, the agent will flip its color. One may ask the reason behind checking the agent if it is unhappy despite it is already in the unhappy agent list. The logic of this, since there might be chosen agents for flipping process before this agent, the changes of the previous agents could make current agent already happy. In such a case, if we do not check the current agent we would make a happy agent to turn unhappy. Therefore, this control process is necessary for the algorithm. If the chosen agent is not unhappy then the algorithm picks another agent in the same fashion. At each choosing attempt no matter whether the agent changes its color or not, the relevant agent will be eliminated from the unhappy list. When there are no elements in the list, the algorithm moves to the next time step

and creates a new unhappy list. The general process terminates when there is no unhappy agent in the network.

Moreover, if we convert the calculations of Barmpalias et al.[2] for 1d schelling segregation model to 2d, the percentage of the unhappy agents in the initial configuration can be calculated as follows;

First, if we assign "η" to total number neighbors of an agent which is $(2w+1)^2-1$. Then, the unhappy agent percentage for $\tau^l_{g(r)}$ and $\tau^u_{g(r)}$ will be:

$$Unhappy\ percentage = (\sum_{i=0}^{\lfloor \tau^l \eta \rfloor} \binom{\eta}{i} + \sum_{i=\lceil \tau^u \eta \rceil}^{\eta} \binom{\eta}{i}) \frac{1}{2^\eta}\ 100 \tag{3.3}$$

Models which are similar to our model have been widely discussed in the literature. For instance, Barmpalias[1] described their model process as follows; at each round all the agents in the network are checked and a list of unhappy agents is created. An unhappy agent is then picked for spin flip from the list with the condition that it is still unhappy when it is being picked up. In Immorlica's model[10], each agent has a Poisson alarm with lambda 1 and when a agent's alarm rings, if this agent is unhappy its type is flipped. The process continues until there are no unhappy agents remain. Finally, Hamed's algorithm[15] works as follow; at each time step t all the agents are checked and the agents are named as unhappy whose neighbor type proportion is less than its lower bound. at that time step t an unhappy agent is chosen uniformly at random and its type is switched.

3.1 Alternative Dynamic models and algorithms

3.1.1 Glauber dynamics

In Glauber dynamic, at each iteration step, if the chosen agent is unhappy this agent switches its type to other type. Barmpalias et al.[1] describes and adapts this algorithm logic to social segragation model as follows; if an agent is unhappy it will immediately leave the network, then from outside the network another agent from opposite type will arrive and locate this vacant place.

3.1.2 Kawasaki dynamics

In Kawasaki dynamic, at each iteration step two unhappy agents are chosen from both two different color types. After that, these two agents swap their positions.

Therefore by executing these dynamic the number of agents from each color type will be same in whole simulation time.

3.1.3 Asynchronous dynamics

In this dynamic model, at each iteration step, the first action is the creation of the unhappy agent list. After that, an agent is chosen uniformly at random from the list and is checked if the agent is still unhappy or not. Next the chosen agent which is unhappy flips its color to the other color type. It is obvious that the first agent which is chosen immediately after the creation of the unhappy agent list is definitely unhappy. Therefore, unhappiness check for the first agent is not crucial for the dynamic. However, after the first agent, we need to check unhappiness since the flipping process of previous agents could make that agent already happy if the previous chosen agents were in the neighborhood of that agent. After the flipping action of the agent is done, again in a same fashion the algorithm chooses another agent uniformly at random and implies the same step until there is no unhappy agent left in the list for this iteration step. Then, again algorithm creates a new unhappy agent list and repeats the same process untill there is no unhappy agent left in the network. The essential point in this dynamic is the algorithm chooses the agents one by one form the unhappy list.

3.1.4 Synchronous dynamics

In synchronous dynamic model again in a same way as in asynchronous model, at each iteration step the algorithm creates the unhappy agent list. After that, different from the asynchronous version, the algorithm converts all of the agents in the list to their opposite types at the same time. Then the next and new unhappy agent list is created at the next iteration step and algorithm continues as in a same fashion until all of the agents in the network are content with their locations .

3.2 Differences between classical Ising model and our model

A Zero temperature Ising model with a Von Neumann neighborhood (4 neighbors) allows both energy lowering and energy conserving spin flips. However, energy conserving spin flips are not allowed in our Ising-Schelling model but only energy lowering spin flips are allowed.

Energy conserving flips occur when an agent has an equal number of neighbors from

both types in its defined neighborhood . In a classical Ising model energy conserving flips have 1/2 probability to occur. Therefore, a 1/2 neighbor type fraction is not a sufficient configuration for an agent in the equilibrium state of the network. According to Spirin et al.[18] a classical Ising model ends up with ground state with high probability which is close to 2/3 or a stripe state with probability close to 1/3.[18] Since energy conserving flips are not allowed in our model, neighbor type fraction is a sufficient state for an agent in the equilibrium. Therefore, we can observe segregated clusters in the network at equilibrium state.

To prove why energy conserving flips cause stripe boundaries between two color clusters; first we need to consider about the equilibrium state of the system and take a 3 x 3 cross section from the boundary where two monochromatic clusters collide. Theoretically, there are 3 possibilities for boundary in this cross section and let us assume that these 3 cases are possible configurations for the equilibrium. With a Von Neumann neighborhood model, for the first case, the red square in the middle column and the green square at the center of the 3 x 3 square are both unhappy. In other words the agents want to flip with 1/2 probability. In the second case, the red square at the center is unhappy, so it wants to flip and also the green squares at the middle column flip by probability . No matter which color starts first to switch, the boundary ends up with stripe state. Because, the algorithm will not stop if there is a agent which has equal proportion for different neighbor types. Since the third case is symmetric with the first one, the logic is identical for it. Therefore, if there are still two colors in the network system at equilibrium, the cluster boundaries have to be stripe.

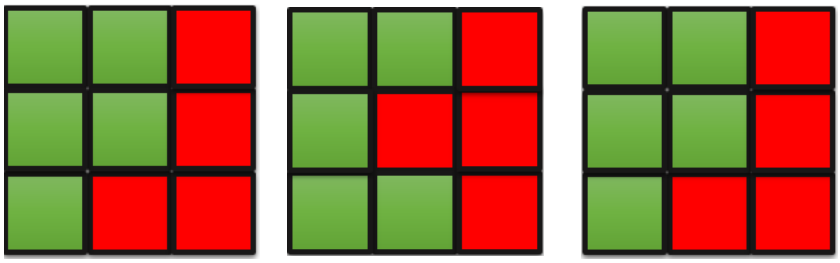


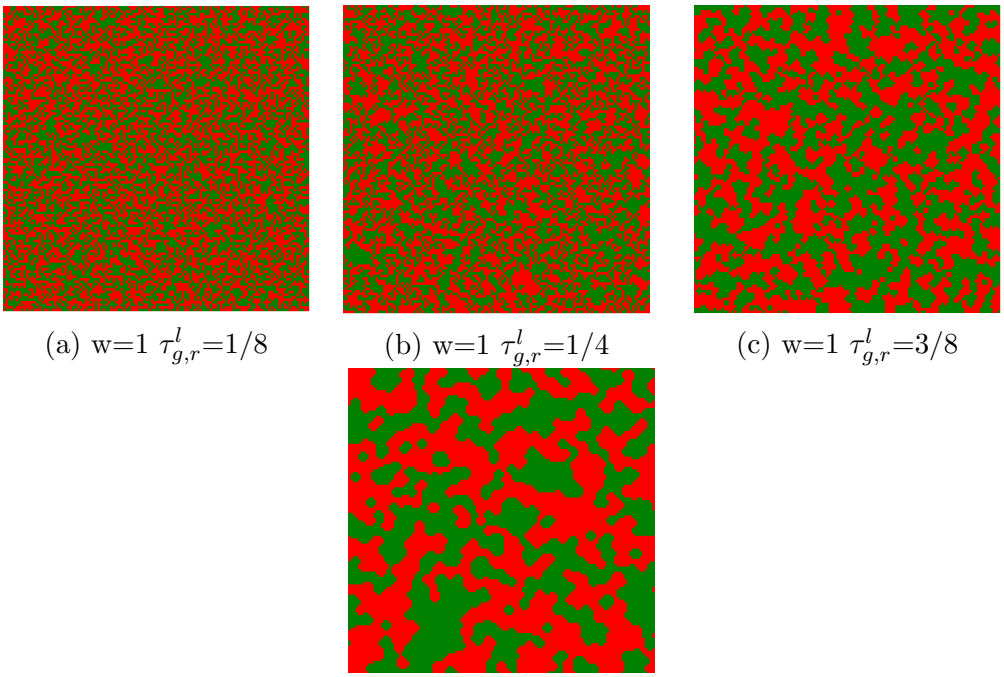
Figure 3.1: The figure depicts the possible cases for cluster boundaries for 3x3 network. On the left the red agent at the center bottom is unhappy. Therefore, it will want to change its type to green. In a same way at the center figure the green agent in the middle column and at the right figure the red agent in the upper side of the middle column is unhappy and they will change their colors.

Chapter 4

Segregation Seeking Model With Lower Bound Thresholds

In this type of cases, there is no upper bound limit for proportion of neighbors for an agent but the lower bound limit for this proportion exists. In other words, an agent can be completely surrounded by neighbors which are same type with that agent but it cannot be completely surrounded by opposite type of agents. An agent demands at least lower bound proportion of its neighbors from its own type.

Our simulation results show that, as can intiutively expected, when the lower bound threshold $\tau^l_{g,r}$ increases for both of the two types, the size of the monochromatic clusters increase and the same color of clusters are getting departed from each other. The intuitive reason for that is the agents desire higher value for same type fraction in their neighborhood so it cause expanded monochromatic clusters. Our results also show that when the neighborhood radius w increases the size of the monochromatic regions also increase. Immorlica et al.[10] proves that for $\tau^l_{g,r}$ close to 1/2 (in their model $\tau^l_{g,r}$ is defined as same type fraction in neighborhood, different from our model, the type of the centered agent in the neighborhood is also included) the expected size of monochromatic region is $e^{\Theta((\frac{1}{2}-\tau)^2 w^2}$ where w is neighborhood radius.



(a) w=1 $\tau^l_{g,r}$=1/8 (b) w=1 $\tau^l_{g,r}$=1/4 (c) w=1 $\tau^l_{g,r}$=3/8

(d) w=1 $\tau^l_{g,r}$=1/2

Figure 4.1: Equilibrium snapshots of $\tau^l_g = \tau^l_r = 1/8$, 1/4, 3/8 and 1/2 cases respectively when color types are Bernoulli distributed for each agent with probability 0.5. One can observe that the most left figure is similar to initial configuration of the network, since the order acquired from the algorithm is smaller than the other setups due to relatively small lower bound, the equilibrium state looks like distributed as almost uniformly at random.

In Figure 4.1, as mentioned before, we can observe that when the lower bound of the agents increases and if we keep other setups fixed, the distance between the different monochromatic clusters increase. Additionally, when the lower bound is equal to 1/8 or 1/4 and the neighborhood radius w is equal to 1, the equilibrium state seems like it consists of uniformly at random distributed agents. The reason behind this, for values less than the 1/4 lower bound, after the distribution of agents, there are only small amount of unhappy agents exist in the network.[1] Therefore, the number of converted agents are considerably small and the network is similar to its initial configuration.

4.1 Changing initial ratios

When the two type of agents distributed uniformly into the network and the lower bound threshold is equal to 1/2, we do not observe any ground states for 100 simulations. Therefore, we decided to manipulate initial fractions of different types in the network to understand the behavior of the dynamics and to find if there is any ground states for different initial configurations. Thus, instead of assigning green

or red to each vacant node with probability 1/2, we first determine initial numbers of agent types and then for each agent, we assign it to random vacant place in the network.

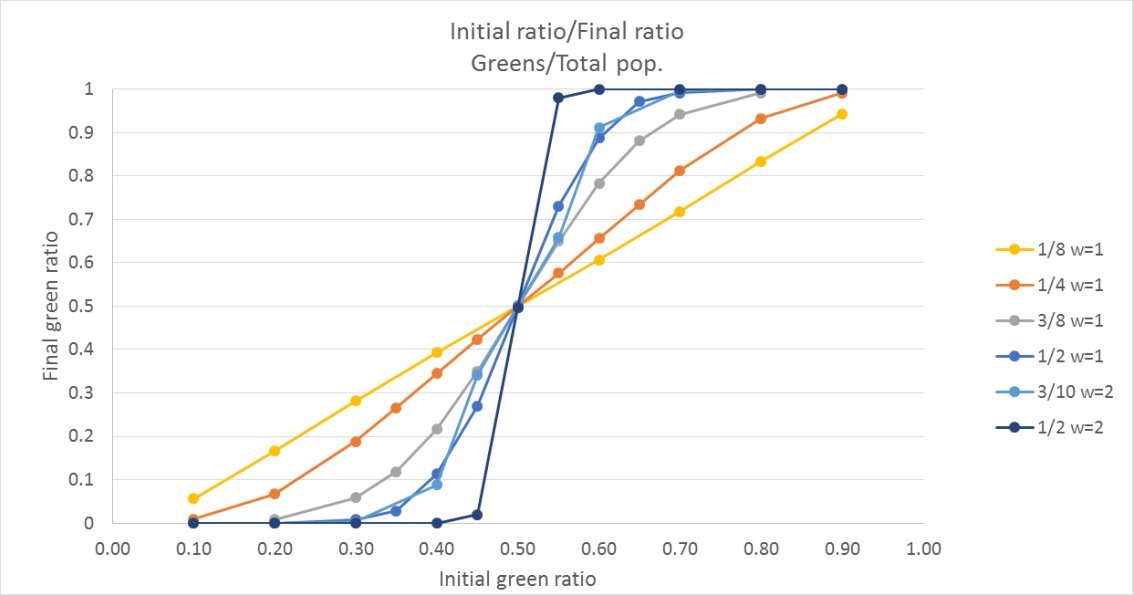


Figure 4.2: Final green ratio versus initial green ratio respect to 1/8, 1/4, 3/8 and 1/2 lower bounds for w=1 and 3/10, 1/2 lower bounds for w=2. When the neighborhood radius increase the slope of the figure increases. In other words, when w increases and the lower bound is fixed, the more crowded color type dominates the network more than the previous w.

Then we extend this idea to not only to observe ground states but also to examine how one color's initial ratio affects its final ratio in the system when two colors have equal lower bounds such as $(\tau_g^l = \tau_r^l)$. According to our simulation results, in Figure 4.2, we can observe that when $\tau_{g,r}^l$ decreases the graph loses its curvature and in fact when $\tau_g^l = \tau_r^l <= 1/4$ the graph becomes almost linear. Therefore, the one can easily inspect that when this inequality holds$(\tau_g^l = \tau_r^l <= 1/4)$ initial and final ratios are almost identical. According to Barmpalias et al.[1] for $\tau > 3/4$ and $\tau < 1/4$ the regular lattice network is static almost everywhere, so we can claim that the initial ratio of agents will fluctuate in a small scale during flipping process but it will remain almost the same at the end of process. We observe that for any $\tau_g^l = \tau_r^l \leq 1/2$ and $w \geq 1$, %50 initial ratio for each color is conserved as simulation average.

Moreover, when initial ratio is larger than %50 for one type and the lower bounds are greater than or equal to 1/4, even an insignificant increase of starting ratio for one type causes an increase for the final ratio and the amount of the growth depends

on the both lower bounds of the agents and the neighborhood radius. That rise in final ratio growths faster when we both increase neighborhood radius(w) and the lower bounds which provoke the graph to become more sharper. In same fashion, if the initial ratio is less than %50 for one agent type then we can assure that final ratio for that agent group will be less than its initial ratio. Lastly, for the initial ratio of more than %60 and w=2 the all of the simulations end up with ground states.

If we want to adapt this result to Schelling's segregation model,we can remark that if a minority group has same desire about their neighborhoods as the majority group, it will be really hard for them to survive in that neighborhood and some amount of people will leave from that society.

4.2 Changing lower bounds(τ^l)

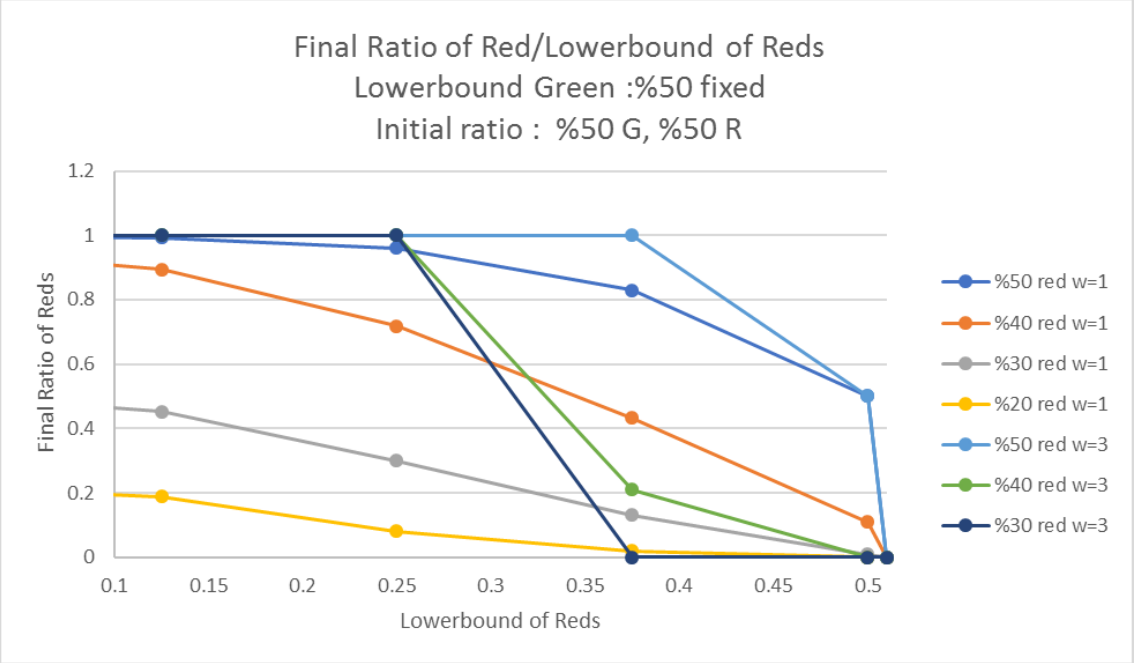


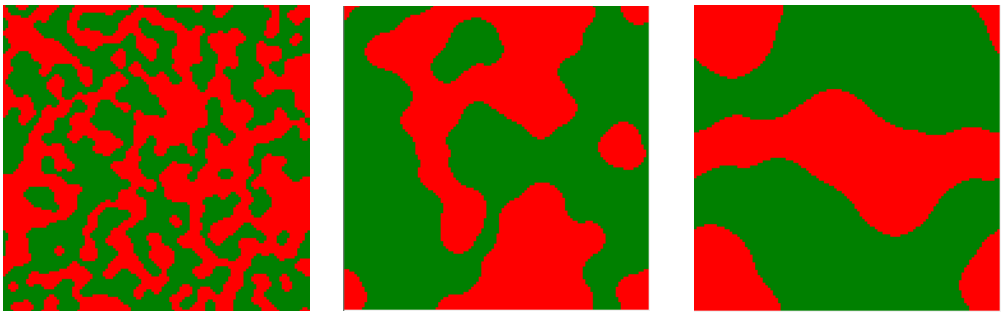
Figure 4.3: Final ratio versus lower bound of agent group if we keep other group's lower bound at %50 and set their inital ratio at %50. Even if one color type starts the simulation with considerably small amount of agents, they can survive in the network in the equilibrium by setting their lower bounds small as possible. Moreover, for some cases there can be a domination of minority groups at the end, for instance, for %40 starting ratio for the reds, when the reds set their lower bounds to 0.125 they almost reach to %85 proportion in the network.

In this analysis, we both manipulate the initial ratio of one type and the also the lower bound of the same type. In Figure 3, the lower bound of the green agents

is fixed at 1/2 and for the different initial ratios of the red agents we analyzed the final proportion of the agent types by changing the lower bound of the red agents. The main motivation behind this logic is that "Can the agent type dominate the system (invasion of all agents) by keeping its lower bound sufficiently lower even if its initial population size is less than other agent type's initial population size?". In a same way one can also be curious about required amount of agents from one type to survive in the network. To examine this question we decided to simulate different scenarios.

First we have started our simulations with %50 - %50 starting proportion for both of the color types and we set neighborhood radius w equals to 1. In this case, as we expected before, the initial and the final proportions are the same and equal to %50. However, when we start to decrease lower bound of the red agents and run the simulation again, we observe that red agents start to dominate the network. For instance, for the %50-%50 setup when we set lower bound of red agents as 3/8, the final proportion of red agents becomes greater than %80 and around %81. In addition to that, an increase in neighborhood radius causes more growth for previous setup. Again in previous setup but now when we set w = 3 and lower bound to 1/2 for red agents, the simulations end up with %50 red proportion. However distinct from the w = 1 case, when we set lower bound of red agents as 3/8, the final ratio of reds become almost %100 and it means that they completely dominate the network.

Then these results remind us the same question;" Can we dominate the system even if our population is remarkably less than other group?" to answer this question we also set different configurations such as the situations when amount of red agents less than the green ones. For %40 red configuration when w = 1 shows us that if the minority group reds want same threshold as greens which is 1/2 lower bound this set up make reds to loose almost half their population size. However when the reds decrease their threshold to 1/4, then they will now become the new majority group in the network. Also as can expected, since increase in w has additional effect on the growth, for w = 3 it will be sufficient set up for the reds to become majority when they decrease their threshold to 3/8. Moreover, when we think about the extreme cases such as when the initial red population is really smaller then the greens like when there is 1:4 proportion and when w = 3, even if the reds decrease their lower bounds to 1/8 they are loosing some population and final proportion end up around %19 for reds. For the clarity we have excluded %0 lower bound case since for this case it is obvious that none of the red agents will switch its color so final proportion will be higher than the initial ratio for the reds.



(a) 120x120 w=1 (b) 120x120 w=2 (c) 120x120 w=3

Figure 4.4: Equilibrium snapshots of $\tau_g^l = \tau_r^l = 1/2$ case with different w values when color types are Bernoulli distributed for each agent with probability 0.5. When the neighborhood radius increases the one colored clusters gets thicker and departed from the each other.

Figure 4.4 depicts the change of equilibrium structures regarding to increase in neighborhood radius. As it can be seen from the figure, when neighborhood radius(w) increases, different color clusters are starting to depart away from each other.

4.3 Square analysis method

We have created a humble method to describe and differentiate the results of different network configurations. The main idea is to count the number of color types in all neighborhoods, and works as follows; from very small scale such as 2 by 2, we simply shifting this square in the network one by one and count the neighborhood fractions. After counting of the all network is finished, we enlarge the size of the measurement square to the next divisor of the network length n. Finally until the some size we finish the process and with the collected data we simply draw the probability histograms to make comparison of different configurations. Figure 6 describes how our method simply works; in this case the network size is 3 x 3 and the size of measurement square is 2 x 2.

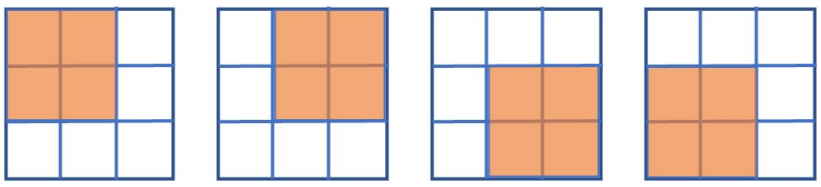


Figure 4.5: Representation of shifting process in square analysis method with 2x2 squares for 3x3 network; by shifting the squares we calculate the inside color ratios of these squares to get probability distribution.

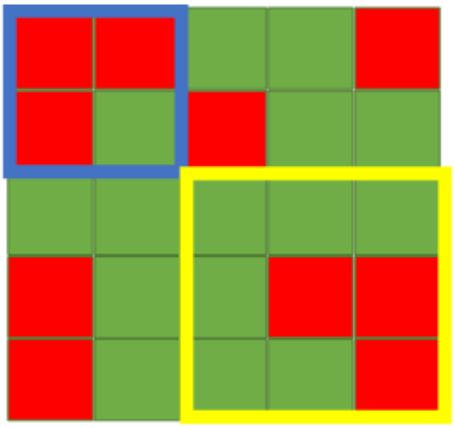


Figure 4.6: In the above figure, if we count the red agents, the blue square is 2x2 and its proportion is 3/4. In a same way, for the yellow square which is 3x3 the proportion is 3/9. We shift these blue and yellow squares in the all network and finally get the probability distribution of proportions.

4.4 Probability distribution of square analysis method

As one can expect when the size of squares in square analysis method increase, the proportion of green and red agents are gathered around 0.5. Additionally, when the square size increases the probability of seeing fully segregated squares closes to 0. Moreover, again intuitively, the limit probability of seeing fully segregated squares goes to 0 when the square size increases therefore we decided to draw the probability distribution graph of monochromatic regions versus square sizes to show that .

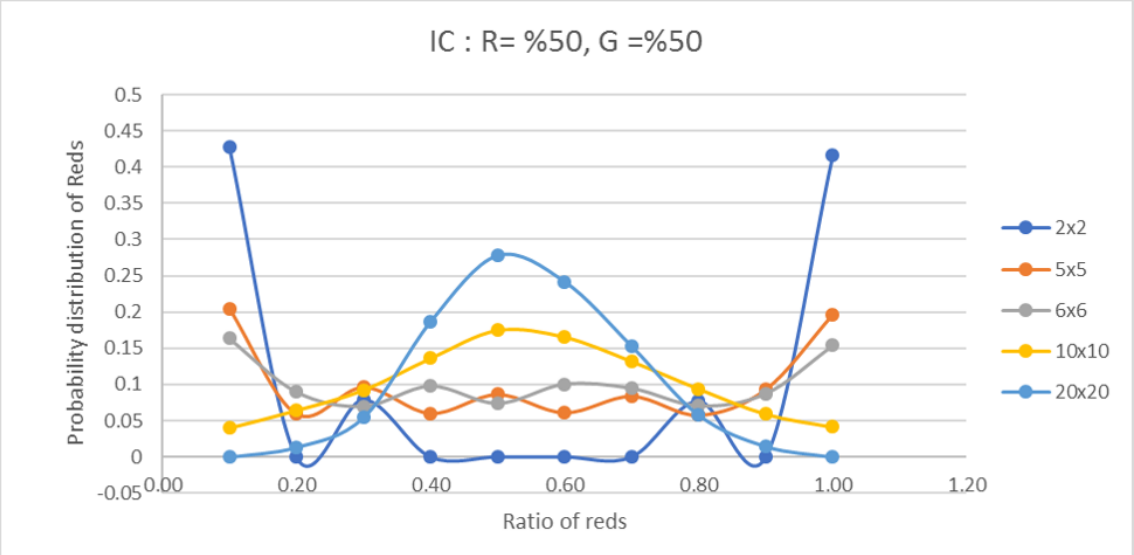


Figure 4.7: Probability distribution of squares with different sizes for w=1 and $\tau_g^l = \tau_r^l = 1/2$. In small scales, the probability of observing fully monochromatic squares is higher than the larger scales. For 20x20 sized squares we do not observe any monochromatic squares.

Even for w = 1, we can draw out from the distribution graph, when our observation square size is small as 2 x 2 we observe monochromatic squares with high frequency and it is marginally lower than %90. When we increase the size of observation squares the distribution graph starts to become flat and finally becomes normal-like shape with very low probabilities for monochromatic squares.

Then, with the data collected from the square analysis method we decide to analyze the probability distribution of monochromatic regions for different size of squares.

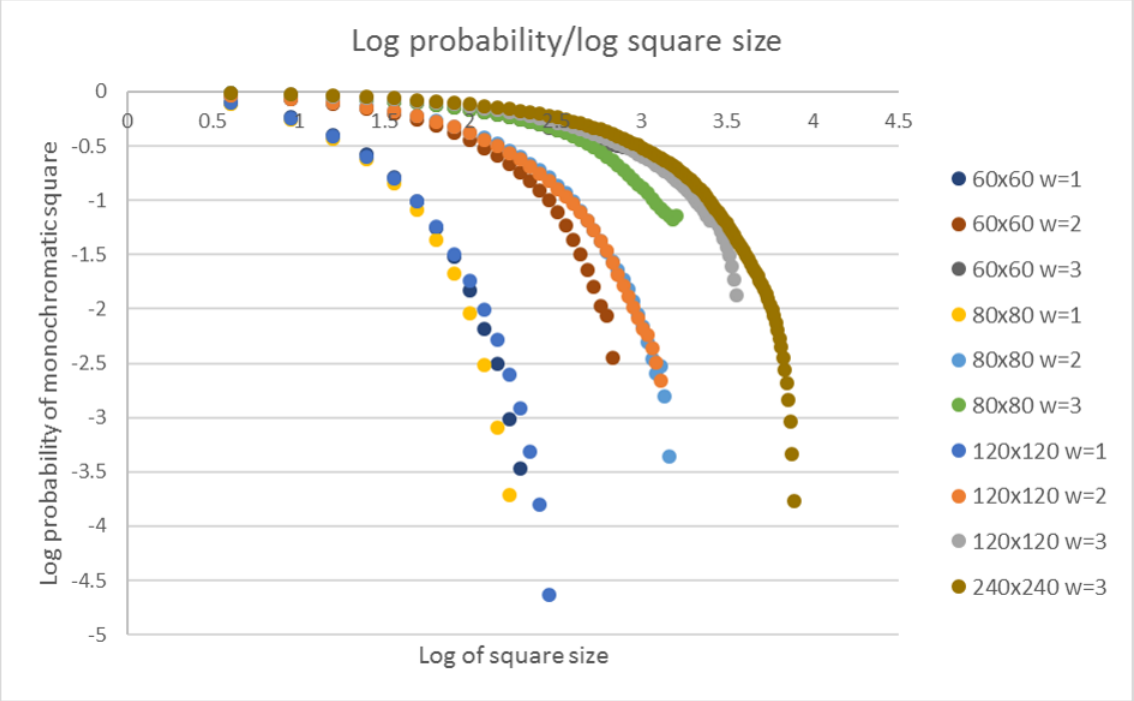


Figure 4.8: The above figure represents square size versus probability of seeing fully segregated(monochromatic) region, even if for larger squares, there is a some probability to see a monochromatic square. In an addition to that, the probability of observing monochromatic squares not depend on network size but depends on neighborhood radius.

Main result in here is probability of monochromatic square size is independent from the network size but this probability depends on the radius of the neighborhood. As can be seen from the figure 4.8, when the neighborhood radius is fixed, simulations with different network sizes almost overlap over each other. However, when the network size is fixed and if the neighborhood radius is different, this time the graphs do not overlap and getting departed from each other.

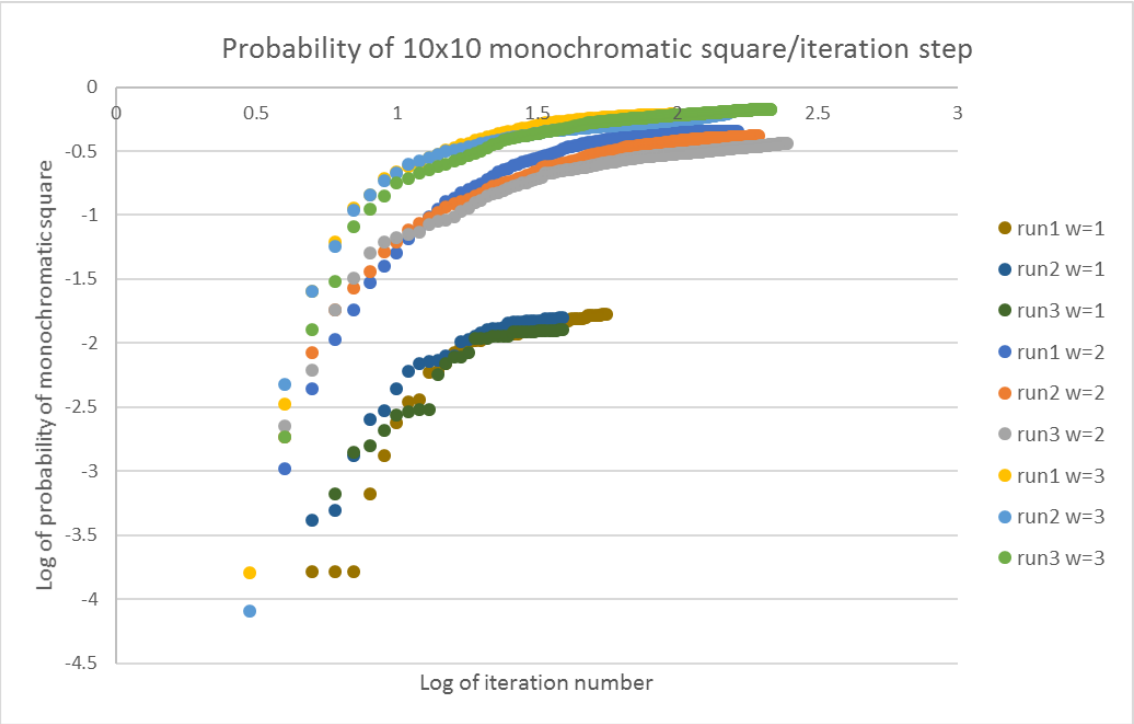


Figure 4.9: The figure depicts log-log curve of probability of monochromatic 10x10 square vs. iteration step. According to that, when the neighborhood radius is higher, the chance of seeing monochromatic clusters increases.

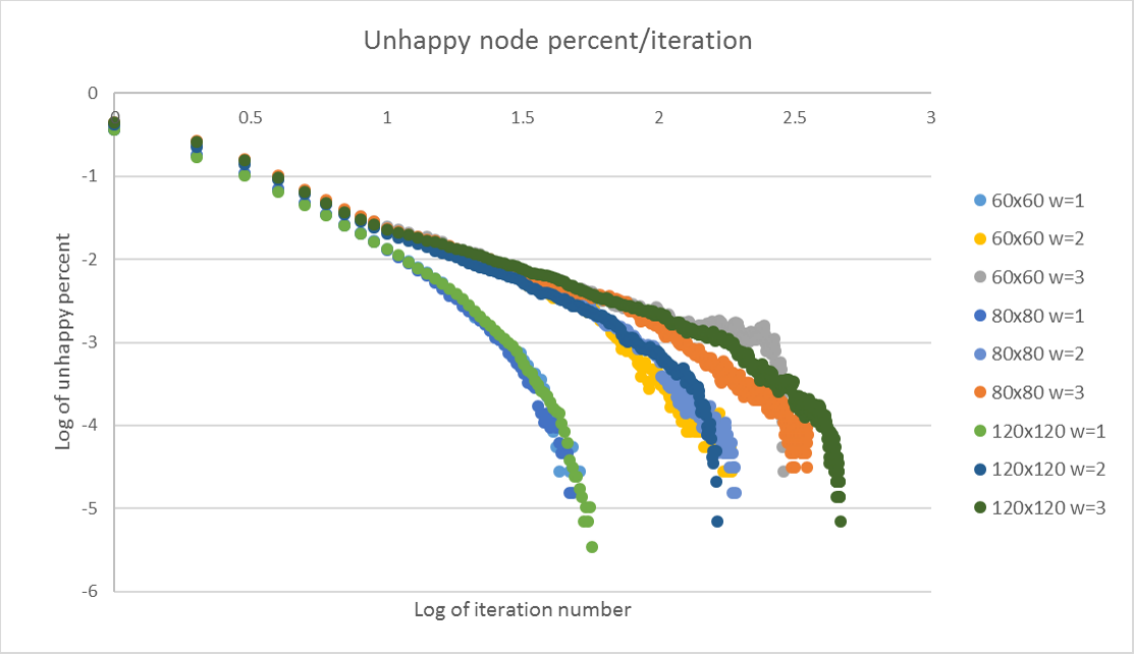


Figure 4.10: The figure of log-log curve of unhappy agent percent vs. iteration step. The networks with same neighborhood radius but different network sizes overlap over each other. Network size do not considerably affect the time for reaching the equilibrium state.

Figure 4.10 demonstrates the log percentage change in unhappy agents respect to log of iteration step; as it is mentioned before, the iteration step in our algorithm is the time between the creation of two different unhappy agents set. The main point in here to extract is again the evolution of coarsening process does not depend on the network size but neighborhood radius. For instance, the simulation averages of 60, 80 and 120 sized networks with w = 1 follow the similar graph pattern and this is also valid for other w values. When the neighborhood radius increase the algorithm spends much more time to converge its equilibrium state.

Chapter 5

Integrity Seeking Model With Upper Bound Thresholds

In this part we define a new configuration and the main difference from the previous model is the existence of upper bound for τ^u which means the agents now desire some amount of agents in their neighborhoods from opposite side. In other words, agents are now going to be unhappy even if they are in a fully segregated region since the neighborhood fraction threshold has an upper bound. Thus, this setup difference causes huge differences respect to previous model simulation model. For instance, if we start with neighborhood radius $w = 1$ and set $\tau^l_{g,r} = 0$ and $\tau^u_{g,r} = 1/2$ then it means that the agents are going to be happy if they have at most 4 neighbors from their own side. Distinct from the previous model, since it was the segregation seeking model, the new version of the model now integrity seeking model.

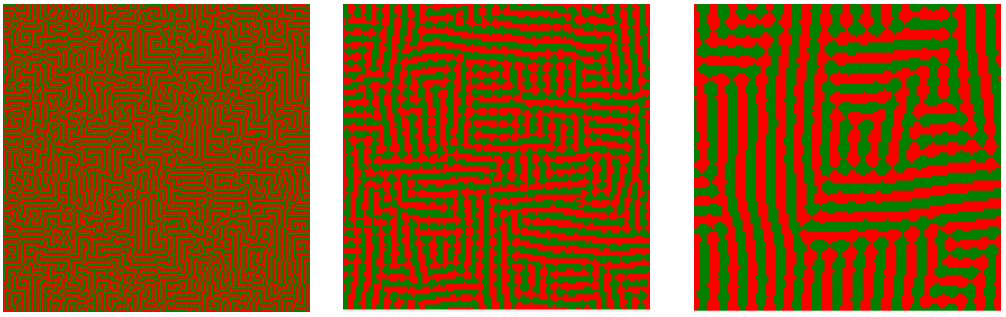
In our study, we have combined different lower bounds and upper bounds and then decided to focus on mainly $\tau^l_{g,r} = 0, 1/4$ and $\tau^u_{g,r} = 1/2$ cases. The first reason behind this decision is when $w = 1$ we had needed a proportional fraction of $1/8$ to fit different setups. For some "w"s especially when they are relatively small respect to network size, the upper bound value needs to implement different setup, in other words to give opportunity to agent to have one more agent from its own side and still become happy we need upper bound to increase as at least:

$$\textit{Addition to upper bound} = 1/((2w+1)^2 - 1) \tag{5.1}$$

For that reason marginally higher upper bound than $1/2$ does not imply any change to the system when smaller w setup configurations.

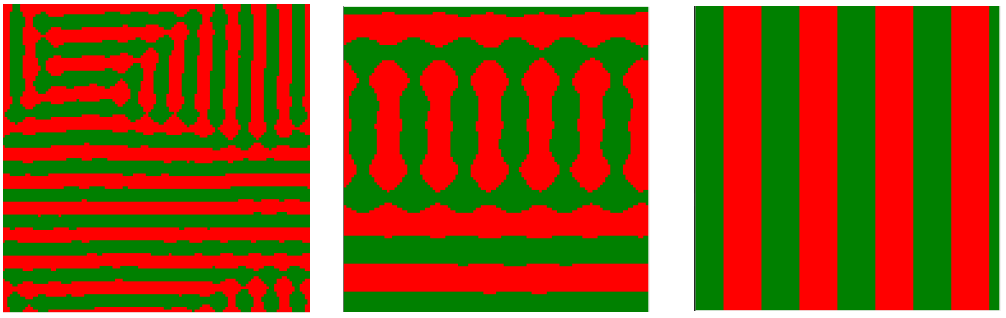
In extended analysis , we observed that for each w ≥ 1, ($0 \leq \tau^l \leq 1/4$, $\tau^u = 1/2$) is special case for different w values and it is depicted in (Figure 5.1). We named these special cases as "maze cases" since the figures look like a maze. In the literature

there are also different cellular automata configurations to reach these type of states [6]. However, in our case we just manipulate the upper bound thresholds for it. If we fix all parameters and start to increase the w the one can observe that the edges are getting thick. When neighborhood radius (w) is sufficiently large so that w/N is not close to 0, the system goes to stripe state.(Figure 5.2, part c). Some w values which are not sufficiently large respect to network size but sufficiently large to distort maze structure constitute stripe like patterns yet, these patterns are not perfectly and literally stripes. As a result of our simulations, perfect stripes occurs for relatively large w values which has the proportion of around $w/N \sim 1/6$. These stripes states can both occur as horizontal or vertical shapes and even in some cases they occurs both in same simulation.



(a) 120x120 w=1 (b) 120x120 w=3 (c) 120x120 w=5

Figure 5.1: Snapshot of $\tau_g^l = \tau_r^l = 1/4$ and $\tau_g^u = \tau_r^u = 1/2$ network for w = 1,3 and 5. Maze structure emerges for all of the three configurations. When the neighborhood radius increases the walls of the maze get thicker and we start to observe longer walls in the network.



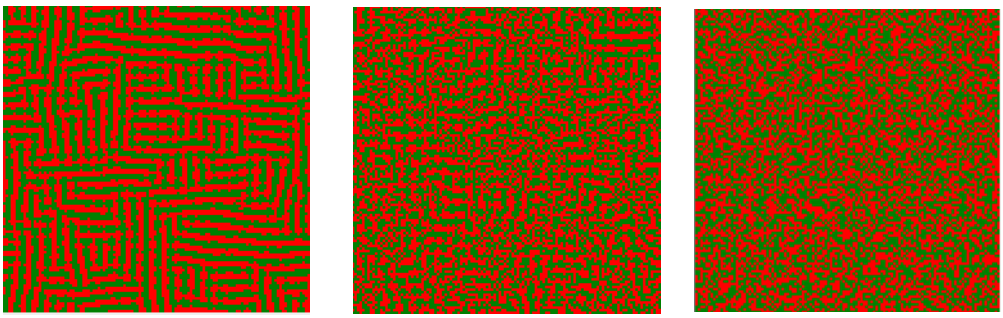
(a) 120x120 w=7 (b) 120x120 w=15 (c) 120x120 w=20

Figure 5.2: Snapshots of $\tau_g^l = \tau_r^l = 1/4$ and $\tau_g^u = \tau_r^u = 1/2$ network for w = 7,15 and 20. On the left we can observe parallel lines different from maze structure, in the middle the maze shape almost completely disappear and parallel and vertical lines emerge and in the right side the network ends up with completely stripe states.

Our simulations show that the system may reach the equilibrium state for $N \gg w \geq 1$ value when we fix $\tau_g^l = \tau_r^l$ at [0, 1/4] and shift $\tau_g^u = \tau_r^u$ in almost any point

between 1/2 and 1. If w is sufficiently large respect to N but N is still larger than w, any point between 1/2 and 1 for τ^u ends up with equilibrium state.

Moreover when τ^u is greater than or equal to $0.5+1/((2w+1)^2-1)$ one can observe that maze shape starts to distort slowly. If we keep τ^u greater than $0.5+1/((2w+1)^2-1)$ and increase the neighborhood radius, this case also results in a distortion.



(a) 25-50 w=3 (b) 25-55 w=3 (c) 25-55 w=6

Figure 5.3: Snapshots of $\tau_g^l = \tau_r^l = 1/4$ and $\tau_g^u = \tau_r^u = 1/2, 11/20$. The figure represents the distortion of maze structure if we relax the upper bound. On the left when $\tau_g^l = \tau_r^l = 1/4$ and $\tau_g^u = \tau_r^u = 1/2$, there is a full maze complex. However, when the upper bound starts to increase the walls of the maze start to loose their shapes.

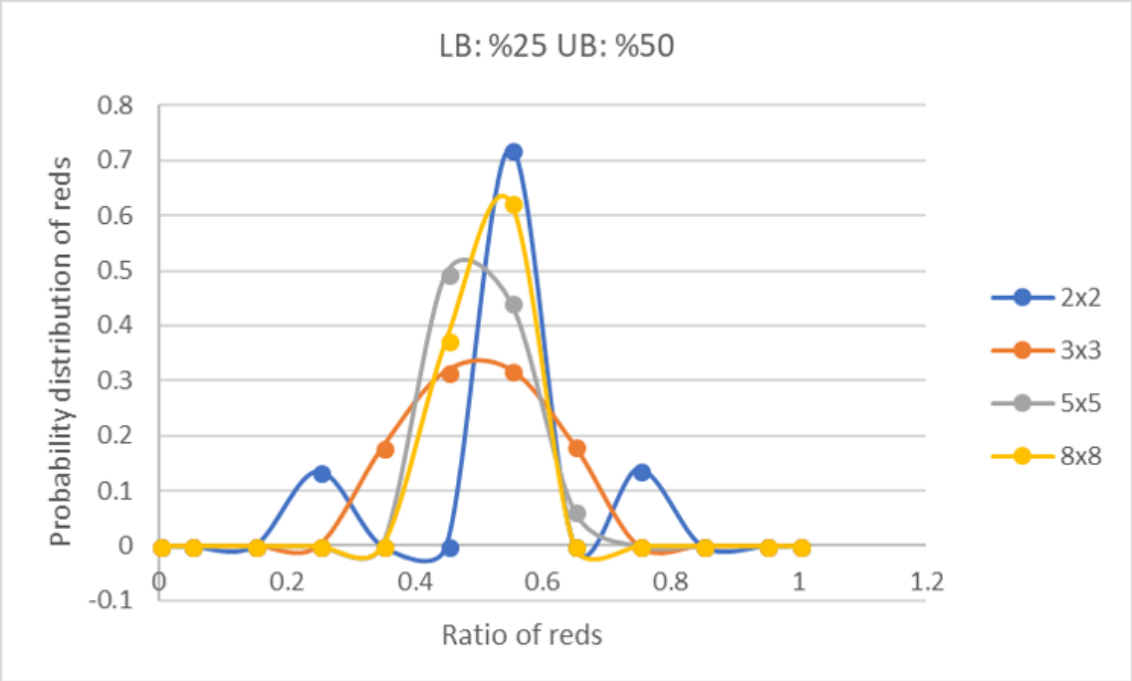


Figure 5.4: Square analysis for $\tau_g^l = \tau_r^l = 1/4$ and $\tau_g^u = \tau_r^u = 1/2$. Since the upper bound models do not allow the complete segregation, in other words fully monochromatic clusters, the probability distributions of different neighborhood radius do not have tails.

As can expected from the snapshot of maze case, we observe integrated regions more frequently around 0.5 probability even if smaller square sizes.

5.1 Distribution of neighbor proportions in upper bound models

In this section we calculate the neighbor fractions of all agents in the network and draw the probability distribution of these fractions and then we again repeat this method for different neighborhood radius configurations.

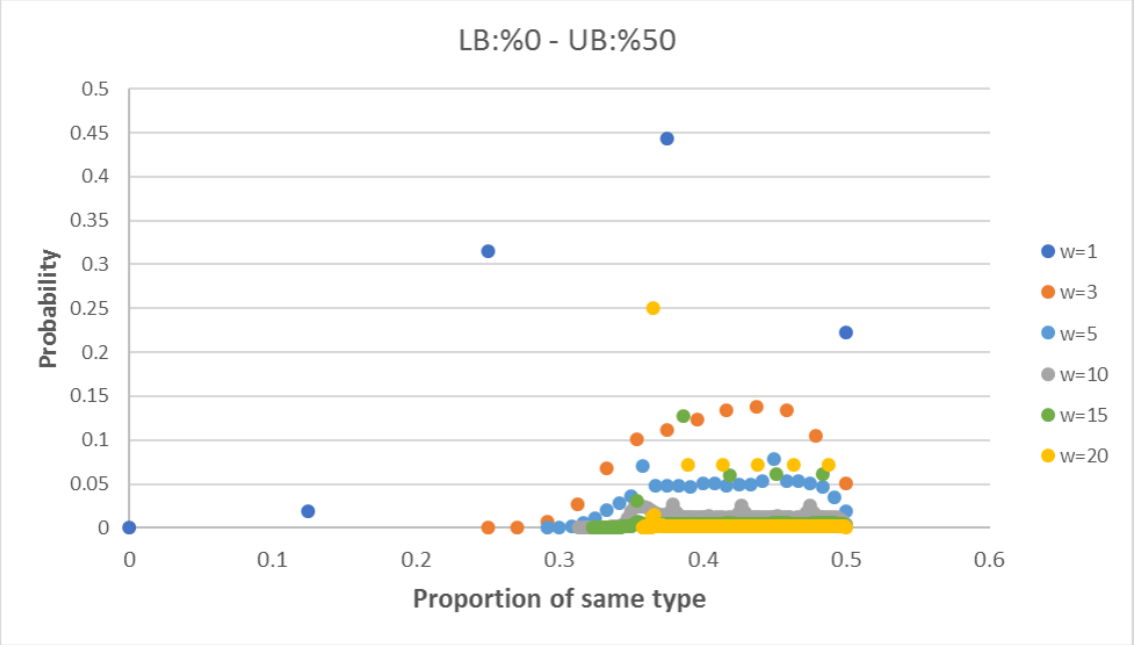


Figure 5.5: Probability distribution of proportion of same type respect to w for $\tau^l = 0$ and $\tau^u = 1/2$. Although the $\tau^l = 0$,starting from w=3, we do not observe any neighbor proportions less than 0.25. In fact, when the neighborhood radius increases the minimum observed neighbor proportion also increases.

As we can observe from the data which is represented as a graph above, even though the lower bound of the agents are 0, at the system equilibrium we do not observe any agents with same type fraction of 0. This situation evolves even more harshly when the neighborhood radius starts to increase; for instance for w = 20 the minimum same type neighborhood fraction one can observe is high as and around %35. This result is quiet interesting since, even the agents independent are independent as to have no neighbors from their own type, the equilibrium state shows us this is not the possible equilibrium configuration for this system with this input setup.

Table 5.1: Hamiltonian energy statistics of $\tau^l = 0$ and $\tau^u = 1/2$

%0-%50	Mean	std	Half width	C.I	
w1	-2.268	0.02	0.016	-2.252	-2.284
w3	-8.381	0.12	0.084	-8.296	-8.465
w5	-20.025	0.26	0.185	-19.839	-20.211
w10	-74.042	2.86	2.046	-71.995	-76.089
w15	-148.02	6.80	4.864	-143.156	-152.884
w20	-287.305	0.66	0.474	-286.831	-287.779

Starting from w =3 , both $\tau^l = 0$ and $\tau^u = 1/2$ $\tau^l = 1/4$ and $\tau^u = 1/2$ configurations give us almost same distribution graphs.

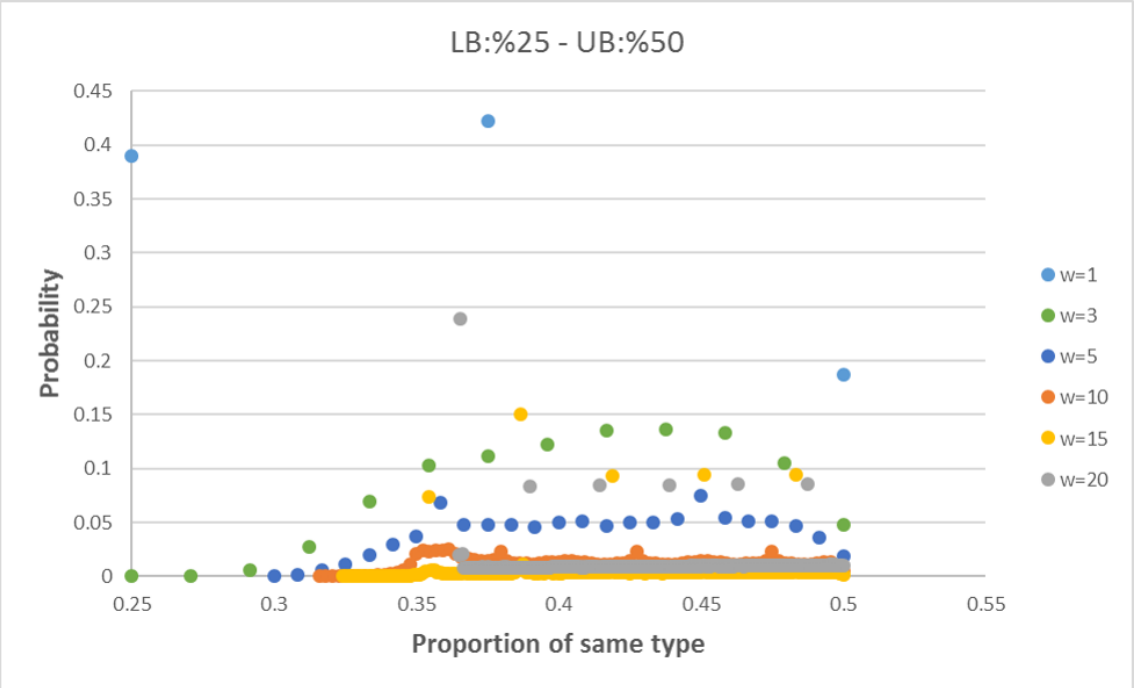


Figure 5.6: Probability distribution of proportion of same type of agents in the neighborhood respect to w for $\tau^l = 1/4$ and $\tau^u = 1/2$

Table 5.2: Hamiltonian energy statistics of $\tau^l = 1/4$ and $\tau^u = 1/2$

%25-%50	Mean	std	Half width	C.I	
w1	-2.48	0.043	0.031	-2.448	-2.511
w3	-8.4	0.089	0.063	-8.336	-8.463
w5	-20.018	6.291	4.500	-15.517	-24.518
w10	-74.5	3.320	2.375	-72.125	-76.875
w15	-151.991	5.722	4.093	-147.897	-156.085
w20	-287.463	0.836	0.598	-286.864	-288.061
w30	-609.448	2.287	1.636	-607.811	-611.084

Proportion of same type agents is greater than or equal to 0.25 after some critical value of radius w when $\tau^l = 0$ and $\tau^u = 1/2$

5.2 Time evaluation of neighbor distribution

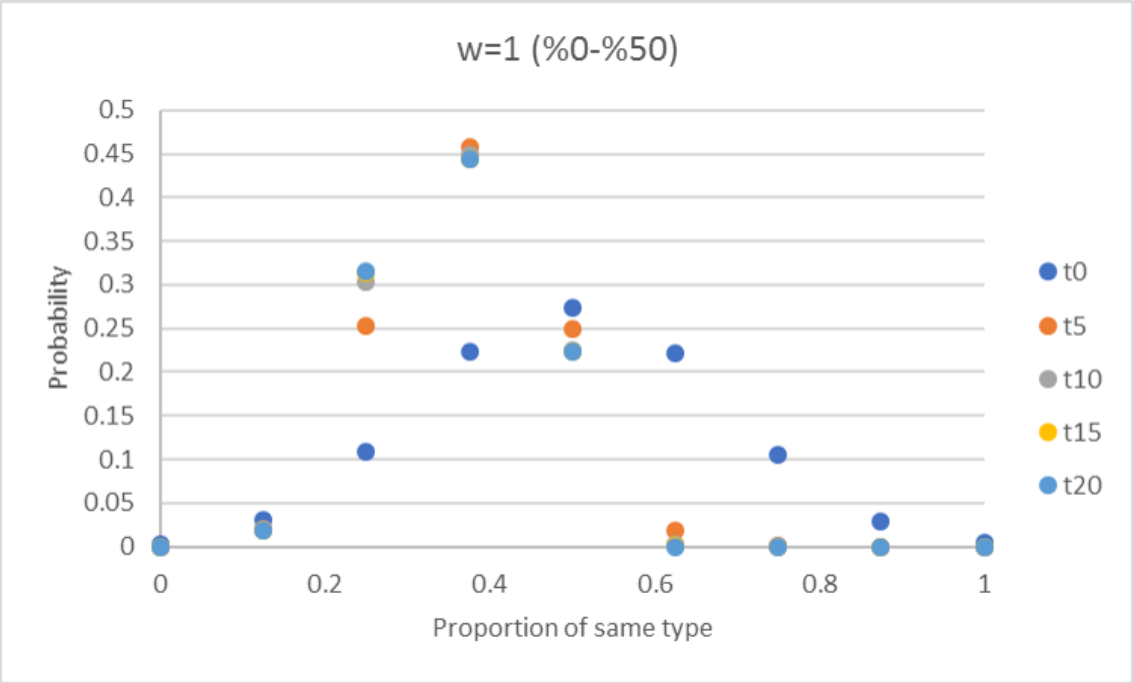


Figure 5.7: Proportion of same type in $w = 1$ for different time steps. The initial configuration is symmetric around 0.5. However, final distribution is shifted to the left.

To understand the above figure, the reader should consider that one time step in the algorithm starts from the creation of unhappy agents list and lasts until there is no unhappy agent left in that list. For instance again in the same figure, we can say that at time "t5" we know that the unhappy agent list at that time is the fifth list that is created since the initial time.

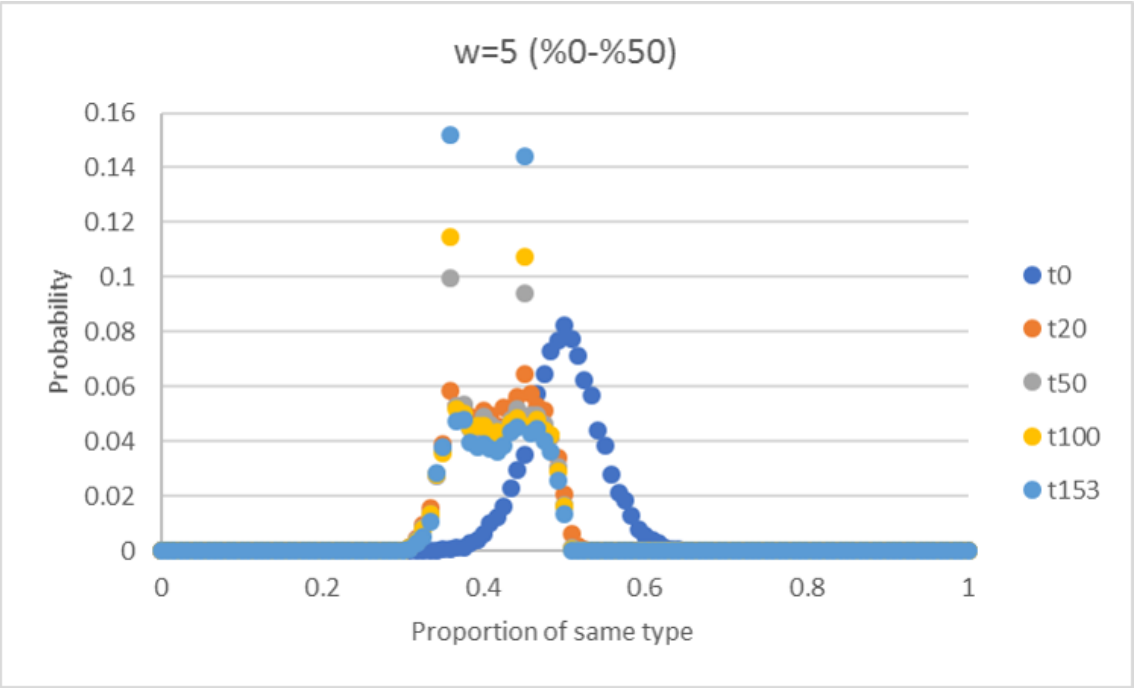


Figure 5.8: Proportion of same type in $w = 5$ for different time steps. For w=5, same as w=1 the mean of the same type distribution move to the left since the upper bound is bounded by 1/2

In the above even if the algorithm ends at t153, the figure gives us a clue at t20; about almost which proportion types are going to dominate the process.

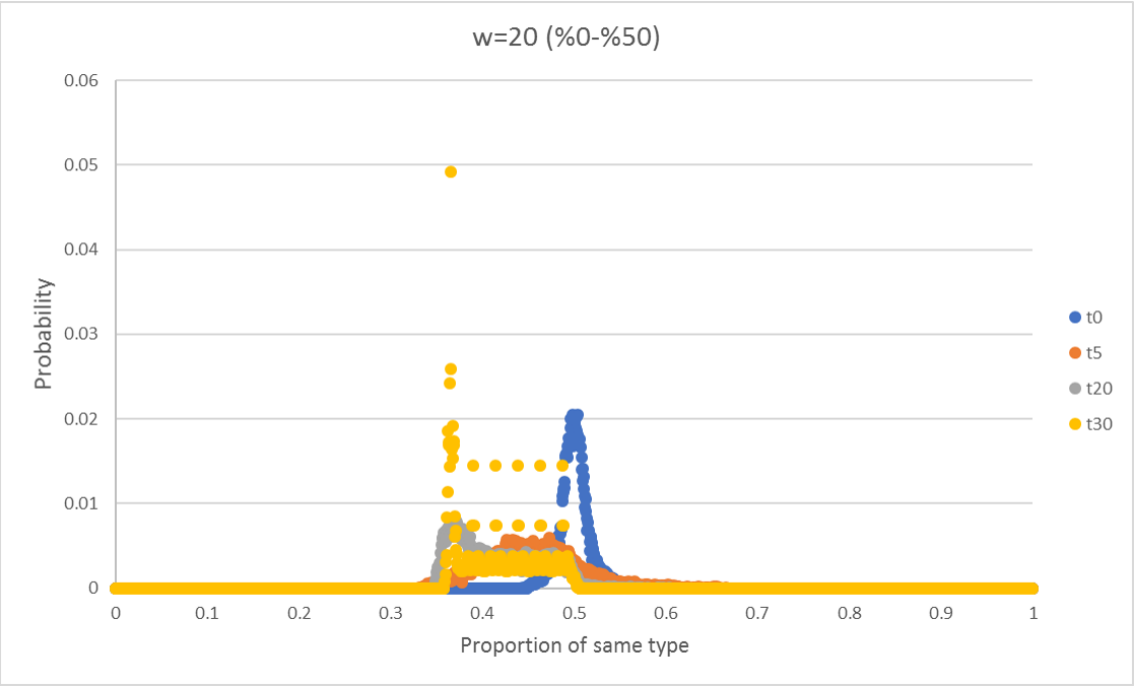


Figure 5.9: Proportion of same type in $w = 20$ for different time steps. The final state distribution shows us that the equilibrium only consists of certain proportions of same types.

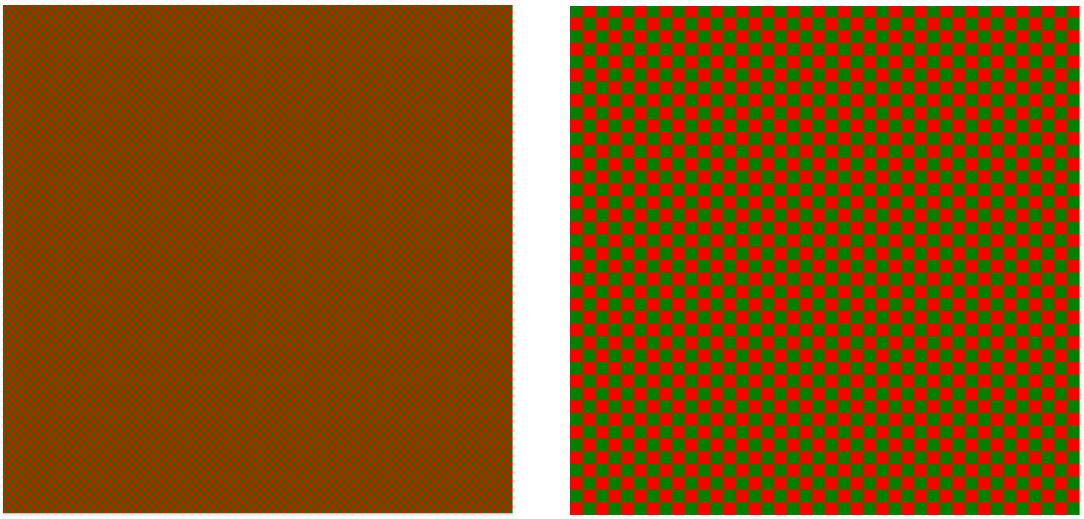
5.3 Single Point Bound Models

Since single point bounds which basically exists when upper bound equals to lower bound value do not reach equilibrium when the simulation starts with uniformly distributed network, we consider to start the simulation from special configurations. In this part we analyzed two different initial configurations for the simulation. These are chess boards and line by line structure. Zhang[22] analyzed this patterns for Schelling model with utility functions.

5.3.1 Chess Board Structure

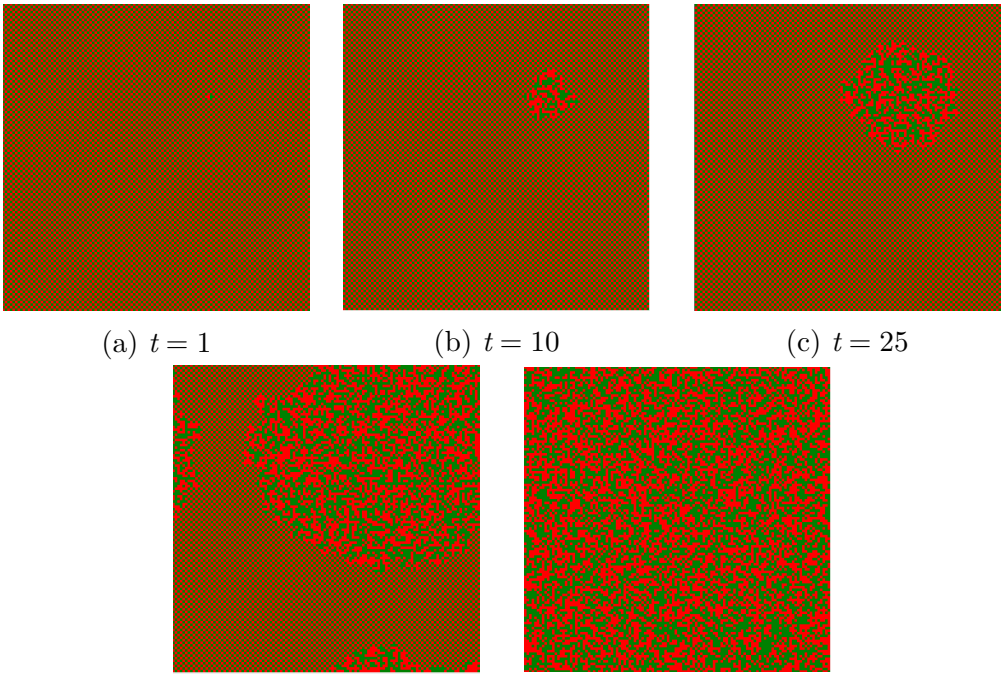
If we start the simulation from chess board equilibrium since this setup ensures $\tau^{l,u} = 1/2$ then because of perfect equilibrium no change will happen. However, if we distort only a single agent which is uniformly at random chosen from the lattice, this created distortion causes cascading effect. By distortion we mean changing the color of the agent. Even we change the color of random agent, the agent still become happy since the ratio of same type of neighbors around still 1/2 but this change of color cause unhappiness around that agent such as all 8 of its neighbors become unhappy. Therefore, when these unhappy agents initiate the flipping process of the algorithm the cascading effect grows and finally all of the lattice reaches the

instability.



(a) 120 x 120 chessboard (b) 40 x 40 chessboard

Figure 5.10: Examples of chess board structure. In each line, green and red agents are aligned one by one.



(a) $t = 1$ (b) $t = 10$ (c) $t = 25$

(d) $t = 50$ (e) $t = 100$

Figure 5.11: Time evaluation of chessboard structure. When a single agent is forced to flip, the cascading process immediately starts and distorted cluster emerges. Finally, all of the network lose its stability.

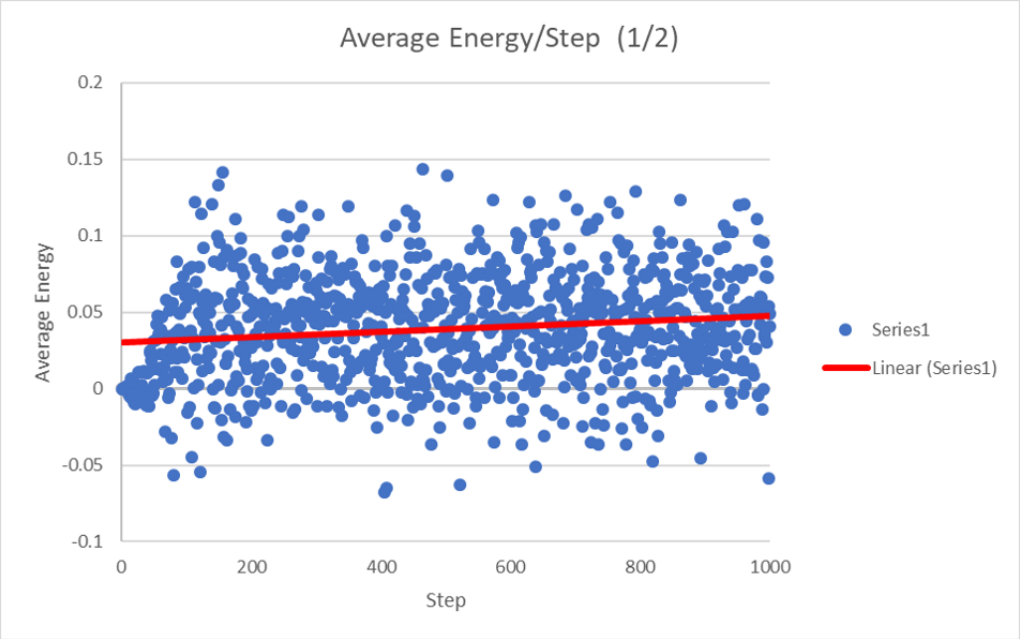


Figure 5.12: The figure represents average energy of the network respect to time step. Since in the initial configuration we distort only one agent, the average energy of the system starts from close to zero and then when the algorithm starts and continues the average energy again vibrates around zero energy level but it has relatively positive slope in the end of the simulation.

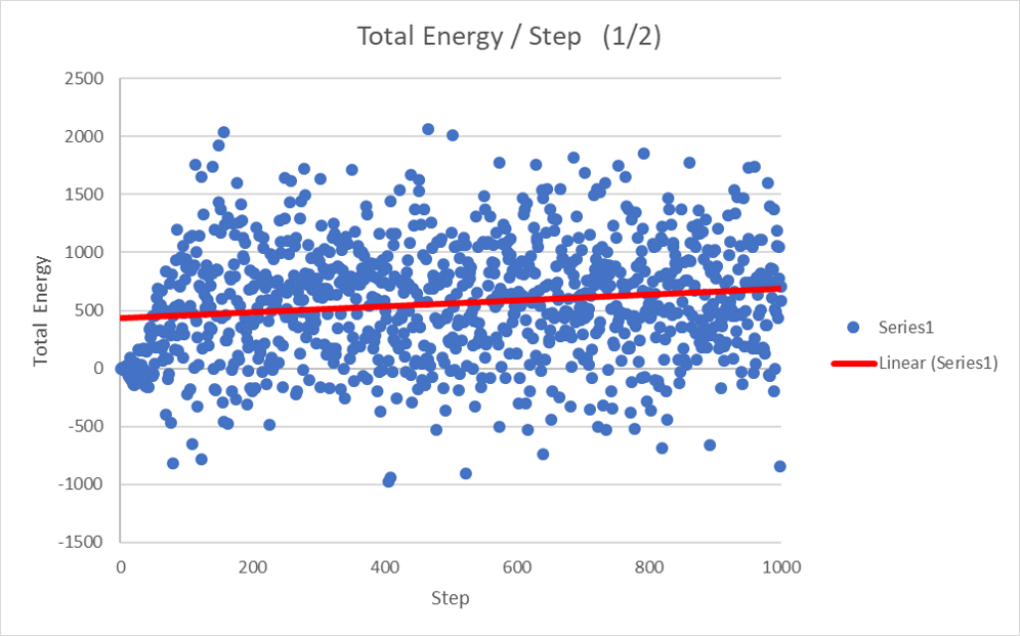
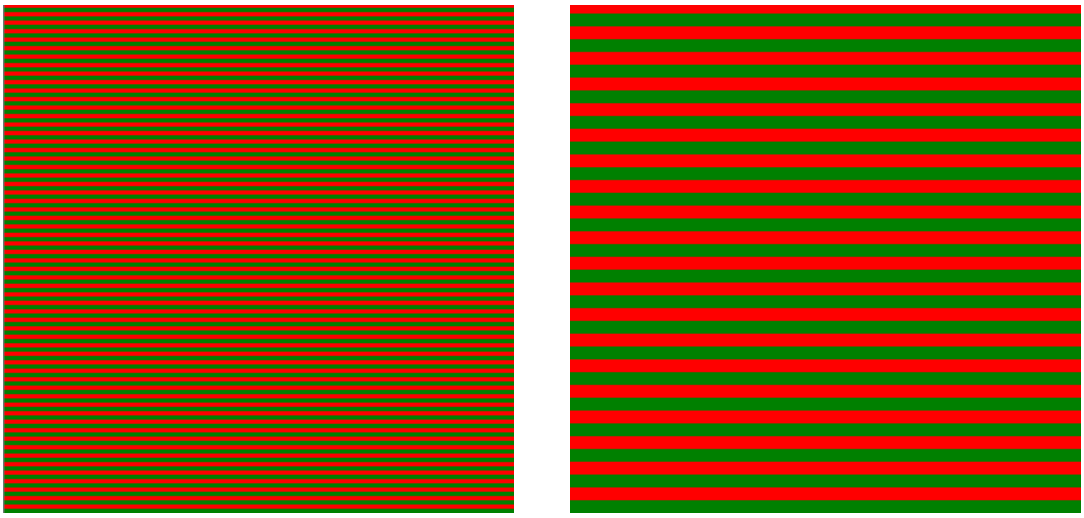


Figure 5.13: Total energy of the network respect to time step

5.3.2 Line by Line Structure

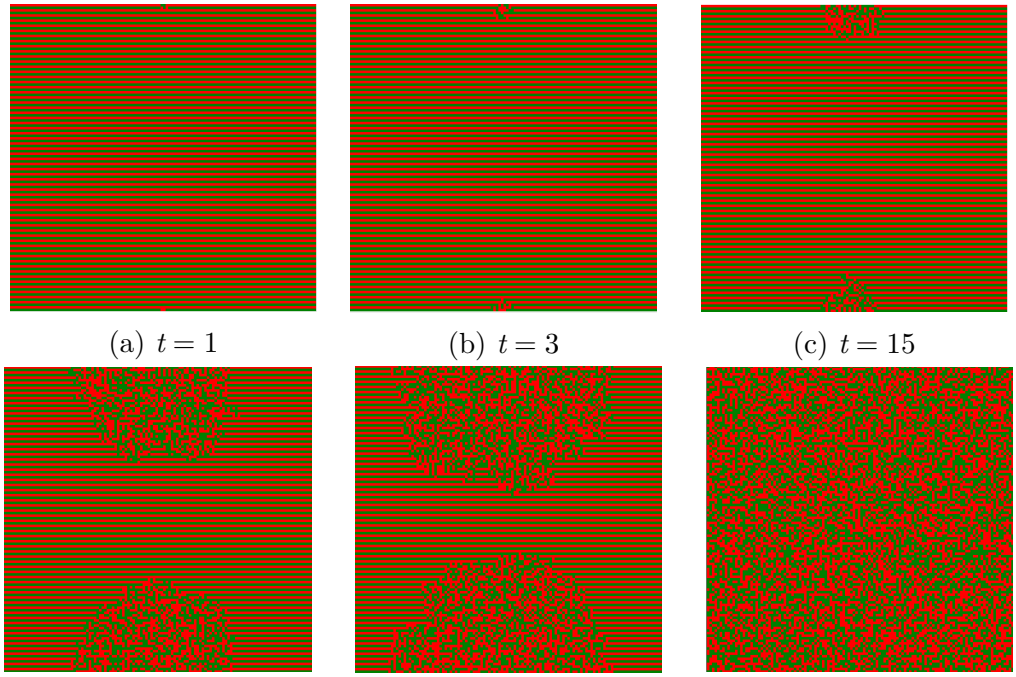
Line by line structure is equilibrium state when both two color types have $\tau^{l,u} = 1/4$. Again, with same logic as in chess board structure, when we start the simulation from

line by line structure nothing happens since all agents hold their fraction demand. However, when we distort a randomly chosen agent and change its color, cascading effect again shows itself. Contrast to chess board model, when we change the color of single agent that agent becomes unhappy since neighbor fraction becomes $\tau^* = 3/4$ and it is less than the threshold $\tau^{l,u} = 1/4$



(a) 120 x 120 line by line (b) 40 x 40 line by line

Figure 5.14: Examples of line by line structure



(a) $t = 1$ (b) $t = 3$ (c) $t = 15$

(d) $t = 45$ (e) $t = 60$ (f) $t = 100$

Figure 5.15: Time evaluation of line by line structure, When a single agent is forced to flip, the cascading process immediately starts and distorted cluster emerges. Finally, all of the network lose its stability.

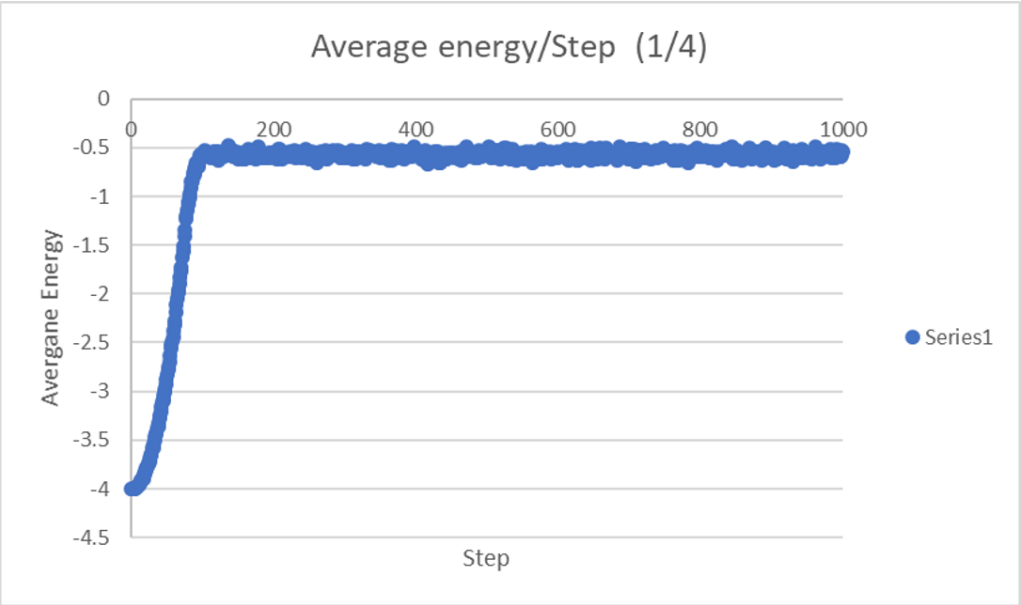


Figure 5.16: Average energy of the network respect to time step

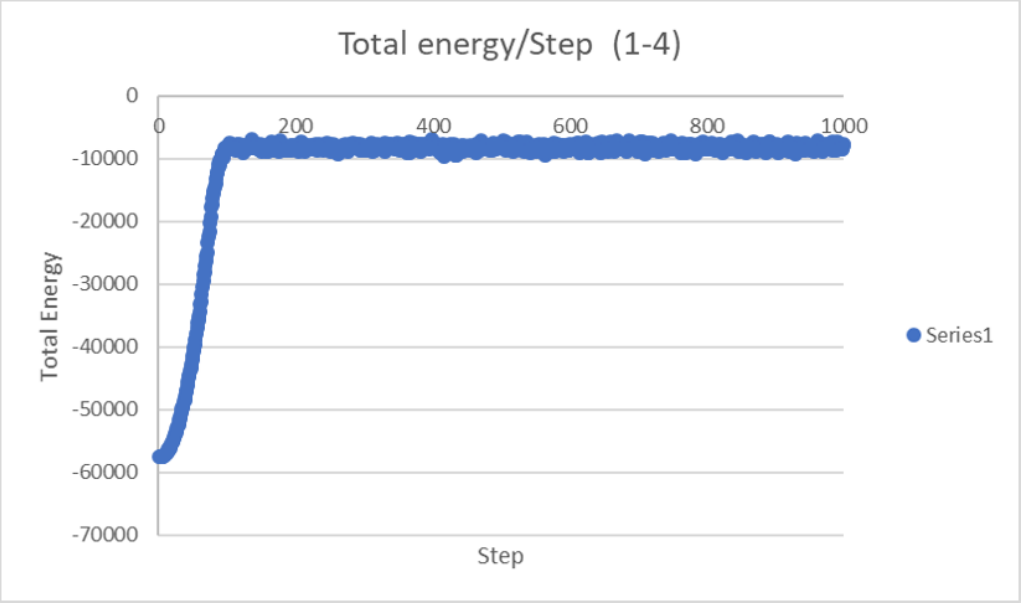


Figure 5.17: Total energy of the network respect to time step

Chapter 6

Applications

6.1 Product Adoption Model

The simulation model that has been investigated in this thesis can be modified and converted to marketing theory applications. For instance, we can adopt this model to simulate customer behaviors such as new product acquirement. We have created a simulation model to analyze new product adoption behaviors in a regular lattice network. The model that we created is similar to one which is created by Campbell [5].

Our model has 2 steps: first step or phase is the spreading of the information of a new product in the market and the second step is the decision process about acquiring the product or not. This application model differs with previous model which is described in this thesis, by in terms of information flow process which is an extension and the way of the creation of unhappy individuals list.

6.1.1 The First Phase (Information Phase)

That information spreading process is similar to the classical linear threshold model as it defined in [12]. However, different from that model, it is sufficient condition to receive the information if an individual has even only one potential neighbor. The model works as follows, at first, for each individual, a purchasing power is assigned between 0 and 1 by uniformly at random. Then a global price index is determined again uniformly at random assigned for the all network. The individuals that have sufficient purchasing power (purchasing power needs to be more than or equal to global price index) to acquire the new product named as potential individuals. However, the individuals that have purchasing power less than the global price index become stubborn. This means that they no longer have any effect on the simula-

tion; they can not diffuse an information or not take any actions but they can affect decision of others with their presence in the neighborhood.

After the role assignments, from a randomly determined square region which is basically the neighborhood of some size of a randomly chosen individual, the information of existence of a new product begins to spread. In other words, that predetermined neighborhood is the first buyers of the product except the stubborn ones in this neighborhood and they start to diffuse the information of a new product to their non-stubborn neighbors. As mentioned before, a stubborn individual can not transmit the information to their neighbors, only the potential individuals can spread the information to their neighbors which are also potentials. When there is no one left from the potential individuals that can be reached the information flow process stops and it is the end of the first phase of the simulation. Remember that there might be some potential individuals who have sufficient purchasing power to acquire the product but cannot receive the information about it because of being surrounded by stubborn individuals. Therefore, in such a case they do not have any effect on the model anymore.

If the main objective of the model is maximizing the product adopters as much as possible by changing initial adoption region, in our case it is a randomly chosen square region, percolation theory and some optimization approaches will be needed. Thus, in such a case choosing the best set of a initial adopters will be though question to answer. Kempe et al. addressed this problem as influence maximizing problem in their study.[12] They proved in their work that finding the best set is a NP-hard problem for the linear threshold model which is equivalent to lower bound threshold model in our study.

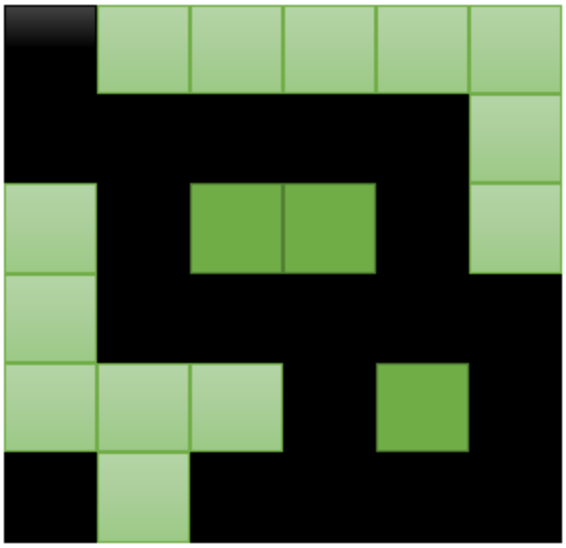


Figure 6.1: The representation of information spreading process.The dark green agents are not informed about the new product. The black agents are the stubborns.

6.1.2 The Second Phase (Decision Phase)

The second phase initiates immediately after the end of the information spreading process. The individuals start to decide if to acquire the product or not with regarding to majority rule. Each individual will calculate the proportion of their neighbors who decided to acquire (are informed about) the product at that moment and then will decide to acquire the product by comparing this ratio with the their lower bound value. If the proportion of that neighbors less than the lower bound threshold of that individual, the individual decide not to buy the product and behave like it doesn't have any information about the new product. From this point of view, this application model differs from the previous Ising- Schelling model. The reason of that is in the previous model an individual can switch its type and turn again the its previous type during the simulation. However, in this application model, if the informed individual decide not to acquire the product and switch its type it cannot return to its previous type again in any time of the simulation. In other words, it cannot decide to buy the product. We can describe this process in the second phase as an elimination process rather than the cascading process. Finally, when there is no one left to make a decision, the remaining informed individuals buy this new product.

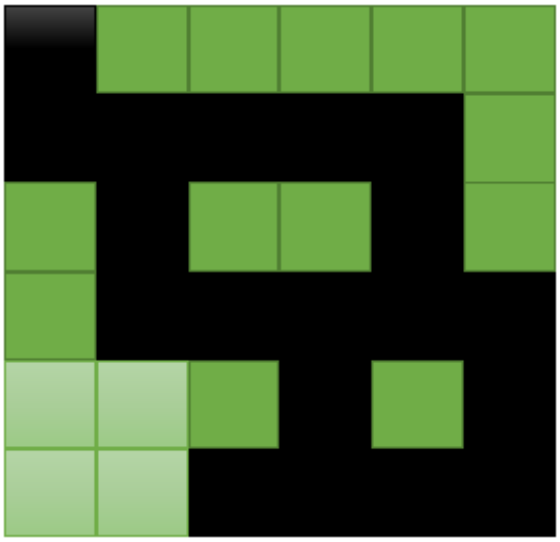


Figure 6.2: The dark green agents decide to not to buy the new product and there only four agents decide to buy which are the light green ones.

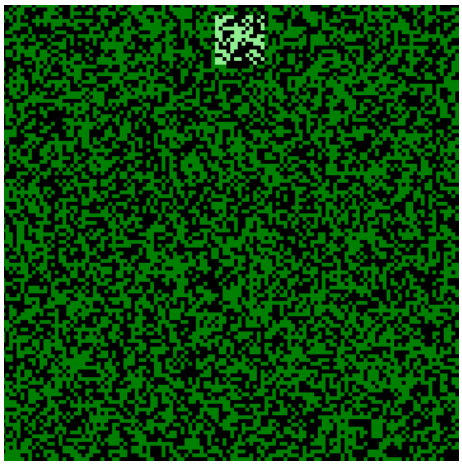


Figure 6.3: The above figure represents the initial configuration of the first phase with p = 0.5. A random individual has been chosen uniformly at random and its neighborhood of w = 6 is initiated as first buyers of the new product except the stubborn ones. Therefore, the light green individuals are the first adopters and the dark greens are capable to acquire the product.

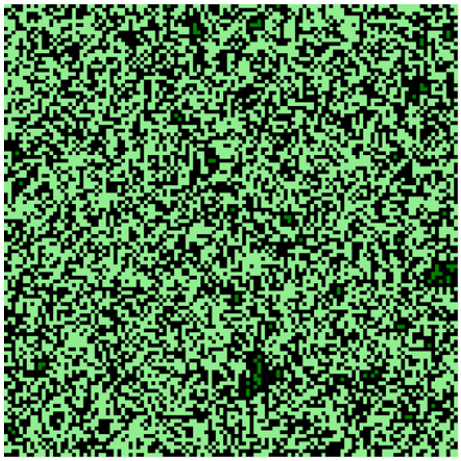


Figure 6.4: The above figure represents the initial configuration of the network immediately after the first phase. The light green individuals are aware of the new product and have financial capability to acquire it. The dark green individuals do not have information about the new product. Finally, the dark individuals, since they don't have sufficient power for acquiring they have also no effect to the information spreading process.

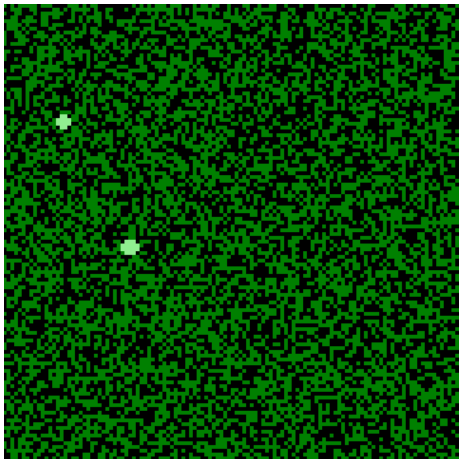


Figure 6.5: Final equilibrium state of the adoption model network with lower bound = 0.5. The above figure represents the end of the second phase. Comparing with the initial configuration, almost all of the light green individuals decide not to buy the product after they asked about the decisions of their neighbors. Therefore, they switch their color to dark green and the remaining light greens are willing to buy the product.

If we examine the Figure 6.3, we can observe that the 1/2 lower bound threshold value discourages almost all of the informed individuals to not buy the new product. There is only small proportion of them that decide to acquire it.

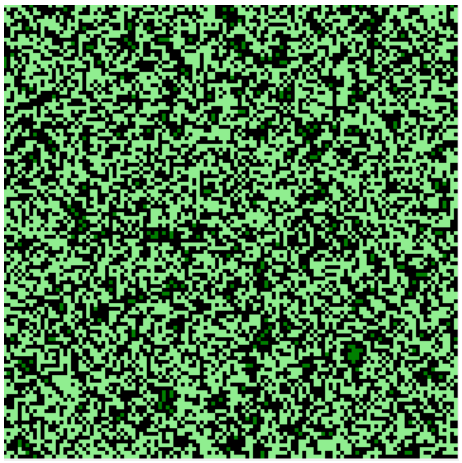


Figure 6.6: Final equilibrium state of the adoption model network with lower bound = 0.25

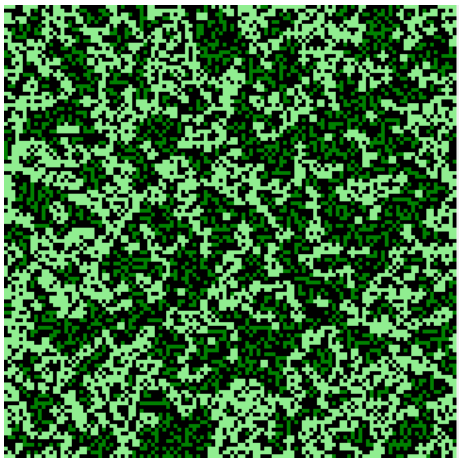


Figure 6.7: Final equilibrium state of the adoption model network with lower bound = 0.375

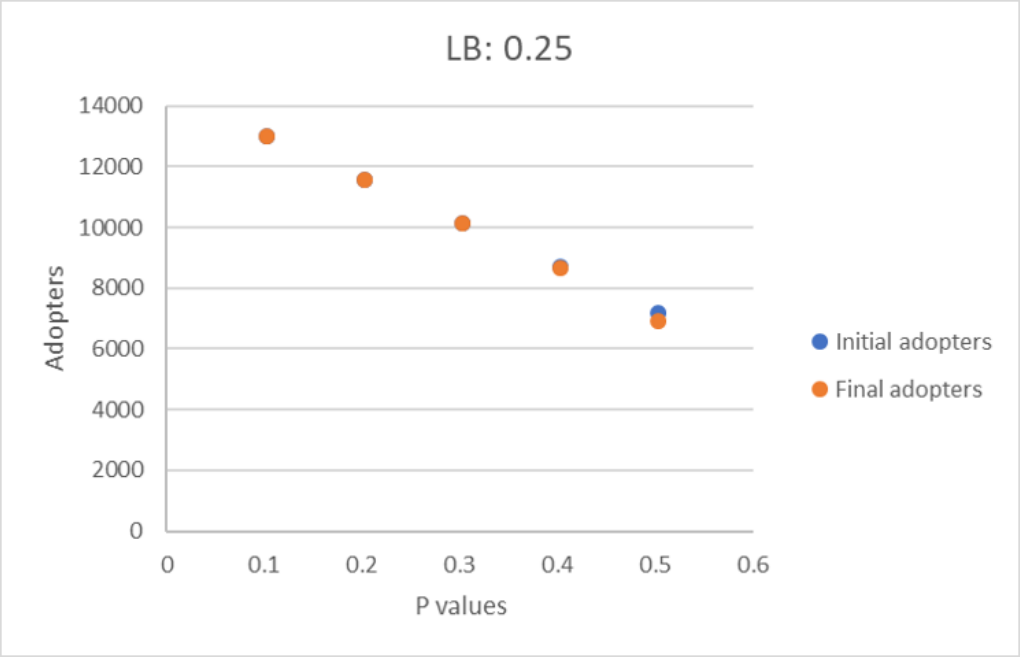


Figure 6.8: Comparison of initial and final adopters respect to different p values for lower bound 1/4

When we examine the above figure,, the initial and the final adopters line overlap over each other. This means that 1/4 lower bound value is almost sufficient value for potential individuals to acquire the new product. We observe that only small proportion of informed individuals renounce to buy the new product for 0.5 p value.

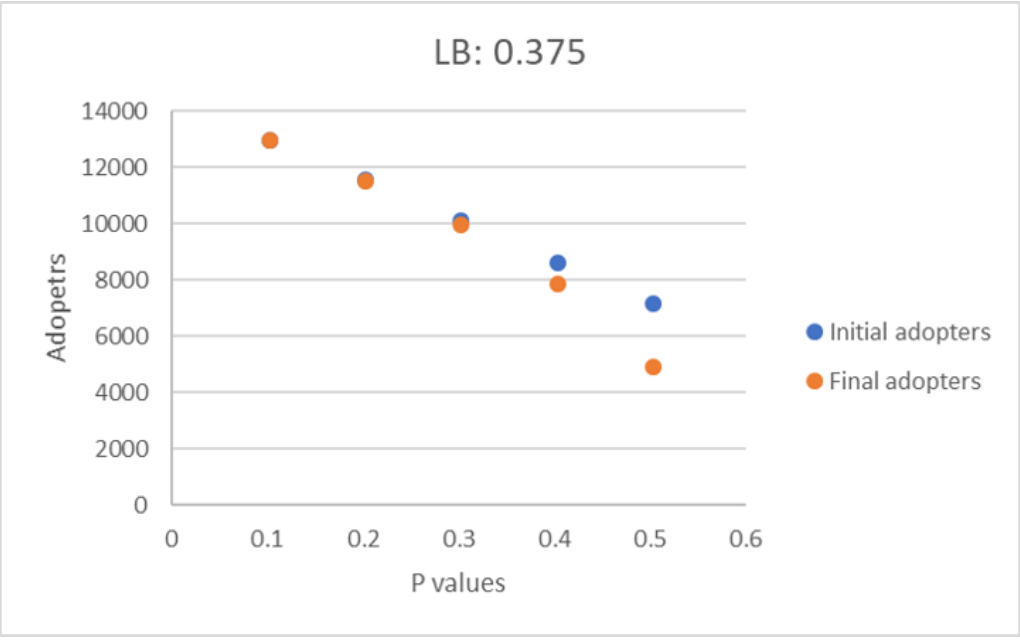


Figure 6.9: Comparison of initial and final adopters respect to different p values for lower bound 3/8

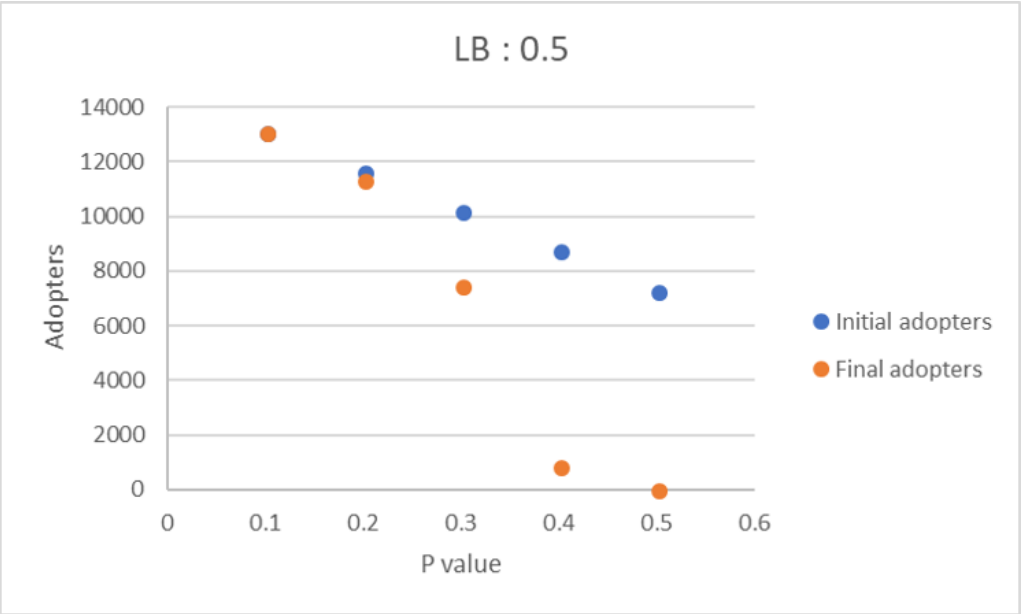


Figure 6.10: Comparison of initial and final adopters respect to different p values for lower bound 0.5

Contrary to Figure 6.6 we observe that when the lower bound increases the gap between the initial adopters and the final adopters also increases for higher p values. This means that even for high proportion of information propagation, because of highness of threshold, the individuals could not decide to acquire the new product.

Table 6.1: Fractions of initial and final adopters by different p and lower bounds

p value / lower bound	0.25		0.375		0.5	
0.1	0.909	0.909	0.908	0.908	0.908	0.909
0.2	0.810	0.810	0.801	0.808	0.809	0.791
0.3	0.710	0.709	0.709	0.698	0.708	0.521
0.4	0.609	0.609	0.607	0.552	0.609	0.061
0.5	0.504	0.488	0.505	0.349	0.505	0.004

Chapter 7

Conclusion And Future Study

Despite the closer similarity between our Ising-Schelling model and classical Ising model, there is crucial difference about accepting equal number of same type of neighbors in the neighborhood and which allows the simulation ends up with different equilibrium states than the classical Ising model. We analyzed different starting proportions with different lower bound thresholds for each agent types to observe effects on final ratios and find that it is possible to reach ground state level by manipulating the initial ratios and the lower bound thresholds. Then we define a new method named as square analysis method to analyze the size of monochromatic clusters. With the help of square analysis method we extract the probability distribution of monochromatic squares. The results say us that, the size of monochromatic squares is independent of network size but highly depends on the neighborhood radius. Additionally, we add upper bounds for each agent type to extend the model. This configurations resulted in maze type structures when upper bound is close to 0.5. Finally, we define an application for new product adoption in a market. These results say that, the chance of survival of a new product in a market highly depends on the lower bound thresholds of the individuals and the chance of the information spreading process.

For a future study, there might be different configurations ans setups to analyze the behavior of the model. One idea can be the addition of the cost for each movement; for instance, in glauber dynamics model we can implement a cost factor for flipping process. In the model, we know that an unhappy agent changes its type and flips for sure in zero temperature model and we can modify this model by making harder for one type of agents to flip. Therefore there will be different coefficients for each agent types for energy calculations.(Remember that for the classical model coefficients are equal to 1 for each agent type or spins)Another idea related to cost issue might be adding penalty to far locations from the current positions of the agent and this idea can be implemented to Kawasaki dynamic, if the two agents are far from each other

and it do not worth so much to swap the positions they will choose another agents to switch their positions.

Furthermore, to extend the analysis forward to social science perspective, the number of agent types can be increased and the relations between different agent clusters can give an idea about local allocations of them. In an addition to this idea, the effects of having minority and majority populations can be analyzed to observe for what thresholds the minorities ca still exist in the simulation model especially in the algorithms like Glauber dynamic.

If we want to adapt this model to real cities and settlements as much as possibly we can, then there might be geographical features such as rivers, seas, mountains that hard to settle and etc. By extending the model with these features we can define a cost for each location regarding to their geographical advantages and the agents who are unhappy and can afford to resettle one of these places can make their actions. Indeed, people usually do not prefer the live on or near the mountain if there is a chance to live near the city centre where the job opportunities are higher than the rural areas. For instance, if we add a river which completely crosses the city and divides it to two separate parts, it would be a interesting idea to catch if there is a complete segregation on different sides of the river.

Addition of new features will enlarge our objective function with new variables for each agent. In order to analyze geographical cost issue, we need to add new variable and our previous hamiltonian energy function will be a some linear combination of proportion of same type of neighbors and the geographical value of the new settlement.

Bibliography

[1] George Barmpalias, Richard Elwes, and Andrew Lewis-Pye. Unperturbed schelling segregation in two or three dimensions. *Journal of Statistical Physics*, 164(6):1460–1487, 2016. 3, 3.1.1, 4, 4.1

[2] George Barmpalias, Richard Elwes, and Andrew Lewis-Pye. Digital morphogenesis via schelling segregation. *Nonlinearity*, 31(4):1593, 2018. 1, 3

[3] George Barmpalias, Richard Elwes, and Andy Lewis-Pye. Tipping points in 1-dimensional schelling models with switching agents. *Journal of Statistical Physics*, 158(4):806–852, 2015. 1

[4] Christina Brandt, Nicole Immorlica, Gautam Kamath, and Robert Kleinberg. An analysis of one-dimensional schelling segregation. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 789–804. ACM, 2012. 1

[5] Arthur Campbell. Word-of-mouth communication and percolation in social networks. *American Economic Review*, 103(6):2466–98, 2013. 6.1

[6] Alihan Celik. How do local interaction patterns affect the global behavior of a community: Schelling models revisited. 2018. 5

[7] P Chen and S Redner. Majority rule dynamics in finite dimensions. *Physical review E*, 71(3):036101, 2005. 1

[8] B Chopard and M Droz. *Cellular automata*. Springer, 1998. 1

[9] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982. 1

[10] Nicole Immorlica, Robert Kleinbergt, Brendan Lucier, and Morteza Zadomighaddam. Exponential segregation in a two-dimensional schelling model with tolerant individuals. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 984–993. SIAM, 2017. 1, 3, 4

[11] Ernst Ising. E. ising, z. phys. 31, 253 (1925). *Z. Phys.*, 31:253, 1925. 1

[12] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003. 6.1.1

[13] Paul L Krapivsky and Sidney Redner. Dynamics of majority rule in two-state interacting spin systems. *Physical Review Letters*, 90(23):238701, 2003. 1

[14] Hidetoshi Nishimori. *Statistical physics of spin glasses and information processing: an introduction*, volume 111. Clarendon Press, 2001. 1

[15] Hamed Omidvar and Massimo Franceschetti. Self-organized segregation on the grid. *arXiv preprint arXiv:1705.08586, 2017 - arxiv.org*, 2017. 3

[16] TC Schelling. Schelling micromotives and macrobehavior (cloth)(fels lectures on public policy analysis). 1978. 1

[17] Thomas C Schelling. Models of segregation. *The American Economic Review*, 59(2):488–493, 1969. 1

[18] V Spirin, PL Krapivsky, and S Redner. Fate of zero-temperature ising ferromagnets. *Physical Review E*, 63(3):036118, 2001. 3.2

[19] Dietrich Stauffer and Sorin Solomon. Ising, schelling and self-organising segregation. *The European Physical Journal B*, 57(4):473–479, 2007. 1

[20] H Peyton Young. *Individual strategy and social structure: An evolutionary theory of institutions*. Princeton University Press, 2001. 1

[21] Junfu Zhang. A dynamic model of residential segregation. *Journal of Mathematical Sociology*, 28(3):147–170, 2004. 1

[22] Junfu Zhang. Tipping and residential segregation: a unified schelling model. *Journal of Regional Science*, 51(1):167–193, 2011. 5.3

Appendices

Appendix A

Different movement algorithms for agents

- Discrete utility: in this algorithm an unhappy agent will move to first vacant place that is randomly chosen.
- Continuous utility: in this algorithm, an unhappy agent will try to choose best vacant place that will maximize the utility of this agent.
- Glauber Dynamics: An unhappy agent changes its type to other, in this dynamic there is no vacant places in the network.
- Kawasaki Dynamics: Randomly chosen two unhappy agents from the unhappy list interchange their positions, there might be vacant places in the network.
- Swap if better: Randomly chosen two unhappy agents interchanges their positions if the total utility of new places is higher than the previous locations' utility.
- All random: An unhappy agent randomly chooses a vacant place and change its position not matter new position makes it happy or not.

Appendix B

Alternative features for the algorithm

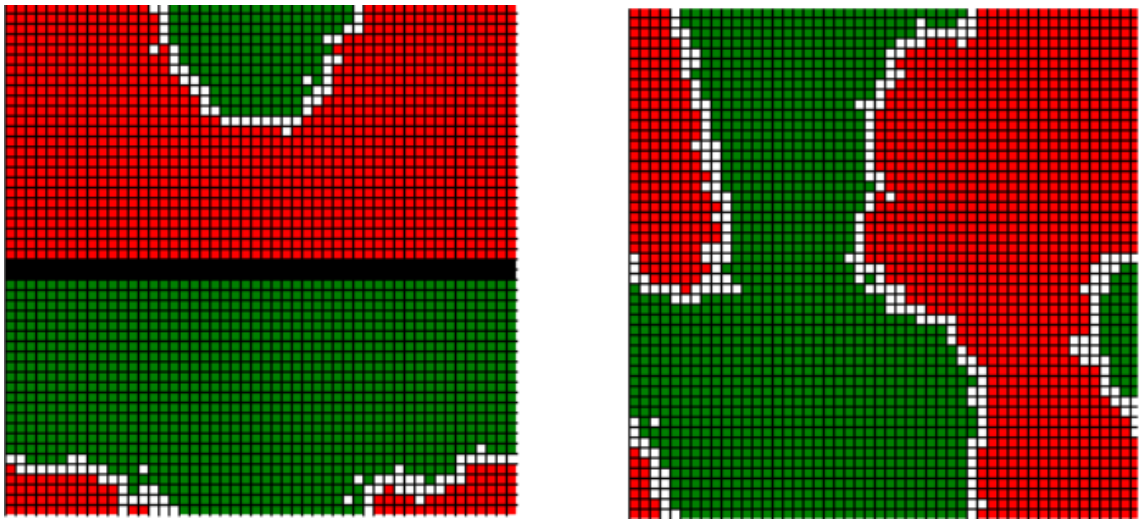


Figure B.1: An example of geographical features with 1150 agents for each type, in the left the black line represents a river in the city. As we can observe from two figures that the immovable cells that represents the river allows agent to segregate more in the sides of the river.

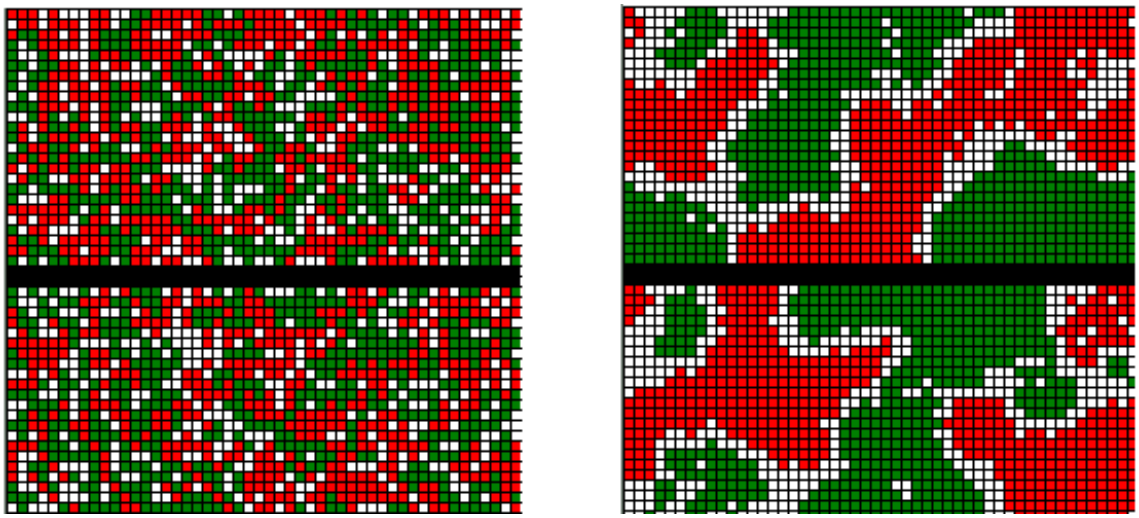


Figure B.2: An example of geographical features with 910 agents for each type, in the left the black line represents a river in the city. The immovable cells that represents the river cannot completely separates the agents as two clusters since there are many vacant positions to allocate.

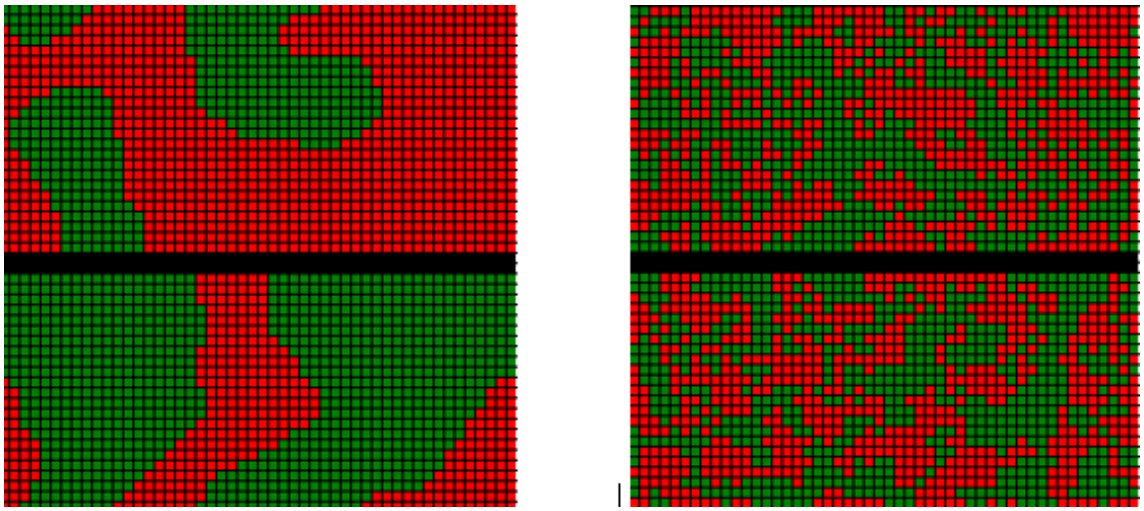


Figure B.3: An example of geographical features with 910 agents for each type, in the left the black line represents a river in the city. The immovable cells that represents the river cannot completely separates the agents as two clusters since there are many vacant positions to allocate.

Appendix C

Pseudo-codes of algorithm of process dynamics

```
Algorithm (Glauber dynamics):
      Main Function ()
   {
      Collect all the unhappy agents in a set (Create unhappy agent list);

      WHILE (count of unhappy agent list greater than zero)
         Roll a total number of unhappy agents sided dice;

         IF (the agent with dice indexed is still unhappy (Choose random agent
         from the list and check))
            Change color of the agent and its type;
            Remove it from the unhappy list;

         ELSE
            Remove it from the unhappy list;
     }
      WHILE (Not all the agents in the network are happy)
         Repeat Main Function();
```

```
First algorithm (Kawasaki dynamic ):

FOR (0 to total number of agents)
   Choose random two agents from both types;
```

```
   IF (Those agents are both unhappy)
      Interchange their positions;

   ELSE
      Choose another two random agents;
IF (One type of agents is all happy and others not)

   FOR (0 to total number of first type of agents)

      IF (Agent is unhappy)
         Change its position with random opposite type?s position

   FOR (0 to total number of second type of agents)

      IF (Agent is unhappy)
         Change its position with random opposite type?s position

WHILE (Not all the agents are happy)
   Return to first for loop;
```

Second Algorithm (Choose one unhappy agent and replace it with random one with another type):

```
FOR (0 to total number of agents)
   Choose random two agents from both types;

   IF (At least one of them are unhappy)
      Interchange their positions;
   Choose another two random agents;

WHILE (Not all the agents are happy)
   Repeat the upper For loop;
```

```
Third Algorithm (All Move randomly):

Choose random two agents from both types;

FOR (0 to total number of agents)
   Interchange their positions;
   Choose random two agents from both types;

WHILE (Not all the agents are happy)
   Repeat the upper For loop;
```