Summer Research Internship Report

21 May 2015 - 21 July 2015

Schelling Segregation Models

Submitted by:

Rachneet Kaur
M.Sc Mathematics, IIT Delhi

Under the guidance of:

Dr. Julien Randon Furling
SAMM, Universite Paris I, Pantheon Sorbonne, France

Abstract

The report is divided in 4 chapters discussing different variants of Schelling segregation models. In chapter 1, the model in [3] where the fixed switching agents are introduced at random locations in the initial grid is discussed and the plots are recreated concluding that the mixity of the system enhances as these type of agents were introduced in the system.
Chapter 2 discusses the effect of introduction of fixed agents inside a fixed annular ring and outer of the circle and in Chapter 3, fixed agents are placed randomly in a gaussian ring and the effects are noted specifically comparing the tolerance level at which complete mixity is attained in this case with the one in standard Schelling dynamics shows that this kind of allocation introduces complete mixity in the grid at a much lower tolerance.

Chapter 1

A variant of Schelling model with random allocation of fixed switching agents

1.1 Introduction

In this chapter, a scenario in which a fraction of fixed randomly allocated switching agents are introduced in the Schelling model is studied. The fixed sites may be interpreted as non-preferential housing allocation in a residential system.
It is observed that the random allocation of the fixed switching agents in the Schelling model enhances the contact density of the system as compared to the pure Schelling dynamics.
The objective of this chapter is to retrieve the results obtained in research paper [3].

1.2 Modified Schelling system with fixed switching agents

The system is a square grid with finite side length N and N^2 sites. Each site can be either vacant, occupied by a red agent, blue agent or a fixed switching agent. The fixed switching agent can be either red or blue with equal probability of both, but once it is examined, it switches its type with probability P_S.
We assume the occupation density of the system as ρ, the fraction of fixed agents to total number of agents in the system as f and an equal number of red and blue agents.
All agents are examined in random order.
If the examined agent is pure blue or pure red type, they move to a vacant site, chosen uniformly at random (referred as blind moves) with probability P_H if the proportion of opposite type in its moore neighbourhood is smaller than the defined tolerance level, T, otherwise if the examined agent is unhappy with its current position, the probability of moving to a randomly chosen vacant site is defined as P_U.
If the examined agent is a switching type, it remains fixed to its position, regardless of the type of its neighbours and switches its type with probability P_S.
We define T_M as the number of time steps, where one step refers to a complete round over all the agents.
For experimental purpose, the following values are chosen for the above mentioned parameters:

Parameter	Value
ρ	0.85
T	0.3
N	30
f	0.20
T_M	500
P_U	0.2
P_H	10^{-4}
P_S	0.05

Table 1.1: Simulation parameters

Here, P_U is kept significantly lower than 1 to reflect factors that may prevent unhappy agents to move, and P_H being non zero reflects the possibility that agents might make a wrong move due to economic factors.
Pure Schelling model reaches a segregated or steady state. The variable that is used to decide the segregation level is called contact density of the system.

Definition 1.2.1. *Contact density*
Contact density x(t) at time step 't' is defined as the average over all occupied sites of the ratio between number of neighbours of opposite type to the total number of non-vacant neighbours. We define x_∞ as the contact density at the segregated stage.

For the following four scenarios, the contact density is plotted for distinct time steps in Figure 1.1 and the MATLAB code for the same using blind moves and using non blind moves can be found in the Appendix.

1. No switching agents are present in the system.
2. Fraction f = 0.2 of deactivated switching agents are present in the system.
3. Fraction f = 0.2 of activated switching agents are present in the system which are activated in the beginning itself.
4. Fraction f = 0.2 of activated switching agents are present in the system which are activated after the half time.

The time taken to segregate for each of the cases is given in Table 1.2.

Observations:

- In absence of switching agents, system segregates quickly to $x_\infty = 0.06$.
- In presence of deactivated switching agents, system segregates bit slower to $x_\infty = 0.26$.
- In presence of activated switching agents, system segregates even slower to $x_\infty = 0.37$.

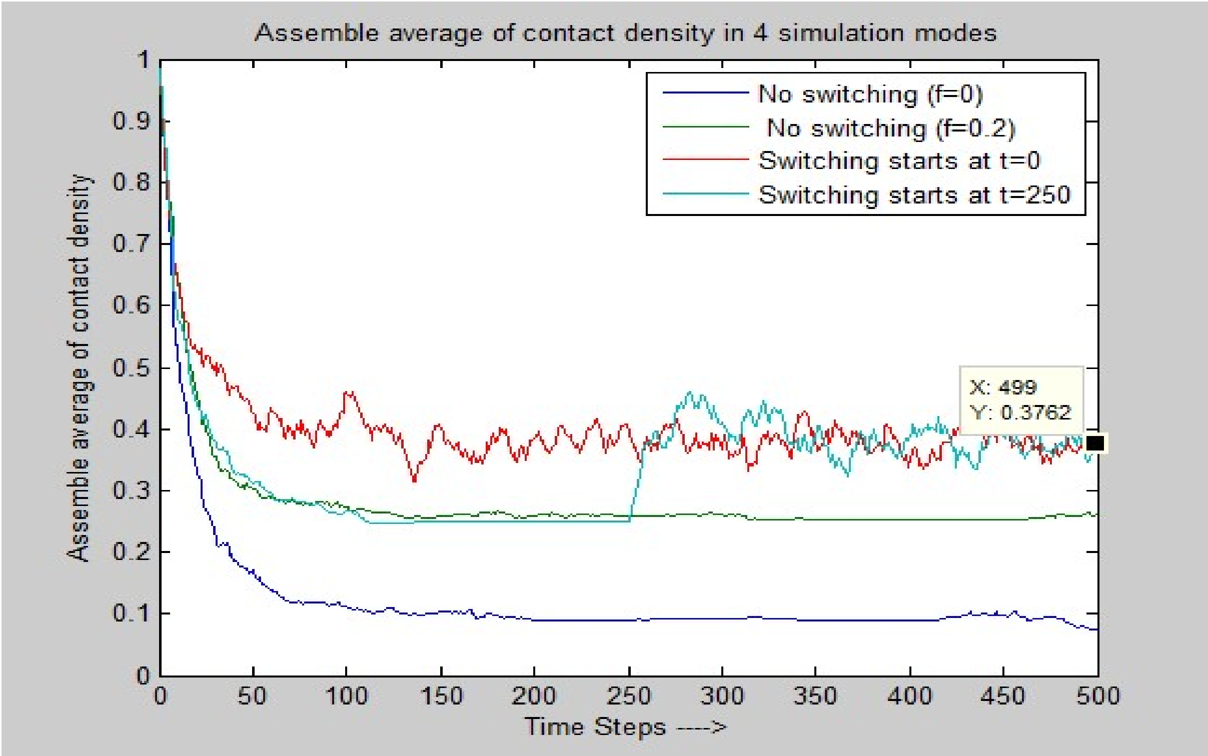


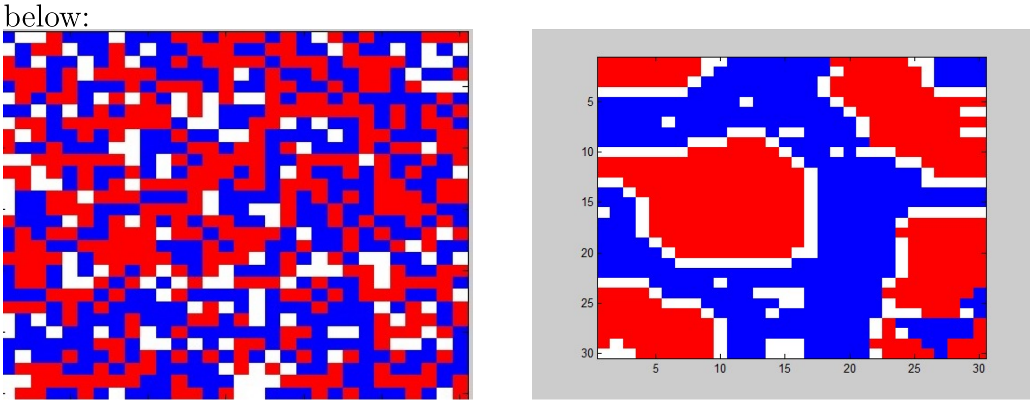
FIGURE 1.1:Contact density in 4 simulation modes

Case	Time taken to segregate (in sec)
No switching agents	39.3
Deactivated switching agents	41.4
Activated switching agents in the beginning	44.8
Activated switching agents after half time	43.2

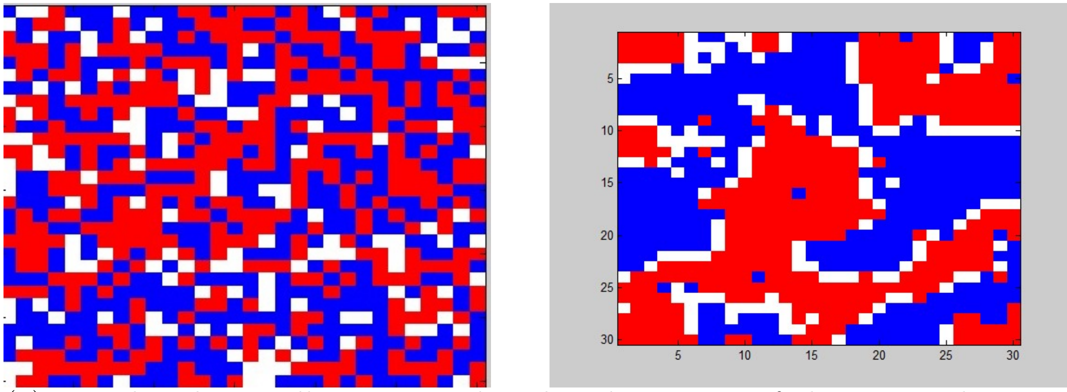
TABLE 1.2: Time taken to segregate

- In presence of activated switching agents after half time, system behaves same as deactivated case until half time and after that segregates to $x_\infty = 0.37$.

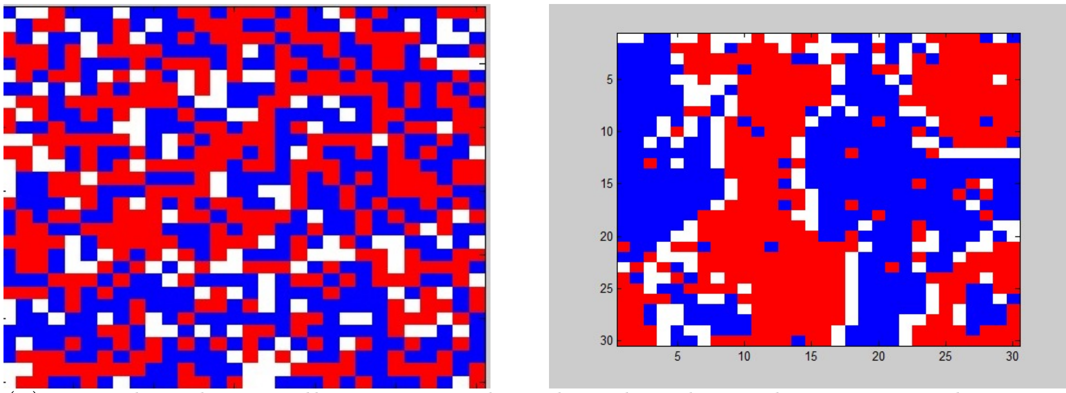
The sample initial lattice and obtained final segregated lattice for the 4 scenarios are depicted below:



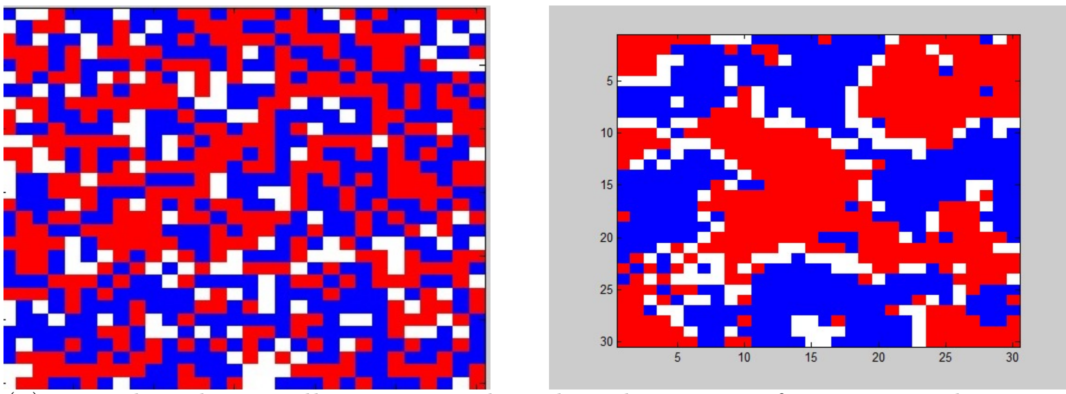
(1) Initial and completely segregated grid in the absence of switching agents.



(2) Initial and partially segregated grid with mixture of clusters in the presence of deactivated switching agents.



(3) Initial and partially segregated grid with enhanced mixity in the presence of activated switching agents.



(4) Initial and partially segregated grid with mixity of agents in the presence of activated switching agents after half time.

Hence, it can be concluded that the introduction of switching agnets in the regular Schelling system leads to a higher contact density and hence mixity during the segregated stage.

1.3 Transition diagrams for different fractions of switching agents

For three different fractions of activated switching agents, namely f = 0 (No switching agents), 0.25 and 0.5, the contact density at the segregated stage is plotted against tolerance values for the system in Fig.1.2, 1.3, 1.3.

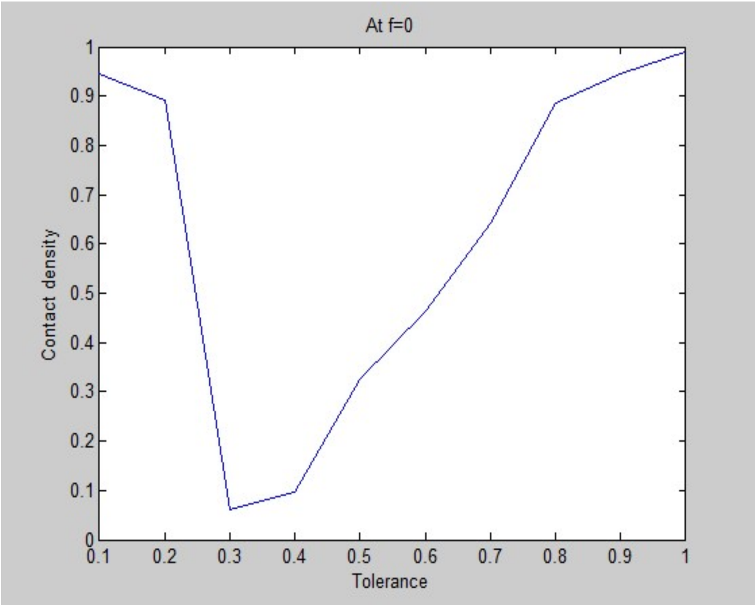


FIGURE 1.2: Contact density for f =0

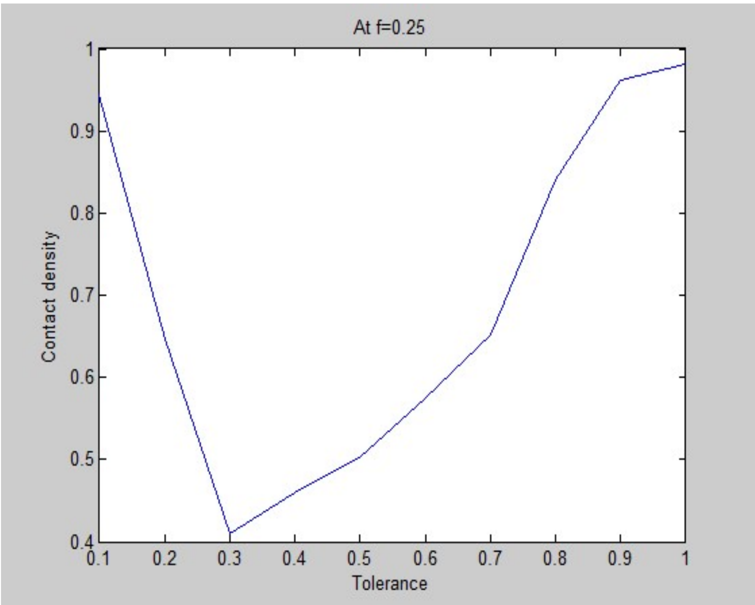


FIGURE 1.3: Contact density for f =0.25

It is observed that the system translates from frozen mixed stage to completely segregated stage at tolerance level 0.25 and again translates from segregated stage to mixed environment at tolerance level 0.75 concluding that system jumps from a stage to another at tolerance values 0.25 and 0.75.

Also, the transition at 0.25 is clearly more abrupt as compared to the one at 0.75.

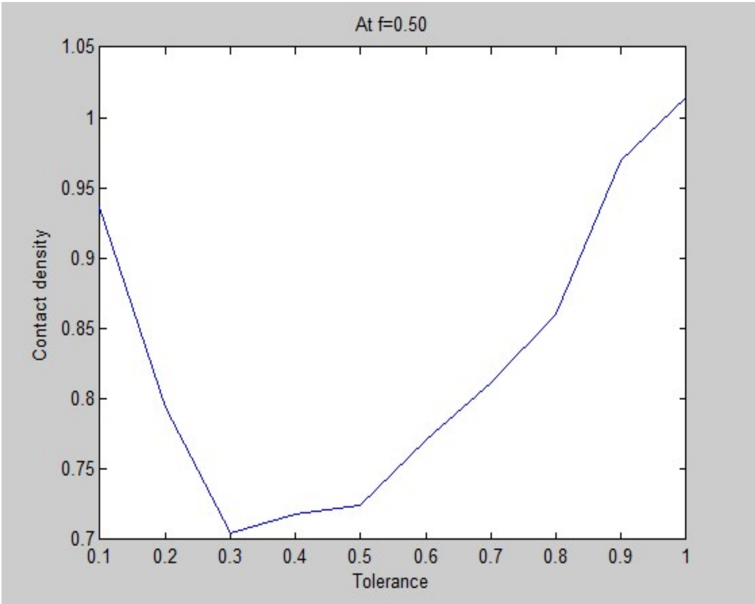


Figure 1.4: Contact density for f =0.50

It is again observed by comparing the contact density at the segregation phase in the 3 plots that as the fraction of activated switching agents increase in the system, it moves towards mixity.

Next, 3D plot of contact density against occupation density of the system and tolerance for f = 0, 0.25, 0.5 is recreated in Fig. 1.5, 1.6 and 1.7 with MATLAB code given in the appendix.

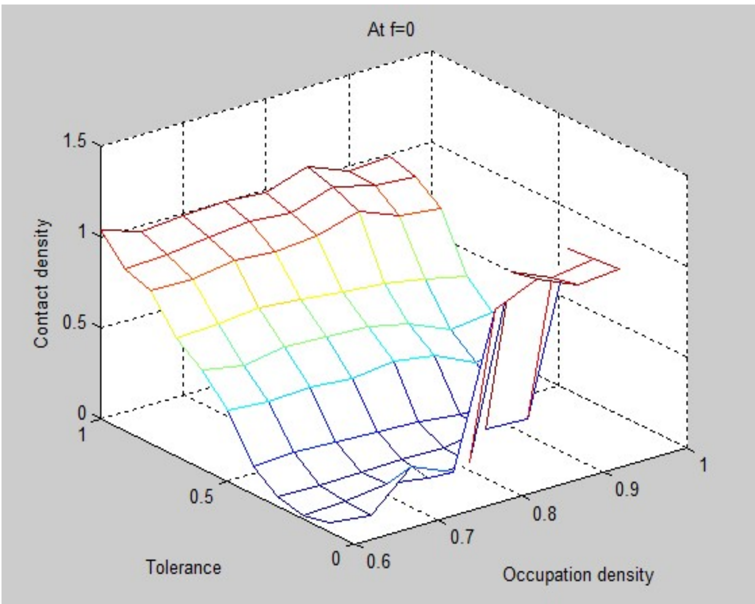


Figure 1.5: Contact density against tolerance and occupation density for f =0

The peaks at tolerance level 0.25 and 0.75 again confirm that the system experiences transition at these values.

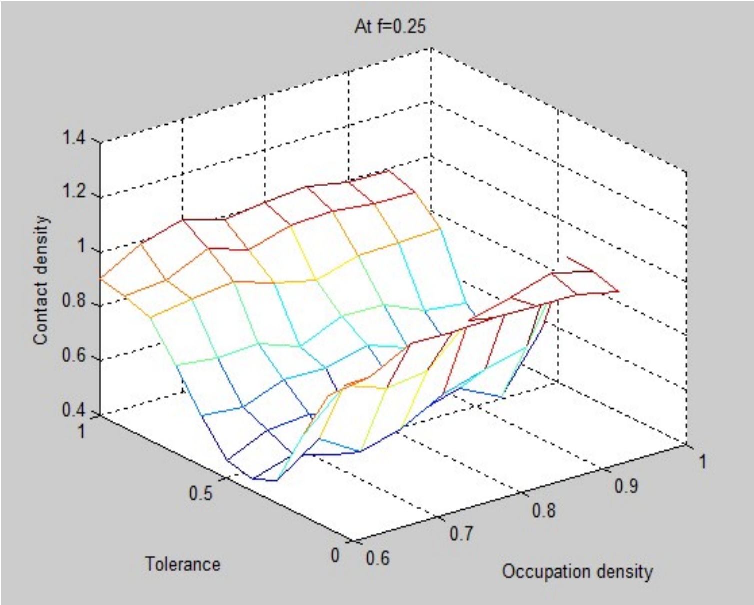


FIGURE 1.6: Contact density against tolerance and occupation density for f =0.25

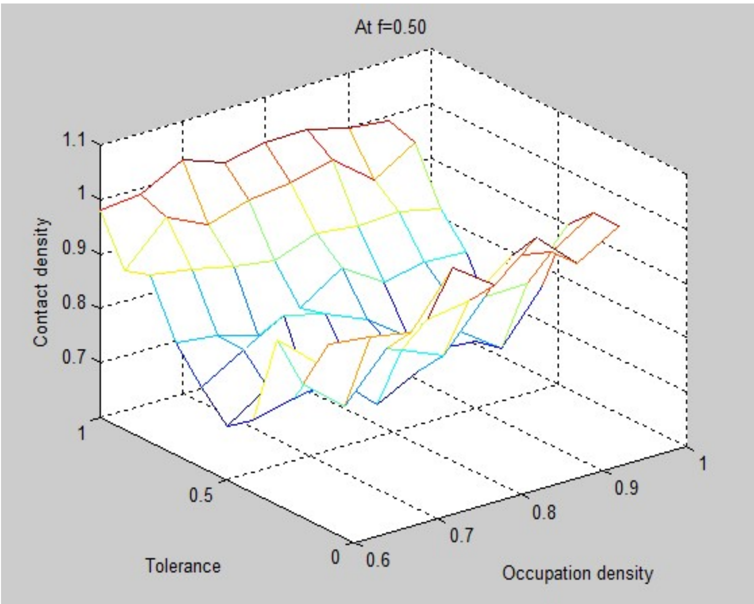


FIGURE 1.7: Contact density against tolerance and occupation density for f =0.5

Next, a 3D plot of contact density against tolerance and the fraction of switching agents in the system (Fig. 1.8) is reproduced, MATLAB code is given in Appendix 4.1.8.

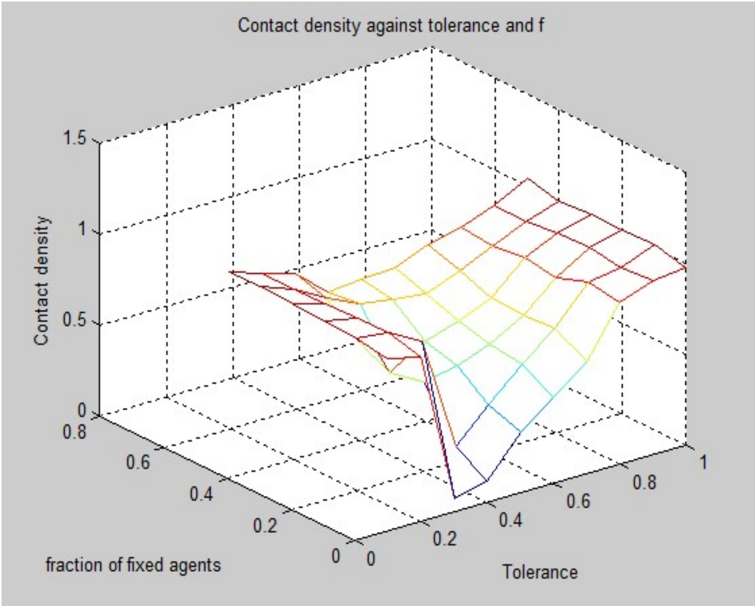


Figure 1.8: Contact density against tolerance and f

In a plot of contact density against f and tolerance, it is observed that the mixity is more and more visible as f increases. As f increases, the mixity is attained at a lower level of tolerance itself.

For a more clear view, we reproduce the plot of contact density at the segregated stage against different fractions of switching agents with constant occupation density 0.85 and tolerance level 0.3 in Fig. 1.9, (MATLAB code is given in Appendix 4.1.9).

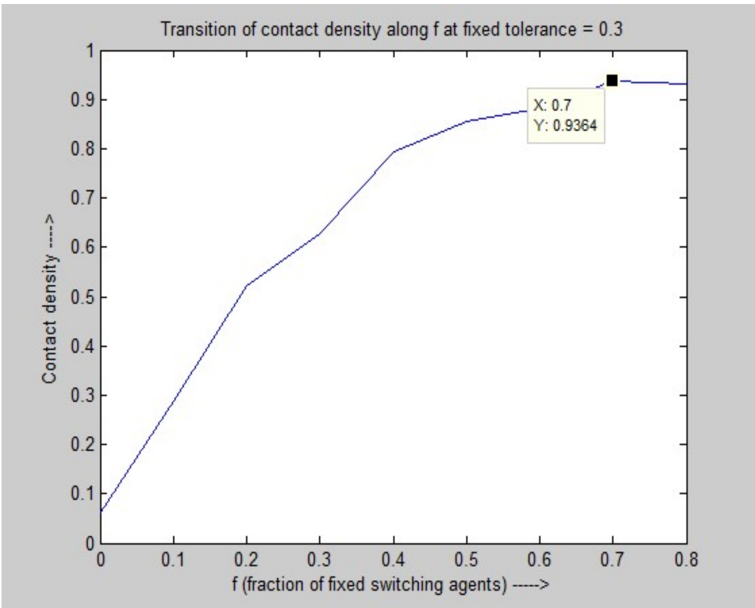


Figure 1.9: Contact density for various f

It is observed that for these values of fixed parameters, as the fraction of switchong agents in the

system increase, the contact density also increases and hence the system posseses mixity rather than reaching a full segregation.
Also notice that the complete mixity of the system is attained when fraction of fixed agents is 0.7 and after that the contact density remains constant, i.e. better miixty is attained by tuning f to an intermediate value rather than setting it as 1.
That is, a combinaton of Schelling dynamics and switching dynamics bring more contact in the system than pure Schelling or pure switching dynamics.

Chapter 2

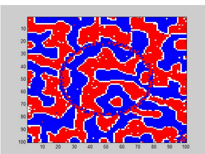
Schelling model with presence of fixed switching agents on a determined annular ring

2.1 Introduction

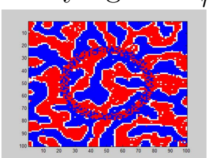
In this chapter, a variant of Schelling segregation model with fixed agents is introduced where the fixed switching agents are placed on an annular ring of inner radius ‘r’ and outer radius ‘r+δ’. We consider the cases where all sites inside the ring are occupied by deactivated fixed agents, activated fixed agents, where only a fraction of sites inside the ring is fixed agents and the rest being an agent or a vacancy with uniform probability. Also a case where the fixed agents are located on the outer part of a circle of radius ‘r’ ‘is explored. Then the Schelling dynamics are run to the randomnly mixed grid so obtained.
The objective of this chapter is to observe the affects of such an allocation of the fixed agents on the contact density of the system.

2.2 Model and observations

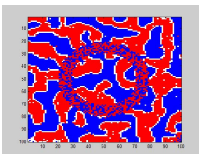
The system for reference is a 100 X 100 grid and we aim to analyse the effect of placing the fixed agents (deactivated and activated) in a fixed annular ring on the grid determined by radius ‘r’ and its width is given by parameter ‘δ’. Also, the plots when the fixed agents were placed on the outer of the circle with radius ‘r’ are given.
For illustration, we use N = 100, $\frac{r}{N} = 0.3$, tolerance = 0.3 and we are placing the fixed agents on the entire annulus.
So, the segregated plots when the deactivated agents are placed for differnt widths of the annulus, i.e. varying $\frac{\delta}{r}$ are given in (1) and the contact density for each of them is noted and hence the plot of contact density against $\frac{\delta}{r}$ is shown in Fig. 2.1.



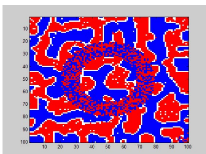
$\frac{\delta}{r} = 0.1$



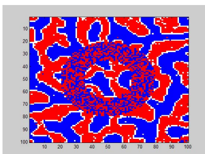
$\frac{\delta}{r} = 0.2$



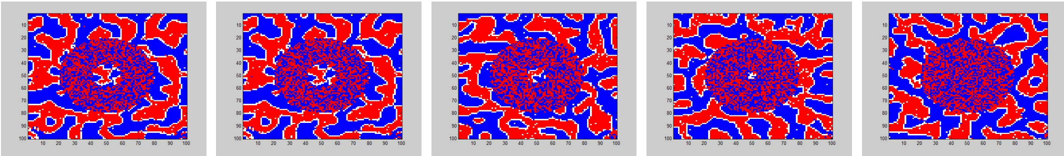
$\frac{\delta}{r} = 0.3$



$\frac{\delta}{r} = 0.4$



$\frac{\delta}{r} = 0.5$



$\frac{\delta}{r} = 0.6$ $\frac{\delta}{r} = 0.7$ $\frac{\delta}{r} = 0.8$ $\frac{\delta}{r} = 0.9$ $\frac{\delta}{r} = 1$

(1) Segregated stages for different widths of annulus where deactivated switching agents are allocated for fixed radius r = 0.3N

Next, the plot of contact density against each of the $\frac{\delta}{r}$ is shown (Fig. 2.1) for deactivated switching agents placed inside annulus of radius r = 0.3N, MATLAB code is given in Appendix 4.2.

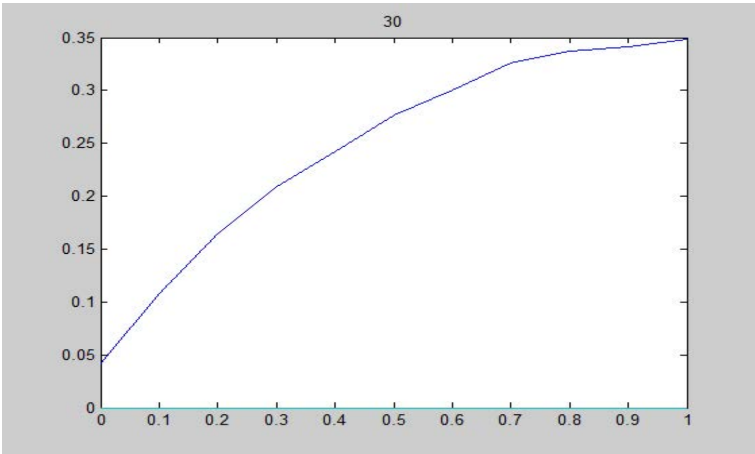
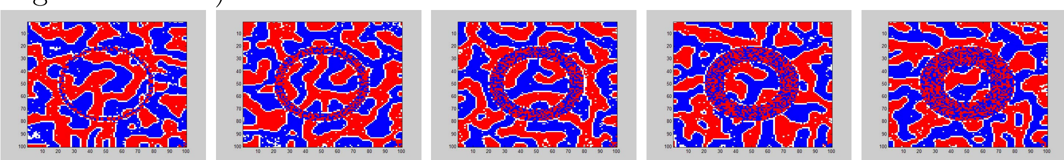
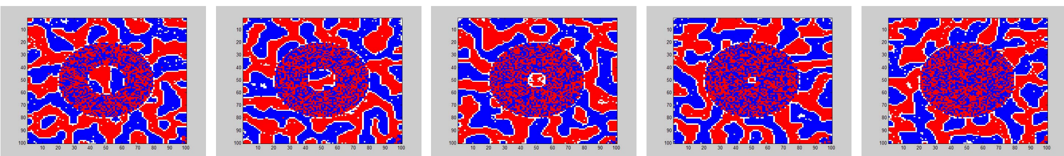


Figure 2.1: Contact density against $\frac{\delta}{r}$ for deactivated switching agents placed inside annulus of radius r = 0.3N

Next, a similar analysis for activated switching agents is given in (2) and Fig. 2.2 (MATLAB code is given in 4.2.4).



$\frac{\delta}{r} = 0.1$ $\frac{\delta}{r} = 0.2$ $\frac{\delta}{r} = 0.3$ $\frac{\delta}{r} = 0.4$ $\frac{\delta}{r} = 0.5$



$\frac{\delta}{r} = 0.6$ $\frac{\delta}{r} = 0.7$ $\frac{\delta}{r} = 0.8$ $\frac{\delta}{r} = 0.9$ $\frac{\delta}{r} = 1$

(2) Segregated stages for different widths of annulus where activated switching agents are allocated for fixed radius r = 0.3N

Next, the plot of contact density against each of the $\frac{\delta}{r}$ is shown (Fig. 2.2) for activated switching agents placed inside annulus of radius r = 0.3N, MATLAB code is given in Appendix 4.2.

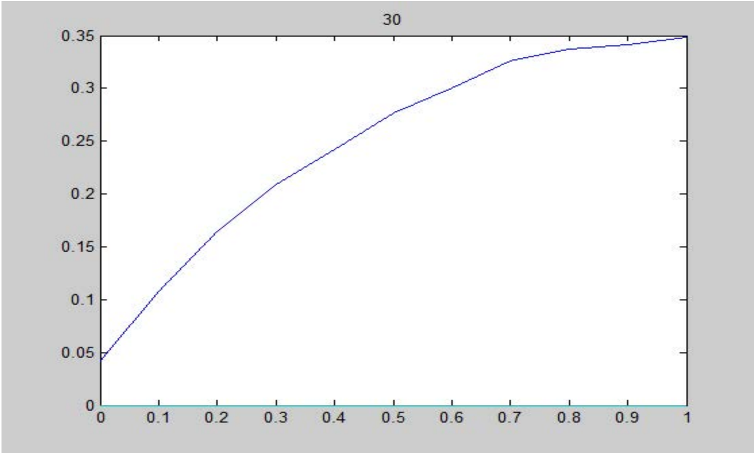
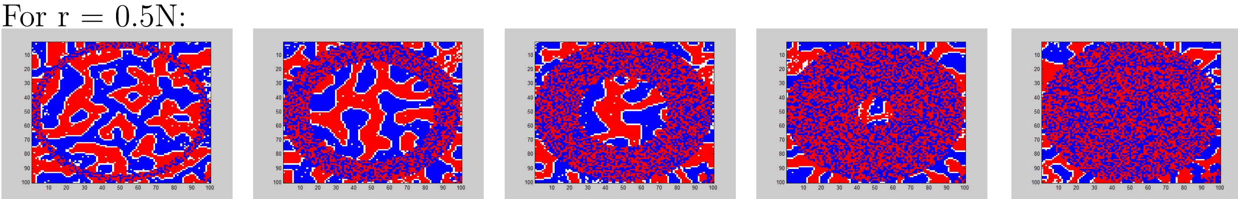


Figure 2.2: Contact density against $\frac{\delta}{r}$ for activated switching agents placed inside annulus of radius r = 0.3N

For r = 0.5N:



$\frac{\delta}{r} = 0.1$ $\frac{\delta}{r} = 0.3$ $\frac{\delta}{r} = 0.5$ $\frac{\delta}{r} = 0.7$ $\frac{\delta}{r} = 0.9$

(1) Segregated stages for different widths of annulus where deactivated switching agents are allocated for fixed radius r = 0.5N

Next, the plot of contact density against each of the $\frac{\delta}{r}$ is shown (Fig. 2.3) for deactivated switching agents placed inside annulus of radius r = 0.5N.

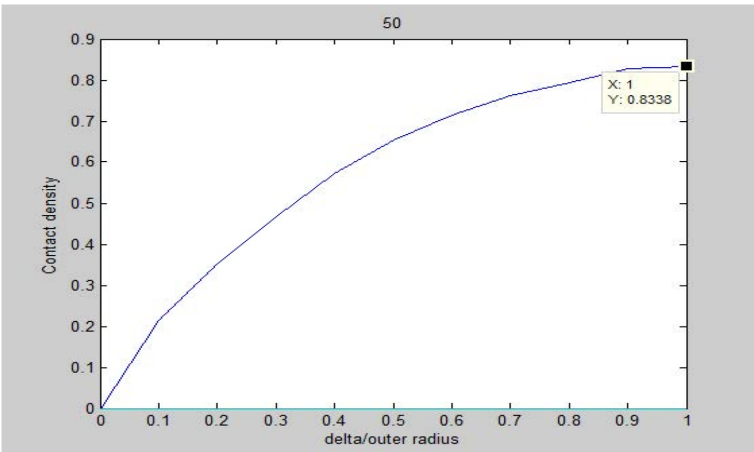
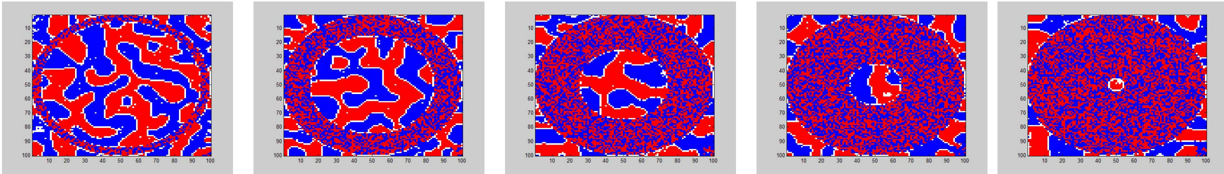


Figure 2.3: Contact density against $\frac{\delta}{r}$ for deactivated switching agents placed inside annulus of radius r = 0.5N



$\frac{\delta}{r} = 0.1$ $\frac{\delta}{r} = 0.3$ $\frac{\delta}{r} = 0.5$ $\frac{\delta}{r} = 0.7$ $\frac{\delta}{r} = 0.9$

(2) Segregated stages for different widths of annulus where activated switching agents are allocated for fixed radius r = 0.5N

Next, the plot of contact density against each of the $\frac{\delta}{r}$ is shown (Fig. 2.4) for activated switching agents placed inside annulus of radius r = 0.5N.

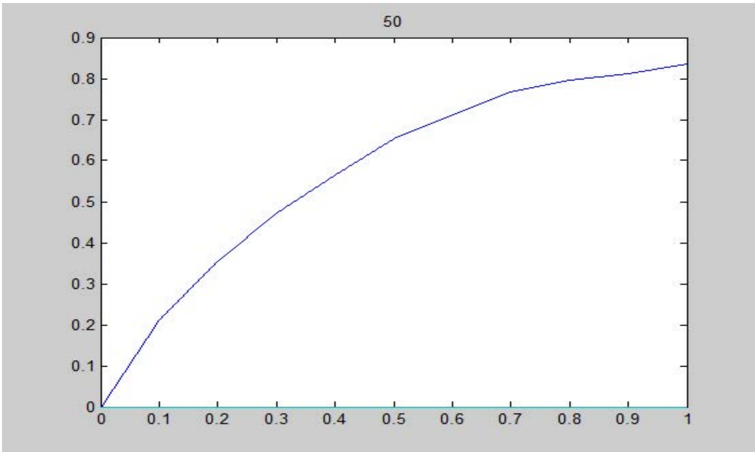
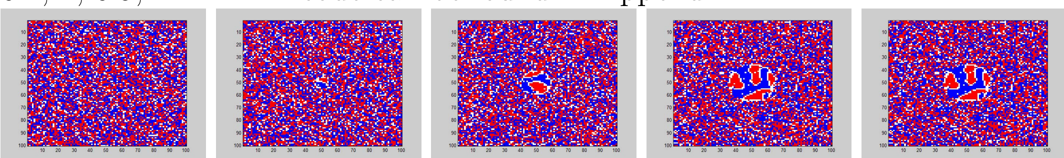
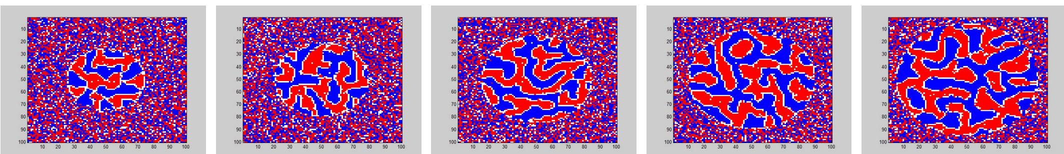


FIGURE 2.4: Contact density against $\frac{\delta}{r}$ for activated switching agents placed inside annulus of radius r = 0.5N

Now, we discuss the scenario where fixed agents are placed on an outer of the circle with different raddii.
And the segregated stages for the activated agents are given in (1) for $\frac{r}{N} = 0$ (all fixed agents), 0.1,..., 0.9, MATLAB code can be found in Appendix 4.2.



$\frac{r}{N} = 0$ $\frac{r}{N} = 0.1$ $\frac{r}{N} = 0.2$ $\frac{r}{N}$ =0.3 $\frac{r}{N} = 0.4$



$\frac{r}{N} = 0.5$ $\frac{r}{N} = 0.6$ $\frac{r}{N} = 0.7$ $\frac{r}{N} = 0.8$ $\frac{r}{N} = 0.9$

(1) Segregated stages for different cases where activated switching agents are allocated outside the circle of radii $\frac{r}{N} = 0$ (no fixed agents), 0.1,..., 0.9.

Next, the plot of contact density against each of the $\frac{r}{N}$ for activated switching agents placed outside the cicle of radius 'r' (Fig. 2.5), MATLAB code is given in Appendix 4.2.

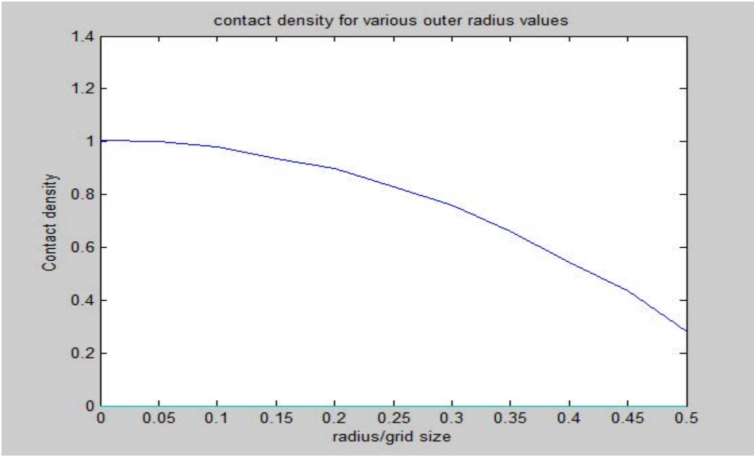
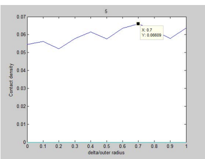
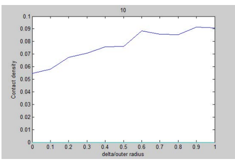


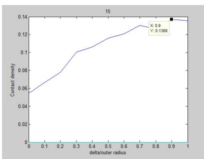
Figure 2.5: Contact density against $\frac{r}{N}$ for activated switching agents placed outside the cicle of radius 'r'

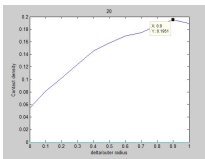
2.3 Transition diagrams for different sizes of annulus consisting of switching agents

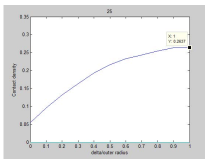
Now, the plot of contact density against $\frac{\delta}{r}$ is shown for deactivated switching agents placed inside annulus for various values of $\frac{r}{N}$.

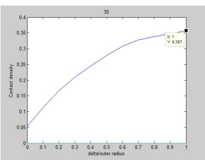

$\frac{r}{N} = 0.05$

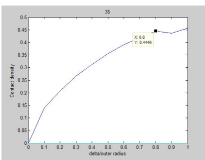

$\frac{r}{N} = 0.1$

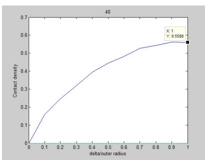

$\frac{r}{N} = 0.15$

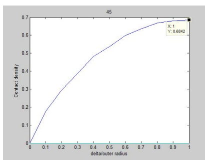

$\frac{r}{N}$=0.2

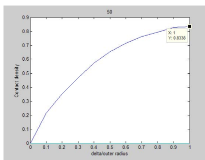

$\frac{r}{N}$=0.25


$\frac{r}{N} = 0.3$

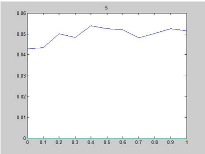

$\frac{r}{N} = 0.35$


$\frac{r}{N} = 0.4$

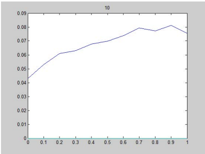

$\frac{r}{N}$=0.45


$\frac{r}{N}$=0.5

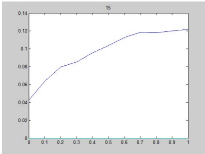
Next, plot of contact density against $\frac{\delta}{r}$ is shown for activated switching agents placed inside annulus for various values of $\frac{r}{N}$.



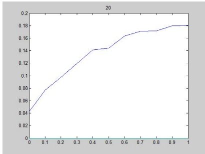
$\frac{r}{N} = 0.05$



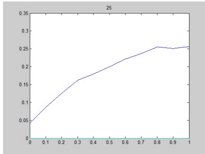
$\frac{r}{N} = 0.1$



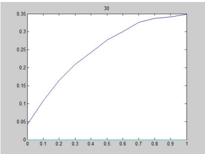
$\frac{r}{N} = 0.15$



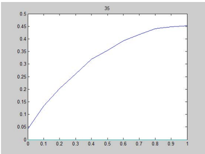
$\frac{r}{N}$=0.2



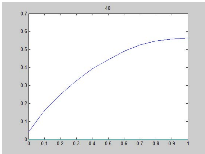
$\frac{r}{N}$=0.25



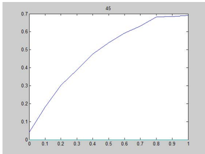
$\frac{r}{N} = 0.3$



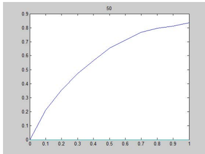
$\frac{r}{N} = 0.35$



$\frac{r}{N} = 0.4$



$\frac{r}{N}$=0.45



$\frac{r}{N}$=0.5

Now, for a better observation, we have a 3D plot of contact density versus $\frac{r}{N}$ and $\frac{\delta}{r}$ for deactivated and activated agents (Fig. 2.6, 2.7).

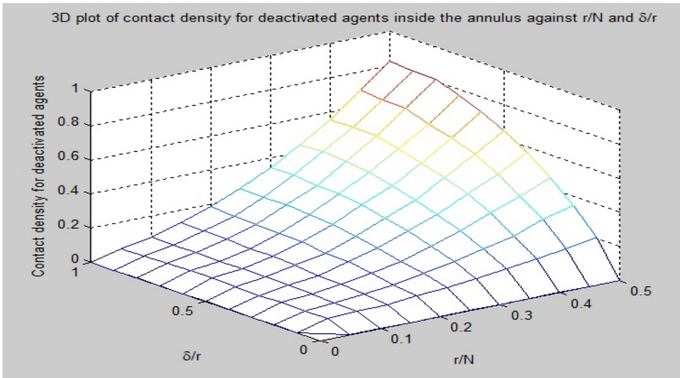


FIGURE 2.6: Contact density against $\frac{r}{N}$ and $\frac{\delta}{r}$ for deactivated switching agents placed in the annular ring

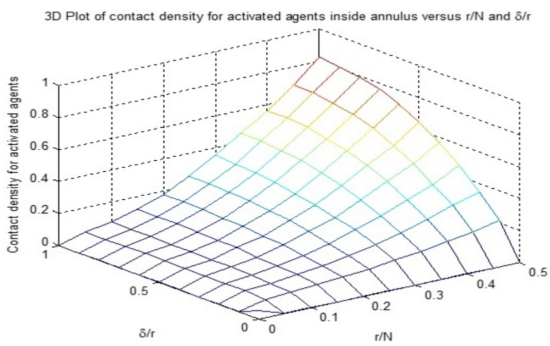


FIGURE 2.7: Contact density against $\frac{r}{N}$ and $\frac{\delta}{r}$ for activated switching agents placed in the annular ring

Next,we have a 3D plot of contact density versus tolerance and $\frac{\delta}{r}$ for deactivated and activated agents (Fig. 2.8, 2.9).

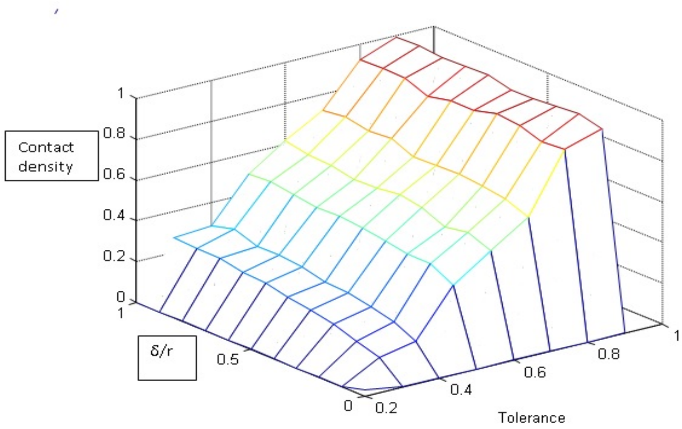


FIGURE 2.8: Contact density against tolerance and $\frac{\delta}{r}$ for deactivated switching agents placed in the annular ring

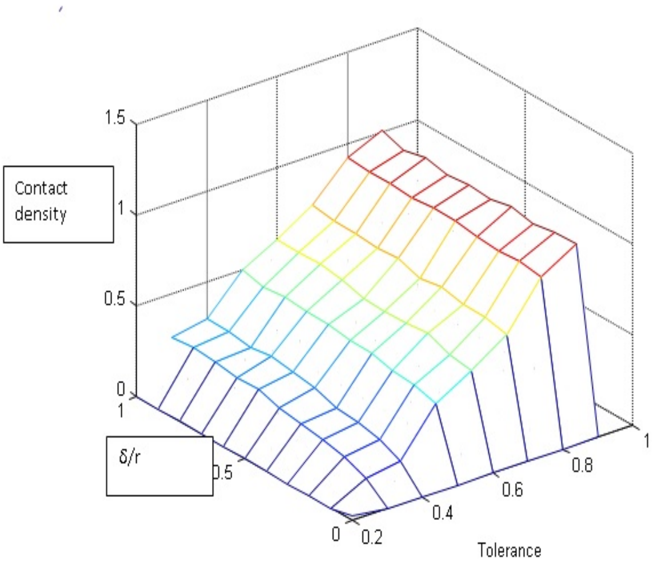
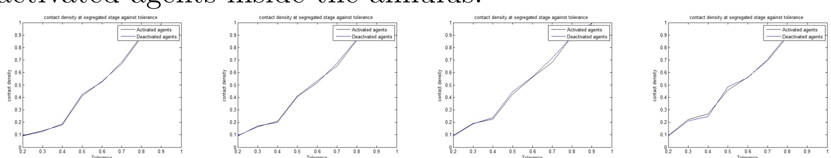
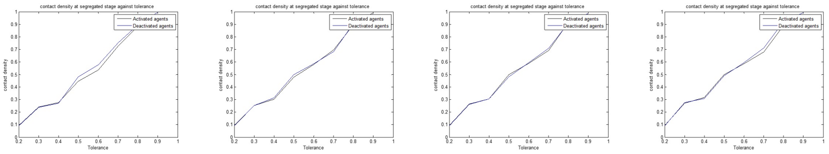


FIGURE 2.9: Contact density against tolerance and $\frac{\delta}{r}$ for activated switching agents placed in the annular ring

Next, to observe at what tolerance level mixity is first attained for different widths of the annulus, we plot tolerance versus contact density for fixed $\frac{r}{N} = 0.25$ and various $\frac{\delta}{r}$ for deactivated and activated agents inside the annulus.



$\frac{\delta}{r} = 0.1$ $\frac{\delta}{r} = 0.2$ $\frac{\delta}{r} = 0.3$ $\frac{\delta}{r} = 0.4$



$\frac{\delta}{r} = 0.5$ $\frac{\delta}{r} = 0.6$ $\frac{\delta}{r} = 0.7$ $\frac{\delta}{r} = 0.8$

(1) Tolerance versus contact density for fixed $\frac{r}{N} = 0.25$ and various $\frac{\delta}{r}$ for deactivated and activated agents inside the annulus.

Next, we have a 3D plot of contact density versus tolerance and occupation density for deactivated (Fig. 10)and activated (Fig. 11) agents distributed on an annular ring.

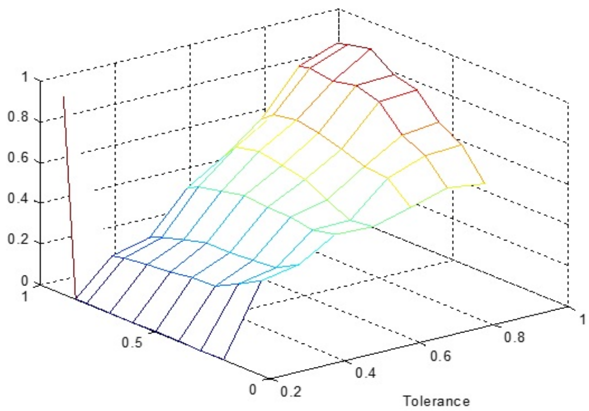


FIGURE 2.10: Contact density against tolerance and occupation density for deactivated switching agents placed in the annular ring

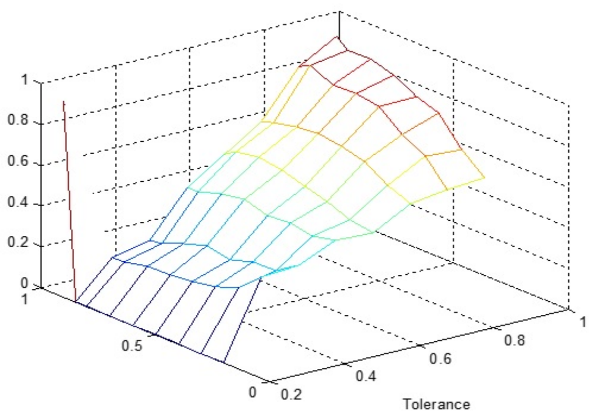


FIGURE 2.11: Contact density against tolerance and occupation density for activated switching agents placed in the annular ring

Chapter 3

Schelling model with presence of fixed switching agents on a gaussian ring

3.1 Introduction

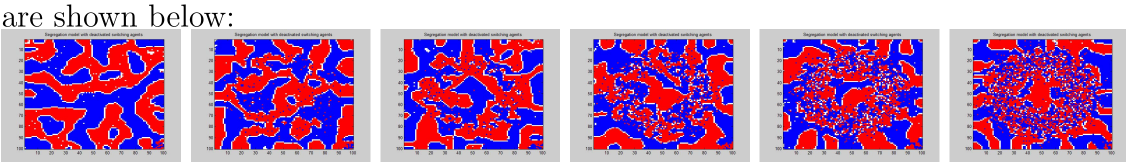
In this chapter, a variant of Schelling segregation model is introduced where a fraction 'f' of occupied sites out of total N^2 sites of the system are fixed switching agents. The sites where these fixed agents are located (d$\cos\theta$, d$\sin\theta$) are determined by drawing a number 'd' using gaussian distribution with mean 'r' and standard deviation 'δ' and an angle 'θ' using uniform distribution. Then the Schelling dynamics are run to the randomnly mixed grid so obtained.
The objective of this chapter is to observe that such a gaussian allocation of the fixed agents enhances the contact density of the system and introduces full segregation in the system at a much lower value of tolerance than was observed with regular Schelling dynamics.

3.2 Model and observations

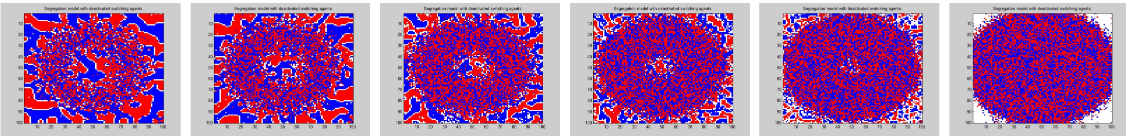
We consider a system which is a N X N grid, where N is chosen as 100 for the experiments with the assumed occupation density of 0.85. We aim to analyse the effect of distributing the deactivated and activated fixed agents in a gaussian ring.
We assume fraction 'f' of occupied sites out of total 100 X 100 sites i.e. 0.85 X 100 X 100 X f as the total no. of fixed agents in the system. The allocation of these fixed agents is done using gaussian distribution as:
For each of the switching agents, pick a random number 'd' using gaussian distribution with mean 'r' and standard deviation 'δ' and pick an angle 'θ' between 0 to 2π using uniform distribution U[0,2π]. Then the location of the fixed agent will be given by ($\lfloor dcos\theta \rfloor$, $\lfloor dsin\theta \rfloor$), where $\lfloor \rfloor$ denotes the floor function.
With such a gaussian allocation, it is clear that most of the fixed agents will be located in an annular ring with inner radius 'r' and outer radius 'r+δ' but still the annulus will have vacancies and pure agents too because of the random allocation.
Once it is decided which site is occupied by fixed agent, we assume equal probability of the agent to be red or blue.
For rest of the sites, we follow the same Schelling allocation procedure where it is equally probable

for a non vacant site to be of type red or type blue. Then starting from an obtained randomnly mixed grid, the standard Schelling dynamics are run with an appropriate tolerance level and the observations are noted.

Now, for a fixed tolerance level of 0.3, the segregated stages for different fractions of deactivated (Fig. (1)) and activated (Fig. (2)) fixed agents to total no. of occupied sites for a 100 X 100 grid are shown below:

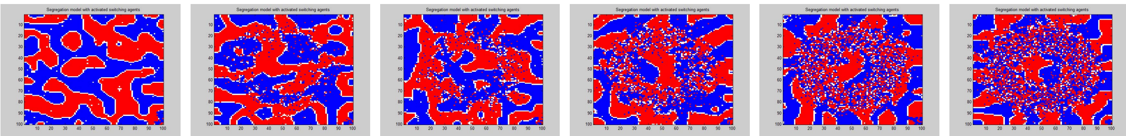


f = 0 f = 0.08 f = 0.1 f = 0.2 f = 0.3 f = 0.4

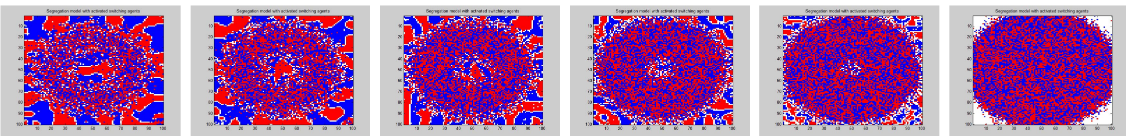


f = 0.5 f = 0.6 f = 0.7 f = 0.8 f = 0.9 f = 1

(1) Segregated stages for different fractions of deactivated switching agents allocated using gaussian distribution with fixed tol.



f = 0 f = 0.08 f = 0.1 f = 0.2 f = 0.3 f = 0.4



f = 0.5 f = 0.6 f = 0.7 f = 0.8 f = 0.9 f = 1

(2) Segregated stages for different fractions of activated switching agents allocated using gaussian distribution with fixed tol.

Observe that for f = 0, i.e. No fixed agents scenario, complete segregation is attained and for f = 1, i.e. all occupied agents being fixed agents where system consists only of fixed agents on the gaussian ring and vacancies on the rest of the sites, complete mixity is attained since the only change happening to the initial grid is switch of agents.
As the value of f increases, it is observed that the cluster size in the segregated stage keeps on decreasing and the system moves towards mixity.

Next, a plot summarizing the above scenario, of contact density at segregated stage versus fraction of fixed agents for both deactivated and activated agents for fixed tol. = 0.3 is given in Fig. 3.1.

Clearly, the mixity of the system increases if the switching agents are activated as compared to

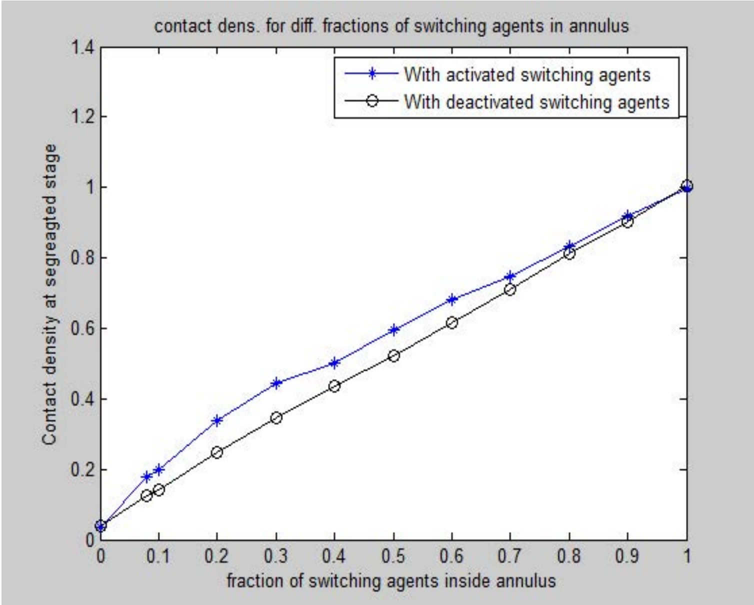


Figure 3.1:Contact density versus fraction of deactivated and activated switching agents

when the switching agents are deactivated.
Observe that the contact density is 1 when f = 1 (all the occupied sites are occupied by fixed agents), 0.01 when no agents are present (f=0) and the system moves towards mixity as f increases.

Next, we observe the effect of tolerance change in the system.
For a fixed fraction of 0.5 to total no. of occupied sites as fixed agents, segregated stages (Deactivated agents in Fig. (1) and activated agents in Fig. (2)) for different values of tolerances is shown:
Before the plots, for the reference initial grid is shown in Fig. 3.2 where the fixed agents are shown in green.

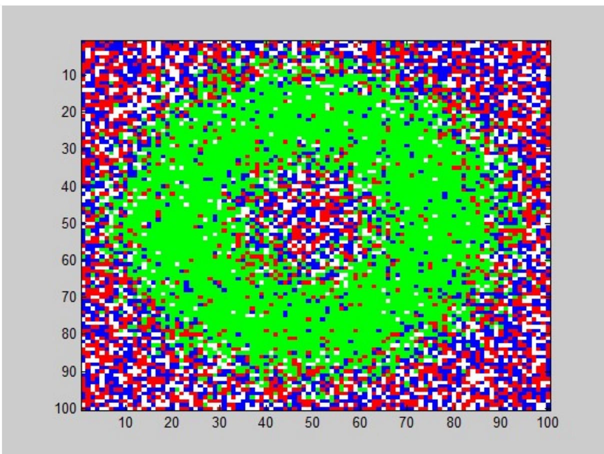
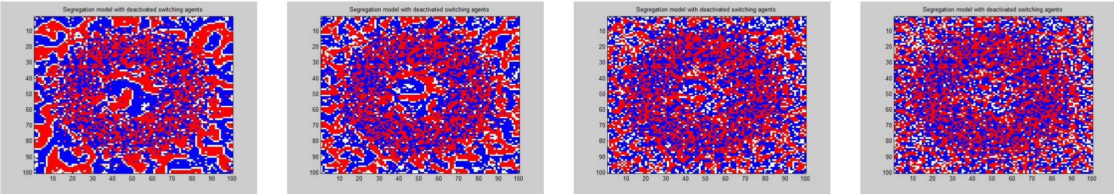
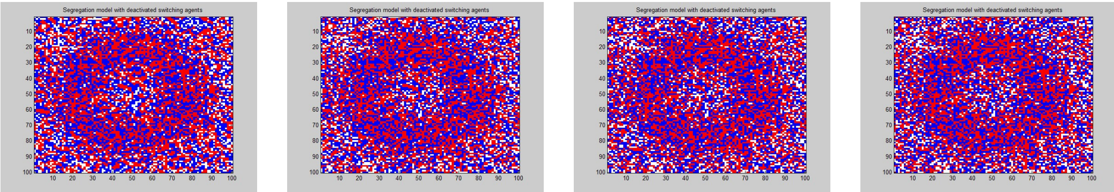


Figure 3.2: Initial grid where r = 0.3N, δ = 0.25r, f =0.5 and fixed agents are shown in green

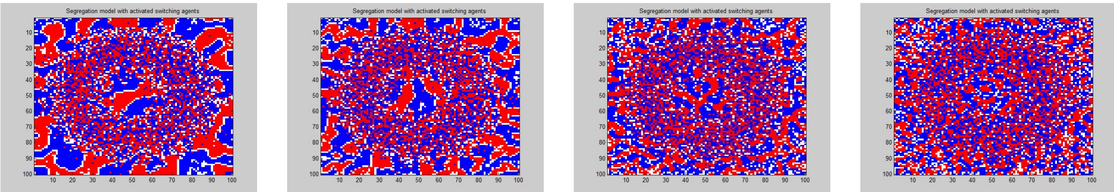


tol = 0.3 tol=0.4 tol=0.5 tol=0.6

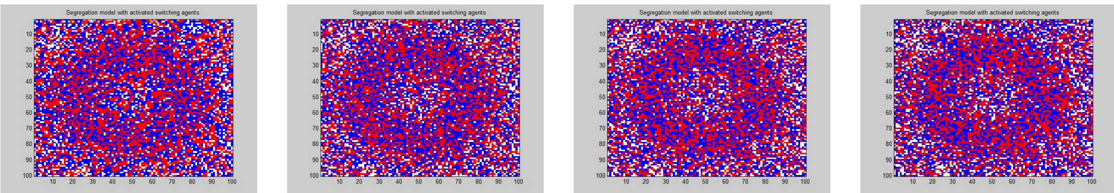


tol = 0.7 tol=0.8 tol=0.9 tol=1

(1) Segregated stages for various tolerance levels with presence of deactivated switching agents on a gaussian ring for f = 0.5.



tol = 0.3 tol=0.4 tol=0.5 tol=0.6



tol = 0.7 tol=0.8 tol=0.9 tol=1

(2) Segregated stages for various tolerance levels with presence of activated switching agents on a gaussian ring for f = 0.5.

It is quite clear that the segregated pattern with fixed agents randomnly distributed on the gaussian ring is seen for tolerance level = 0.3 and 0.4, and the cluster size keeps on reducing as the tolerance increases. Also, in case of f = 0.5, complete mixity is seen at tol = 0.5 and above, which is quite low when compared with the standard schelling case.

Hence concluding that the addition of fixed agents on the gaussian ring produce mixity at a much lower tolerance level in the system.

Now the plot of contact density at segregated stage versus tolerance for deactivated and activated switching agents is given in Fig. 3.3 for a fraction f =0.5 of occupied sites as fixed, MATLAB code is given in Appendix 4.3.

The system translates from frozen mixed stage to dilutely segregated and again to mixed for

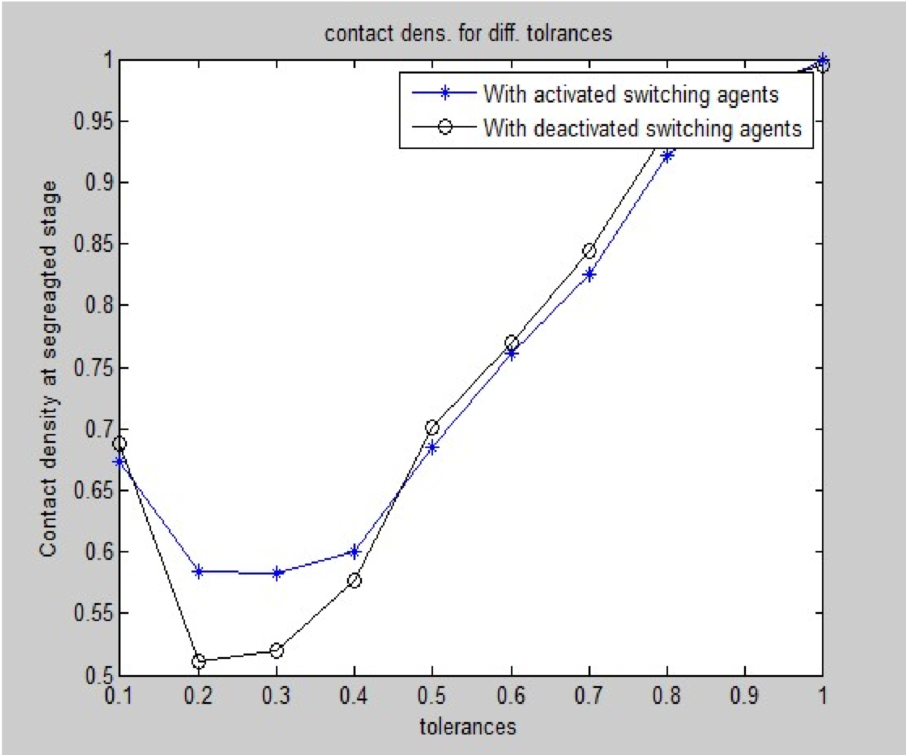
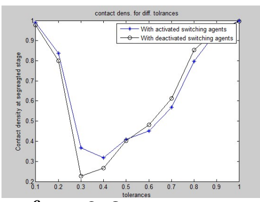


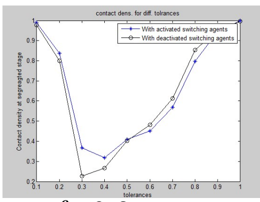
FIGURE 3.3: Contact density at segregated stage versus tolerance for deactivated and activated switching agents.

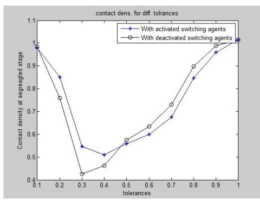
higher tolerance levels and observe that complete mixity is achieved at tol. = 0.5.

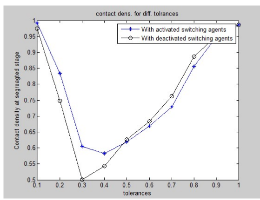
3.3 Transition diagrams for different fractions of switching agents

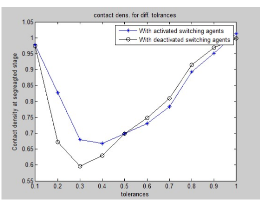
Now, the plots of Contact density versus tolerance for different fractions of deactivated and activated fixed agents to the total no. of occupied agents are given, MATLAB code for the same can be found in Appendix 4.3.

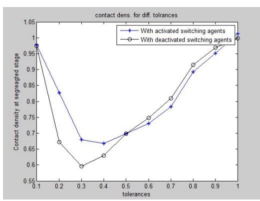

f = 0.2

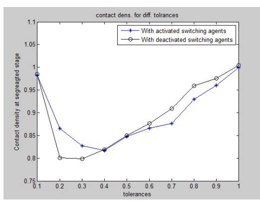

f=0.3

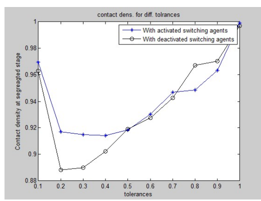

f=0.4


f=0.5


f = 0.6

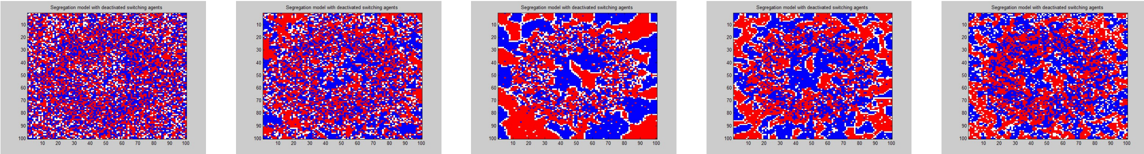

f=0.7


f=0.8

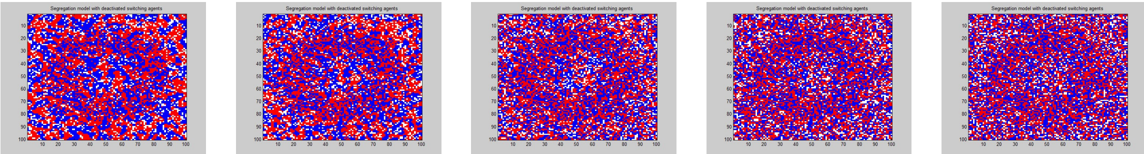

f=0.9

It is clearly observed that the system with fixed agents distributed on a gaussian ring translates

from frozen mixed stage to a dilutely segregated stage and again translates from segregated stage to mixed environment at much lower level of tolerance than the usual Schelling dynamics.
As an illustration, consider when N =100, $\frac{r}{N}$ = 0.3, $\frac{delta}{r}$ = 0.25, f = 0.35, for various values of tolerances, segregated stages are shown for deactivated agents (1) and activated agents (2) and contact density is plotted against tolerance simultaneously for both the cases(Fig. 3.4).

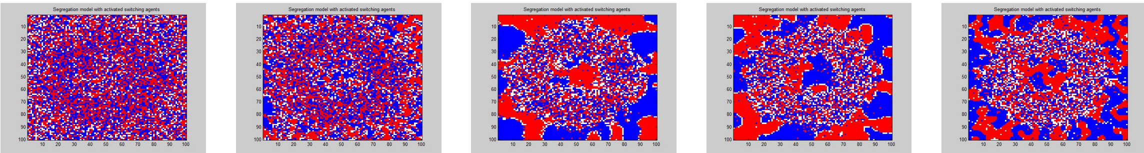


tol = 0.1 tol=0.2 tol=0.3 tol=0.4 tol=0.5

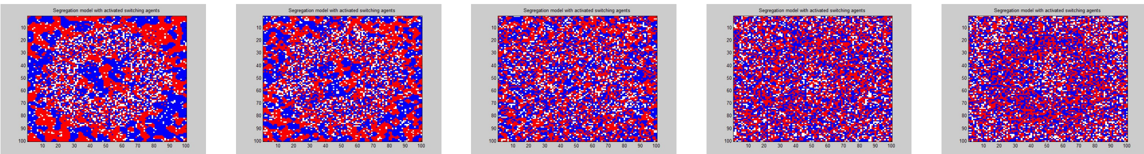


tol = 0.6 tol=0.7 tol=0.8 tol=0.9 tol= 1

(1) Segregated stages for various tolerance levels with presence of deactivated switching agents on a gaussian ring for f = 0.35.



tol = 0.1 tol=0.2 tol=0.3 tol=0.4 tol= 0.5



tol = 0.6 tol=0.7 tol=0.8 tol=0.9 tol= 1

(2) Segregated stages for various tolerance levels with presence of activated switching agents on a gaussian ring for f = 0.35.

Observe from the graphs above that the contact density and hence mixity in the system for frozen stage (i.e. before the first transition stage)is more in case of activated agents for all the values of f than in the case of deactivated agents and same is the scenario in the segregated stage but when the system again translates from segregated stage to mixity then in all the graphs, contact density for the deactivated agents is more as compared to the activated ones.
Hence, we can make an important observation here:
The system with fixed agents distributed on a gaussian ring when compared in terms of deactivated and activated agents, has higher contact density and hence mixity for activated agents before the second transition (i.e. from segregated to mixed) and for deactivated agents after the transition.
Also, when compared in terms of 2 transition stages, it is clear that the transition from frozen

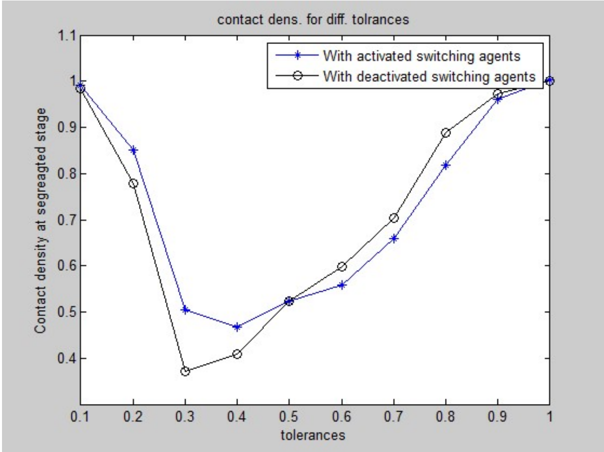
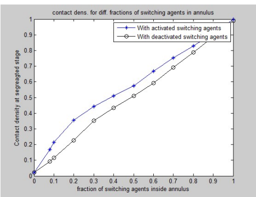
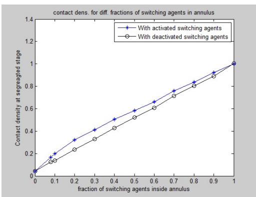


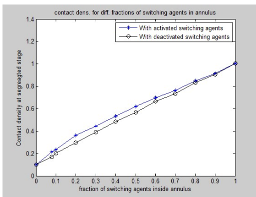
Figure 3.4: Contact density at segregated stage versus fraction of deactivated and activated switching agents.

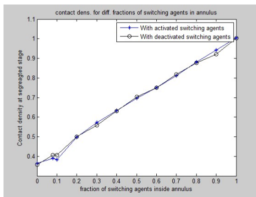
stage to segregated is much more abrupt than from segregated to mixed.
It can be concluded that in case of fixed agents (specially for deactived ones) distributed on a gaussian ring, complete mixity of the system is achieved at a much lower level of tolerance than in the regular Schelling dynamics.

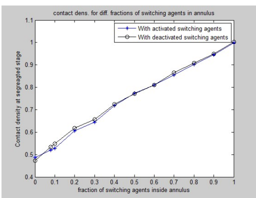
Here, we give the plots of contact density versus fraction of fixed agents (deactivated and activated superimposed in same graph) for various values of tolerances.

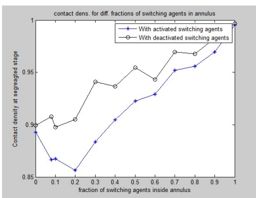

tol. = 0.2

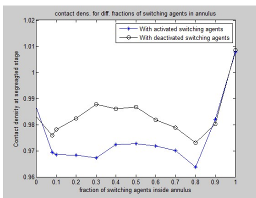

tol. =0.3


tol. =0.4


tol. =0.5


tol. = 0.7


tol. =0.8


tol. =0.9

Next for a better observation, 3D plot of contact density of the system versus fraction of decativated agents to total no. of occupied agents on a distributed on a gaussian ring and tolerance is given. MATLAB code is give in Appendix 4.3.

Again the transition of the system from a frozen mixed stage to segregated stage and again to a mixed scenario for higher tolerance levels is clearly visible in the plot, but the tolerance level after which the system remains mixed is quite subjective to f, as for higher values of f (say 0.75-1) the plot clearly shows that the mixity is attained for a very low level of tolerance (say 0.15-0.2).

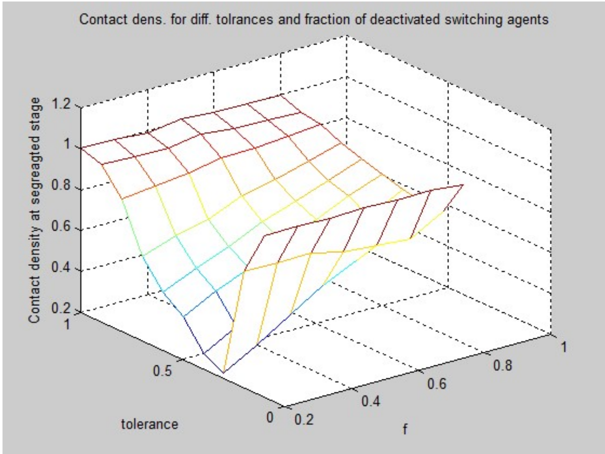


FIGURE 3.5: 3D plot of Contact density at segregated stage versus tolerance and fraction of deactivated switching agents.

Next a similar plot and analysis is done for the case when the fixed agents are activated.

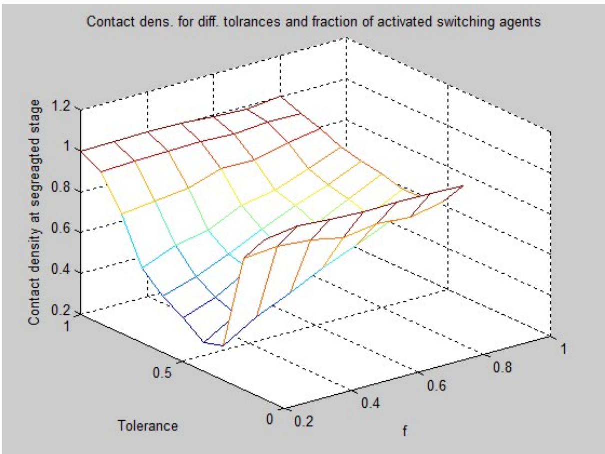


FIGURE 3.6: 3D plot of Contact density at segregated stage versus tolerance and fraction of activated switching agents.

Now, we have the 3d plot of contact density against f and $\frac{\delta}{r}$ for deactivated fixed agents distributed on a gaussian ring (Fig. 3.7).
Clearly the contact density and hence mixity of the grid increases linearly as fraction of agents and width of the gaussian ring increase.
Next, the similar plot is done for activated fixed agents on a gaussian ring (Fig. 3.8).

Next, a 3D plot of contact density versus tolerance and $\frac{\delta}{r}$ for deactivated and activated agents is given .

Next, we give a phase transition diagram of the studied variant of Schelling model.

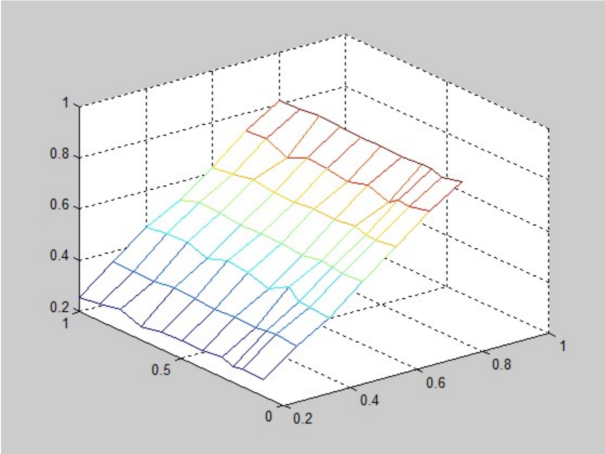


Figure 3.7: 3D plot of Contact density at segregated stage versus fraction of deactivated switching agents and $\frac{\delta}{r}$.

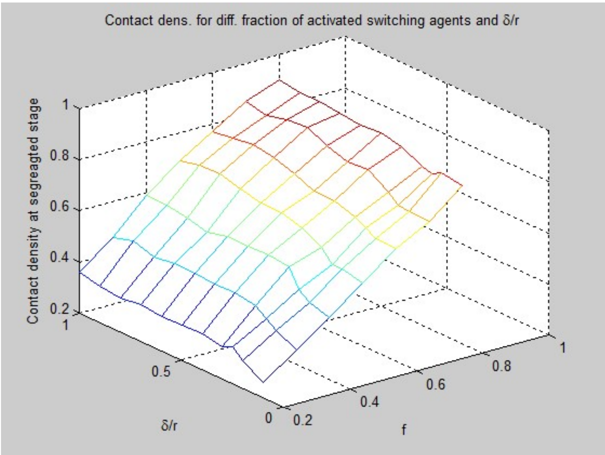


Figure 3.8: 3D plot of Contact density at segregated stage versus fraction of activated switching agents and $\frac{\delta}{r}$.

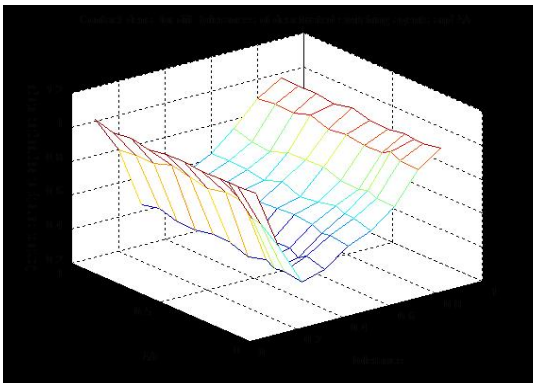


Figure 3.9: 3D plot of Contact density at segregated stage versus tolerance and $\frac{\delta}{r}$ for deactivated agents.

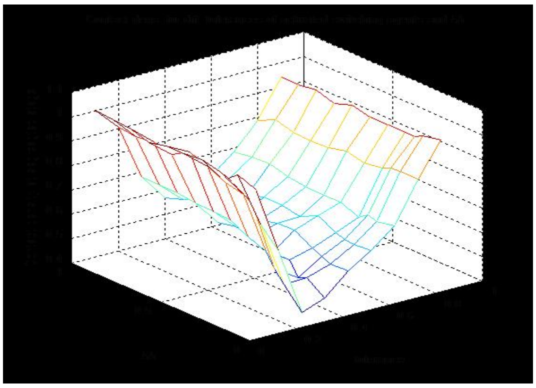


FIGURE 3.10: 3D plot of Contact density at segregated stage versus tolerance and $\frac{\delta}{r}$ for activated agents.

For illustration purpose, we are using N = 100, $\frac{r}{N} = 0.3$, $\frac{\delta}{r} = 0.25$, vacancy density = 0.15, deactivated agents allocated in a gaussian ring and we observe that for different values of f, upto what tolerance, the system is in frozen mixed stage and its transition to segregated stage and again translating from segregated stage to dilutely segregated and finally to completely mixed stage.

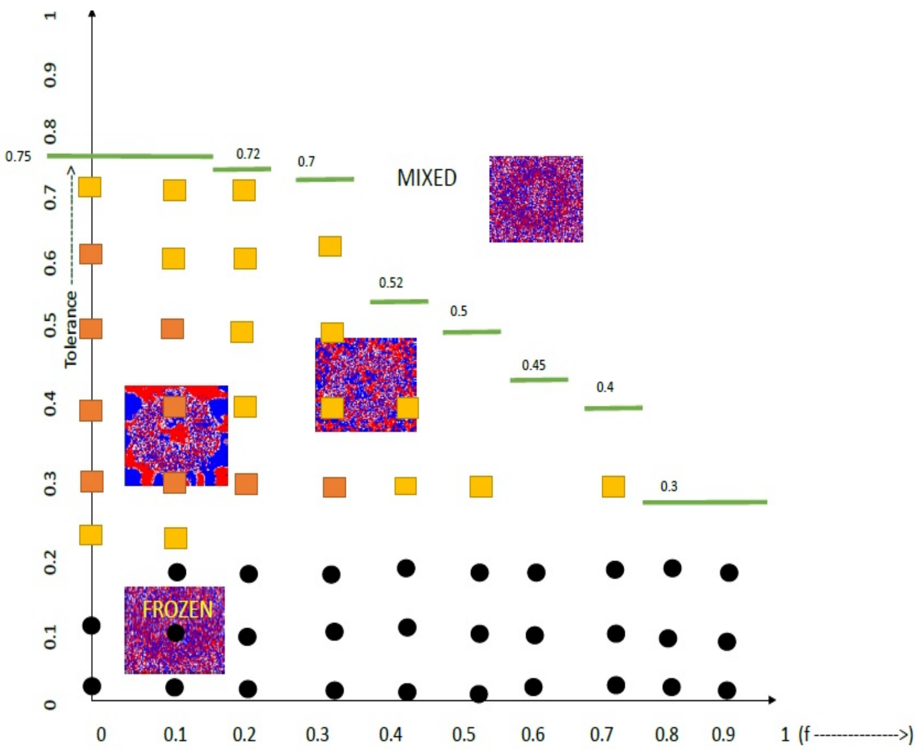


FIGURE 3.11: Phase diagram

Here, the black circles represent frozen stage, yellow rectangles dilutely segregated stage, orange rectangles completely segregated stage and above the green lines, we have complete mixity of the system.
So, the system translates from a frozen mixed stage to segregated and again to mixed. TThe com-

plete mixity is attained at lower levels of tolerance as compared to standard Schelling dynamics.

Now, for $\frac{\delta}{r} = 0.15$:

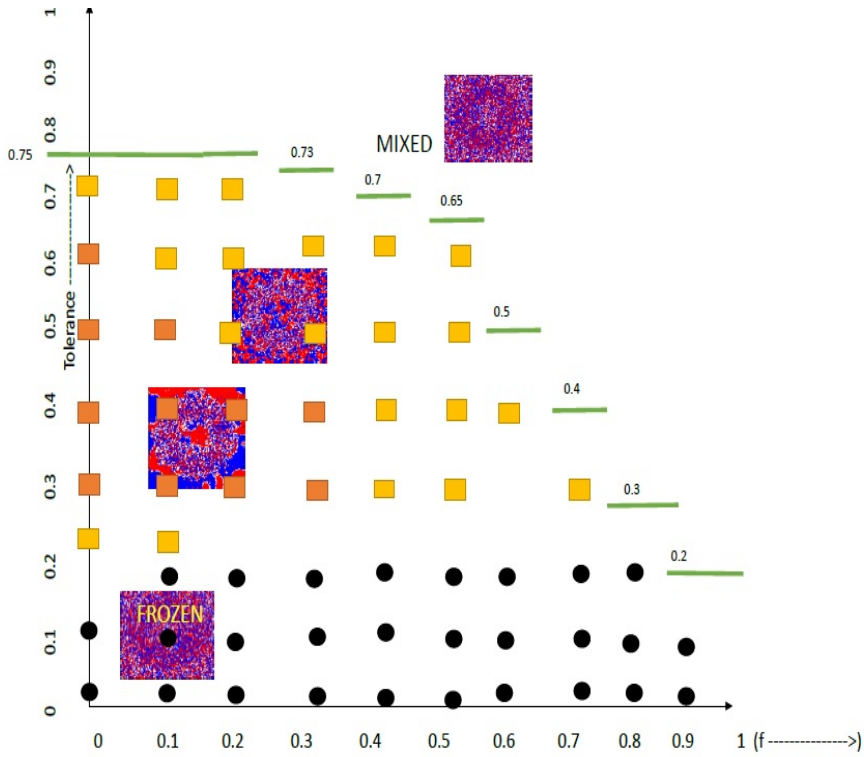


FIGURE 3.12: Phase diagram

The transition from frozen stage to segregated is more abrupt than the transition from segregated to mixed stage.
For activated agents, as compared to deactivated agents, for the same tolerance, we have higher contact density and hence mixity in the grid before the second transition from segregated stage to dilutely segregated and finally to complete mixity but after the transition, scenario completely opposes, i.e. less mixity for the same tolerance as compared to deactivated agents.
Hence, the phase diagram for activated agents when compared to deactivated agents, will have higher level of tolerance upto which the grid is frozen mixed i.e. first transition from frozen to segregated takes place at a higher level of tolerance but again the segregation with deactivated agents is much more as compared to these, i.e. with activated agents, system reaches a dilutely segregated stage at a higher tolerance and hence the second transition from dilutely seg. towards mixeity takes place at a lower level of tolerance than with deactivated agents and again the complete mixity is attained at a higher tolerance (refer transition figures on Pg. 22).

Examining the effect of grid size

Now, we will examine the effect of grid size on the plots.
We plot the conatct density at the segregated stage against the tolerance for 0.5 fraction of deactivated fixed agents (Fig. 3.13) and activated fixed agents (Fig. 3.14)on the gaussian ring for 3

different grid sizes, N = 30, 50 and 100 where r = 0.3N and δ = 0.25r.

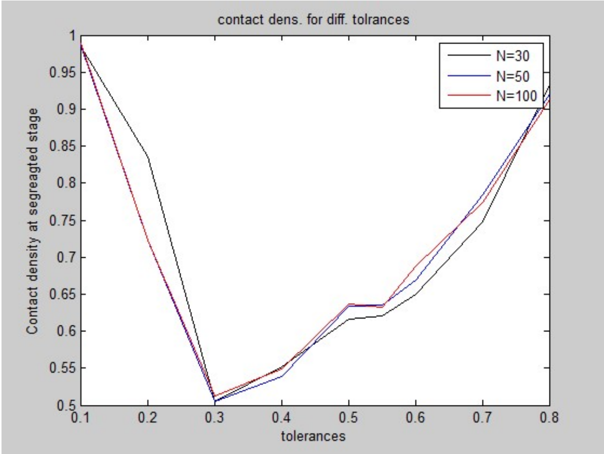


FIGURE 3.13: Plot of contact density versus tolerance for 3 different grid sizes (N = 30, 50, 100) with 0.5 fraction of deactivated fixed agents.

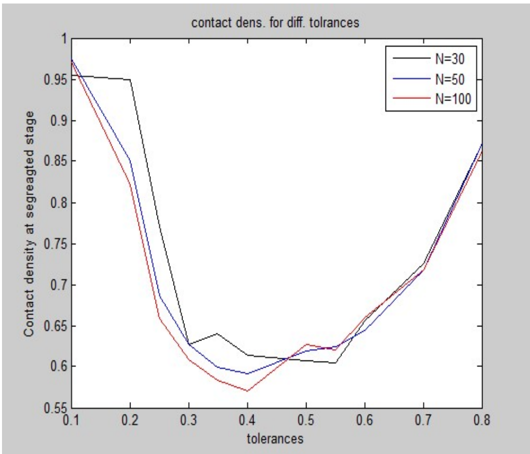


FIGURE 3.14: Plot of contact density versus tolerance for 3 different grid sizes (N = 30, 50, 100) with 0.5 fraction of activated fixed agents.

It is clearly observed that the in the frozen mixed stage, the contact density and hence the mixity is higher for larger grid size but the scenario opposes after the transition from segregated stage to mixed, i.e. in this phase the contact density is lower for larger grids.

Next, we plot contact density against fraction of deactivated (Fig. 3.15) and activated (Fig. 3.16) fixed agents for grid sizes N = 30, 50, 100.

Now, we plot the graphs for contact density versus $\frac{\delta}{r}$ for N = 30, 50, 100 for deactivated agents (Fig. 3.17) and activated agents (Fig. 3.18).

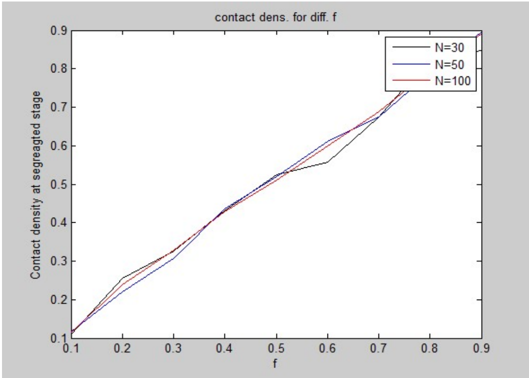


FIGURE 3.15: Plot of contact density versus f (deactivated agents) for N = 30, 50, 100 with fixed tolerance = 0.3.

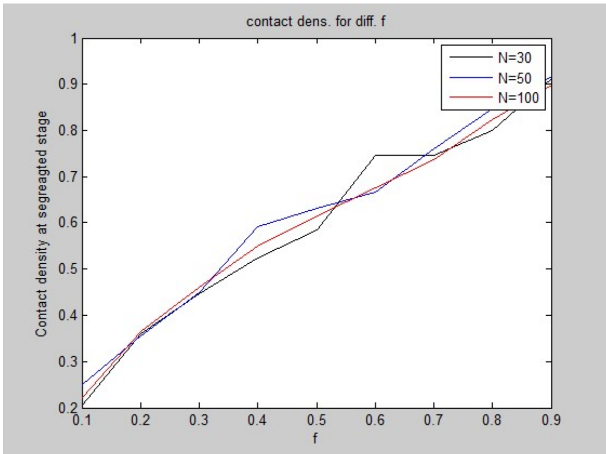


FIGURE 3.16: Plot of contact density versus f (activated agents) for N = 30, 50, 100 with fixed tolerance = 0.3.

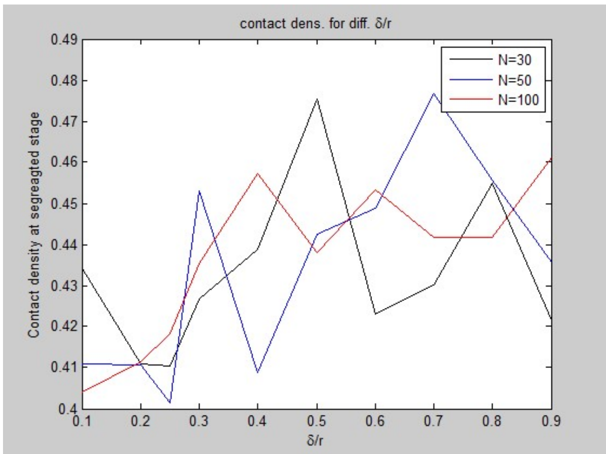


FIGURE 3.17: Plot of contact density versus $\frac{\delta}{r}$ for N = 30, 50, 100 with fixed tolerance = 0.3 for deactivated agents.

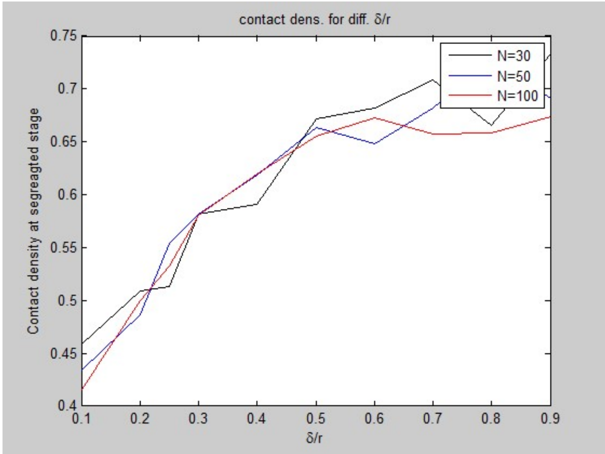
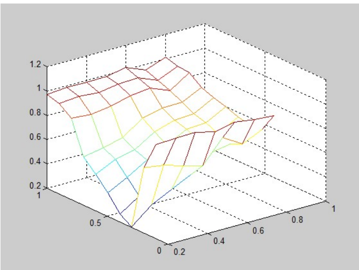
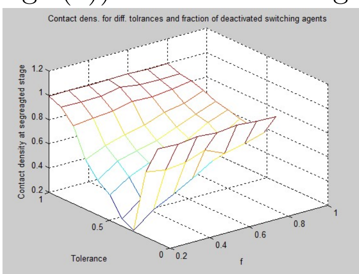
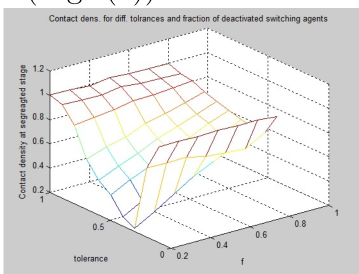


FIGURE 3.18: Plot of contact density versus $\frac{\delta}{r}$ for N = 30, 50, 100 with fixed tolerance = 0.3 for activated agents.

Next, we have 3D plots of contact density versus f and tolerance for 3 grid sizes N = 30, 50 and 100 for deactivated agents (Fig. (1)) and activated agents (Fig. (2)).

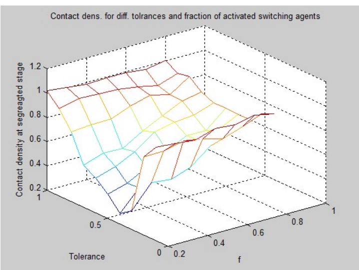


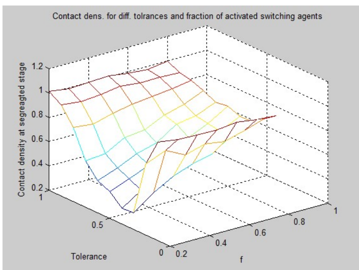


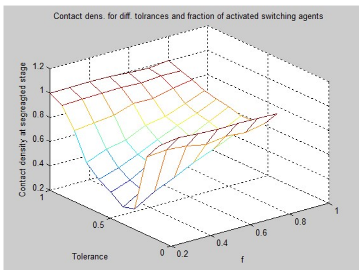


N = 30 N = 50 N = 100

(1) 3D plot of contact density versus f and tolerance for deactivated fixed agents for grid sizes N = 30, 50, 100.







N = 30 N = 50 N = 100

(2) 3D plot of contact density versus f and tolerance for activated fixed agents for grid sizes N = 30, 50, 100.

4. Appendix

The appendix is divided into 3 sections, section 4.1 includes the MATLAB codes for Chapter 1, 4.2 for Chapter 2 and 4.3 for Chapter 3.

4.1 MATLAB codes for random allocation of fixed agents

4.1.1 MATLAB code to plot contact density against time steps for 4 simulation modes (Fig. 1.1)

```
function contact_plot
clear all
close all
clc
toler=0.3;
f=0.20;
N=30;
[grid1 dummygrid1] = initial_mesh(f,N);
a=tic
[contact1 susep1]=schelling1a(grid1, dummygrid1,toler,N) ;
time1=toc(a)
b=tic
[contact2 susep2]=schelling1b(grid1, dummygrid1,toler,N) ;
time2=toc(b)
c=tic
 [contact3 susep3]=schelling1c(grid1, dummygrid1,toler,N) ;
 time3=toc(c)
d=tic
[contact4 susep4]=schelling1d(grid1, dummygrid1,toler,N) ;
time4=toc(d)
t=1:500;
figure
plot (t,contact1', t,contact2',t,contact3',t,contact4');
legend('No switching (f=0)',' No switching (f=0.2)',
'Switching starts at
 t=0','Switching starts at t=250');
title('Assemble average of contact density in 4 simulation modes');
xlabel('Time Steps ---->')
ylabel('Assemble average of contact density')
end
```

4.1.2(a) MATLAB code for running the Schelling dynamics with blind moves for the

case when no fixed agents are present in the system

```
function [contact susep] = schelling1a_blind(lattice,dummylattice
,toler,l_size)
figure
disp(lattice);
max = 500;                  % no. of time steps
visit= zeros(l_size);
contact = zeros(max,1);
tem = zeros(max,1);
susep = zeros(max,1);
u_move = 0.2;               % probability of moving if unhappy
h_move = 1e-04;             % probability of moving if happy
for t=1:max
    s=0;
    s1=0;
    c=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j,lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
            end
        end
    end
    contact(t)=2*s/c;
    tem(t) = 2*s1/c;
    susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
    xdist = randperm(l_size);
    ydist= randperm(l_size);
    for i=1:l_size
        for j=1:l_size
            x = xdist(mod(i+j,l_size)+1);
            y = ydist(j);
            if ( visit(x,y) == 0 && lattice(x,y) ~= 0)
                fr = opposite_frac(lattice,x,y,lattice(x,y));
                if (fr > toler)
                    if ( rand(1)< u_move )
                        [newx,newy] = moveto(lattice);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
```

```
                        tmp = lattice(x,y);
                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            else
                if ( rand(1)< h_move )
                    [newx,newy] = moveto(lattice);
                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                        tmp = lattice(x,y);
                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            end

        end
    end
  end
  disp(lattice);
  visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with no switching agents');
pause(0.015);
end
function flag = in(lattice , x, y)
flag = 0;
l_size = length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
end
function [agent1,frac] = n_frac(lattice , x, y)
n = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
frac = 0;
agent1 = 0;
count = 0;
```

```
for k=1:8
    n_x = x + n(k, 1);
    n_y= y + n(k, 2);
    if (in(lattice, n_x, n_y))
        if (lattice(n_x,n_y) ~= 0)
            count = count+1;
            if (lattice(n_x,n_y) == 1)
                agent1 = agent1 + 1;
            end
        end
    end
end
if (agent1 > 0)
    frac = agent1 / count;
end
end
function fr = opposite_frac(lattice,x,y,color)
[c,fr] = n_frac(lattice, x, y);
if (color == 1)
    fr = 1 - fr;
end
end
function [nx,ny] = moveto(lattice)
ka=0;
l_size= length(lattice);
for i=1:l_size
    for j=1:l_size
        if(lattice(i,j)==0)
            ka=ka+1;
            vac(ka,1)=i;
            vac(ka,2)=j;
        end
    end
end
r=randi([1,ka]);
nx = 0;
ny = 0;
flag = 0;
while (~flag)
    nx=vac(r,1);
    ny=vac(r,2);
    flag = 1;
    continue;
```

```
end
end
```

4.1.2(b) MATLAB code for running the Schelling dynamics with non blind moves for the case when no fixed agents are present in the system

```
%No switching agents
function [contact susep] = schelling1a(lattice,dummylattice,toler,l_size)
figure
disp(lattice);
pause(100)
max = 500;                  % no. of time steps
visit= zeros(l_size);
contact = zeros(max,1);
tem = zeros(max,1);
susep = zeros(max,1);
u_move = 0.2;               % probability of moving if unhappy
h_move = 1e-04;             % probability of moving if happy
for t=1:max
    s=0;
    s1=0;
    c=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j,lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
            end
        end
    end
    contact(t)=2*s/c;
    tem(t) = 2*s1/c;
    susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
    xdist = randperm(l_size);
    ydist= randperm(l_size);
    for i=1:l_size
        for j=1:l_size
            x = xdist(mod(i+j,l_size)+1);
            y = ydist(j);
            if ( visit(x,y) == 0 && lattice(x,y) ~= 0)
                fr = opposite_frac(lattice,x,y,lattice(x,y));
```

```
                if (fr > toler)
                    if ( rand(1)< u_move )
                        [newx,newy] = moveto(lattice ,x,y,lattice(x,y),toler);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
                            tmp = lattice(x,y);
                            lattice(x,y) = lattice(newx,newy);
                            lattice(newx,newy) = tmp;
                        end
                    end
                else
                    if ( rand(1)< h_move )
                        [newx,newy] = moveto(lattice ,x,y,lattice(x,y),toler);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
                            tmp = lattice(x,y);
                            lattice(x,y) = lattice(newx,newy);
                            lattice(newx,newy) = tmp;
                        end
                    end
                end

            end
        end
    end
    disp(lattice);
    visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with no switching agents');
pause(0.015);
end
function flag = in(lattice , x, y)
flag = 0;
l_size = length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
```

```
end
function flag = allowed(lattice, x1, y1, x0, y0)
flag = 0;
if (x1 ~= x0 || y1 ~= y0)
    if(lattice(x1, y1) == 0)
        flag = flag+1;
    end
end
end
function [agent1,frac] = n_frac(lattice, x, y)
n = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
frac = 0;
agent1 = 0;
count = 0;
for k=1:8
    n_x = x + n(k, 1);
    n_y= y + n(k, 2);
    if (in(lattice, n_x, n_y))
        if (lattice(n_x,n_y) ~= 0)
            count = count+1;
            if (lattice(n_x,n_y) == 1)
                agent1 = agent1 + 1;
            end
        end
    end
end
if (agent1 > 0)
    frac = agent1 / count;
end
end
function fr = opposite_frac(lattice,x,y,color)
[c,fr] = n_frac(lattice, x, y);
if (color == 1)
    fr = 1 - fr;
end
end
function r = neighbourhood(lattice, x, y,o)
h = 2*o;
l= 2*o-1;
first = [];
last = [];
r = [];
x1 = x - o;
```

```
y1 = y - o;
x2 = x1 + 2*o;
y2 = y1 + 2*o;
for i=0:h
    x3= x1+i;
    if (in(lattice,x3,y1) && allowed(lattice, x3, y1, x, y))
        first = [first ; [x3,y1]];
    end
    if (in(lattice,x3,y2) && allowed(lattice, x3, y2, x, y))
        last = [last ; [x3,y2]];
    end
end
for i=1:l
    y3= y1+i;
    if (in(lattice,x1,y3) && allowed(lattice,x1, y3, x, y))
        r = [r ; [x1,y3]];
    end
    if (in(lattice,x2,y3) && allowed(lattice, x2, y3, x, y))
        r = [r ; [x2,y3]];
    end
end
r = [first ; r ; last];
end
function [nx,ny] = moveto(lattice, x, y, color, delta)
nx = 0;
ny = 0;
flag = 0;
o = 1;
l_size= length(lattice);
omax = abs(max([(x-l_size),(l_size-x),(y-l_size),(l_size-y)]));
while (~flag)
    neigh = neighbourhood(lattice, x,y,o);
    N = size(neigh,1);
    dist= randperm(N);
    for k=1:N
        nx = neigh(dist(k), 1);
        ny = neigh(dist(k), 2);
        f = opposite_frac(lattice,nx,ny,color);
        if (delta > f)
            flag = 1;
            continue;
        end
    end
```

```
    o = o + 1;
    if ( o > omax )
        nx = 0;
        ny = 0;
        flag = 1;
        continue;
    end
end
end
```

4.1.3(a) MATLAB code for running the Schelling dynamics with blind moves for the case when deactivated fixed agents are present in the system

```
%Deactivated switching agents and moves are blind
function [contact susep] = schelling1b_blind(lattice,
dummylattice,toler,l_size)
figure
disp(lattice);
max = 500;
visit= zeros(l_size);
contact = zeros(max,1);
tem = zeros(max,1);
susep = zeros(max,1);
u_move = 0.2;
h_move = 1e-04;
for t=1:max
    s=0;
    s1=0;
    c=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j,lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
            end
        end
    end
    contact(t)=2*s/c;
    tem(t) = 2*s1/c;
    susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
    xdist = randperm(l_size);
```

```
    ydist= randperm(l_size);
    for i=1:l_size
        for j=1:l_size
            x = xdist(mod(i+j,l_size)+1);
            y = ydist(j);
            if ( visit(x,y) == 0 && lattice(x,y) ~= 0
            && dummylattice(x,y)~=3)
                fr = opposite_frac(lattice,x,y,lattice(x,y));
                if (fr > toler)
                    if ( rand(1)< u_move )
                        [newx,newy] = moveto(lattice);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
                            tmp = lattice(x,y);
                            lattice(x,y) = lattice(newx,newy);
                            lattice(newx,newy) = tmp;
                        end
                    end
                else
                    if ( rand(1)< h_move )
                        [newx,newy] = moveto(lattice);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
                            tmp = lattice(x,y);
                            lattice(x,y) = lattice(newx,newy);
                            lattice(newx,newy) = tmp;
                        end
                    end
                end

            end
        end
    end
    disp(lattice);
    visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with deactivated switching agents');
```

```
pause(0.015);
end
function flag = in(lattice, x, y)
flag = 0;
l_size = length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
end
function [agent1,frac] = n_frac(lattice, x, y)
n = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
frac = 0;
agent1 = 0;
count = 0;
for k=1:8
    n_x = x + n(k, 1);
    n_y= y + n(k, 2);
    if (in(lattice, n_x, n_y))
        if (lattice(n_x,n_y) ~= 0)
            count = count+1;
            if (lattice(n_x,n_y) == 1)
                agent1 = agent1 + 1;
            end
        end
    end
end
if (agent1 > 0)
    frac = agent1 / count;
end
end
function fr = opposite_frac(lattice,x,y,color)
[c,fr] = n_frac(lattice, x, y);
if (color == 1)
    fr = 1 - fr;
end
end
function [nx,ny] = moveto(lattice)
ka=0;
l_size= length(lattice);
for i=1:l_size
    for j=1:l_size
        if(lattice(i,j)==0)
            ka=ka+1;
```

```
                vac(ka,1)=i;
                vac(ka,2)=j;
            end
        end
end
r=randi([1,ka]);
nx = 0;
ny = 0;
flag = 0;
while (~flag)
    nx=vac(r,1);
    ny=vac(r,2);
    flag = 1;
    continue;
end
end
```

4.1.3(b) MATLAB code for running the Schelling dynamics with non blind moves for the case when deactivated fixed agents are present in the system

```
%Rachneet Kaur
%Inactive switching agents
function [contact susep] = schelling1b(lattice,dummylattice,toler,l_size)
figure
disp(lattice);
max = 500;
u_move = 0.2;
h_move = 1e-04;
contact = zeros(max,1);
tem = zeros(max,1);
susep = zeros(max,1);
visit = zeros(l_size);
for t=1:max
    s=0;
    c=0;
    s1=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j, lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
```

```
        end
    end
end
contact(t)=2*s/c;
tem(t) = s1/c;
susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
xdist = randperm(l_size);
ydist= randperm(l_size);
for i=1:l_size
    for j=1:l_size
        x = xdist(mod(i+j,l_size)+1);
        y = ydist(j);
        if ( visit(x,y) == 0 && lattice(x,y) ~= 0
        && dummylattice(x,y)~=3)
            frac = opposite_frac(lattice,x,y, lattice(x,y));
            if (frac > toler)
                if ( rand(1)< u_move )
                    [newx,newy] = moveto(lattice,dummylattice,x,y,
                    lattice(x,y),toler);

                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                        tmp = lattice(x,y);

                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            else
                if ( rand(1)< h_move )
                    [newx,newy] = moveto(lattice,dummylattice,x,y,
                    lattice(x,y),toler);
                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                        tmp = lattice(x,y);

                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            end
```

```
            end
        end
    end
    disp(lattice);
    visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with deactivated switching agents');
pause(0.015);
end
function flag = in(lattice, x, y)
flag = 0;
l_size = length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
end
function flag = allowed(lattice, x1, y1,x0,y0)
flag = 0;
if (x1 ~=x0 || y1 ~=y0)
    if (lattice(x1, y1) == 0)
        flag = flag+1;
    end
end
end
function [agent1,frac] = n_frac(lattice, x, y)
neigh = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
frac = 0;
agent1= 0;
count = 0;
for k=1:8
    n_x = x + neigh(k, 1);
    n_y = y + neigh(k, 2);
    if (in(lattice, n_x, n_y))
        if (lattice(n_x,n_y) ~= 0)
            count = count+1;
            if (lattice(n_x,n_y) == 1)
                agent1 = agent1+ 1;
            end
        end
```

```
    end
end
if (agent1 > 0)
    frac = agent1/ count;
end
end
function fr = opposite_frac(lattice,x,y,type)
[c,fr] = n_frac(lattice, x, y);
if (type == 1)
    fr = 1 - fr;
end
end
function r = neighbourhood(lattice, x, y, o)
h= o*2;
l = o*2-1;
first = [];
last = [];
r = [];
x1 = x - o;
y1 = y - o;
x2 = x1 + 2*o;
y2 = y1 + 2*o;
for i=0:h
    x3 = x1+i;
    if (in(lattice,x3,y1) && allowed(lattice, x3, y1, x, y))
        first = [first ; [x3,y1]];
    end
    if (in(lattice,x3,y2) && allowed(lattice, x3, y2, x, y))
        last = [last ; [x3,y2]];
    end
end
for i=1:l
    y3 = y1+i;
    if (in(lattice,x1,y3) && allowed(lattice, x1, y3, x, y))
        r = [r ; [x1,y3]];
    end
    if (in(lattice,x2,y3) && allowed(lattice, x2, y3, x, y))
        r = [r ; [x2,y3]];
    end
end
r = [first ; r ; last];
end
function [nx,ny] = moveto(lattice,dummylattice, x, y, color, delta)
```

```
nx = 0;
ny = 0;
flag = 0;
o = 1;
l_size = length(lattice);
maxo= abs(max([(x-l_size),(l_size-x),(y-l_size),(l_size-y)]));
while (~flag)
    neigh = neighbourhood(lattice, x,y,o);
    l_size = size(neigh,1);
    dist = randperm(l_size);
    for k=1:l_size
        nx = neigh(dist(k), 1);
        ny = neigh(dist(k), 2);
        f = opposite_frac(lattice,nx,ny,color);
        if (delta > f)
            if(dummylattice(nx,ny)~=3)
                flag = 1;
                continue;
            end
        end
    end
    o = o + 1;
    if ( o > maxo )
        nx = 0;
        ny = 0;
        flag= 1;
        continue;
    end
end
end
```

4.1.4(a) MATLAB code for running the Schelling dynamics with blind moves for the case when activated fixed agents are present in the system

```
%Rachneet Kaur
%Activated switching agents activated in the
beginning itself and moves are blind
function [contact susep] = schelling1c_blind(lattice,
dummylattice,
toler,l_size)
figure
disp(lattice);
max = 500;
visit= zeros(l_size);
```

```
contact = zeros(max,1);
tem = zeros(max,1);
susep = zeros(max,1);
u_move = 0.2;
h_move = 1e-04;
p_switch=0.05;
for t=1:max
    s=0;
    s1=0;
    c=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j,lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
            end
        end
    end
    contact(t)=2*s/c;
    tem(t) = 2*s1/c;
    susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
    xdist = randperm(l_size);
    ydist= randperm(l_size);
    for i=1:l_size
        for j=1:l_size
            x = xdist(mod(i+j,l_size)+1);
            y = ydist(j);
            if ( visit(x,y) == 0 && lattice(x,y) ~= 0
            && dummylattice(x,y)~=3)
                fr = opposite_frac(lattice,x,y,lattice(x,y));
                if (fr > toler)
                    if ( rand(1)< u_move )
                        [newx,newy] = moveto(lattice);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
                            tmp = lattice(x,y);
                            lattice(x,y) = lattice(newx,newy);
                            lattice(newx,newy) = tmp;
                        end
                    end
```

```
                else
                    if ( rand(1)< h_move )
                        [newx,newy] = moveto(lattice);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
                            tmp = lattice(x,y);
                            lattice(x,y) = lattice(newx,newy);
                            lattice(newx,newy) = tmp;
                        end
                    end
                end

            end
            if ( dummylattice(x,y)== 3 && visit(x,y)==0)
                visit(x,y)=1;
                if (rand(1)< p_switch)
                    if(lattice(x,y)==1)
                        lattice(x,y)=2;
                    else
                        lattice(x,y)=1;
                    end
                end
            end
        end
    end

    disp(lattice);
    visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with activated switching agents');
pause(0.015);
end
function flag = in(lattice, x, y)
flag = 0;
l_size = length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
```

```
end
function [agent1,frac] = n_frac(lattice, x, y)
n = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
frac = 0;
agent1 = 0;
count = 0;
for k=1:8
    n_x = x + n(k, 1);
    n_y= y + n(k, 2);
    if (in(lattice, n_x, n_y))
        if (lattice(n_x,n_y) ~= 0)
            count = count+1;
            if (lattice(n_x,n_y) == 1)
                agent1 = agent1 + 1;
            end
        end
    end
end
if (agent1 > 0)
    frac = agent1 / count;
end
end
function fr = opposite_frac(lattice,x,y,color)
[c,fr] = n_frac(lattice, x, y);
if (color == 1)
    fr = 1 - fr;
end
end
function [nx,ny] = moveto(lattice)
ka=0;
l_size= length(lattice);
for i=1:l_size
    for j=1:l_size
        if(lattice(i,j)==0)
            ka=ka+1;
            vac(ka,1)=i;
            vac(ka,2)=j;
        end
    end
end
r=randi([1,ka]);
nx = 0;
ny = 0;
```

```
flag = 0;
while (~flag)
    nx=vac(r,1);
    ny=vac(r,2);
    flag = 1;
    continue;
end
end
```

4.1.4(b) MATLAB code for running the Schelling dynamics with non blind moves for the case when activated fixed agents are present in the system

```
%Rachneet Kaur
%Active switching agents ( activated in the beginning itself )
function [contact susep] = schelling1c(lattice,dummylattice,toler,l_size)
figure
disp(lattice);
max = 500;
u_move = 0.2;
h_move = 1e-04;
p_switch = 0.05;
contact = zeros(max,1);
visit = zeros(l_size);
tem = zeros(max,1);
susep = zeros(max,1);
for t=1:max
    s=0;
    s1=0;
    c=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j, lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
            end
        end
    end
    contact(t)=2*s/c;
    tem(t) = s1/c;
    susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
    xdist = randperm(l_size);
```

```
ydist = randperm(l_size);
for i=1:l_size
    for j=1:l_size
        x = xdist(mod(i+j, l_size)+1);
        y = ydist(j);
        if ( visit(x,y) == 0 && lattice(x,y) ~= 0 && dummylattice(x,y)~=3
            fr = opposite_frac(lattice,x,y, lattice(x,y));
            if (fr > toler)
                if (rand(1)< u_move )
                    [newx,newy] = moveto(lattice,dummylattice,x,y,
                    lattice(x,y),toler);
                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                        tmp = lattice(x,y);
                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            else
                if ( rand(1)< h_move )
                    [newx,newy] = moveto(lattice,dummylattice,x,y,
                    lattice(x,y),toler);
                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                        tmp = lattice(x,y);
                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            end
        end
        if ( dummylattice(x,y)== 3 && visit(x,y)==0)
            visit(x,y)=1;
            if (rand(1)< p_switch)
                if(lattice(x,y)==1)
                    lattice(x,y)=2;
                else
                    lattice(x,y)=1;
                end
            end
        end
```

```
        end
    end
    disp(lattice);
    visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with activated switching agents')
pause(0.015);
end
function flag = in(lattice, x, y)
flag = 0;
l_size = length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
end
function flag =allowed(lattice,x1,y1, x0,y0)
flag = 0;
if (x1 ~= x0 || y1 ~=y0)
    if (lattice(x1,y1) == 0)
        flag = flag+1;
    end
end
end
function [agent1,ret] = n_frac(lattice, x, y)
agent1 = 0;
ret = 0;
n = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
occupy = 0;
for k=1:8
    n_x= x + n(k, 1);
    n_y = y + n(k, 2);
    if (in(lattice,n_x, n_y))
        if (lattice(n_x,n_y) ~= 0)
            occupy = occupy+1;
            if (lattice(n_x,n_y) == 1)
                agent1 = agent1 + 1;
            end
        end
    end
```

```
end
if (agent1 > 0)
    ret = agent1 / occupy;
end
end
function pr= opposite_frac(lattice,x,y,color)
[c,pr] = n_frac(lattice, x, y);
if (color == 1)
    pr = 1 - pr;
end
end
function r = neighbourhood(lattice, x, y, o)
h = 2*o;
l = 2*o-1;
x1 = x - o;
y1 = y - o;
x2= x1+ 2*o;
y2= y1 + 2*o;
first = [];
last= [];
r = [];
for i=0:h
    x3 = x1+i;
    if (in(lattice,x3,y1) && allowed(lattice, x3, y1, x, y))
        first = [first ; [x3,y1]];
    end
    if (in(lattice,x3,y2) && allowed(lattice, x3, y2, x, y))
        last = [last ; [x3,y2]];
    end
end
for i=1:l
    y3 = y1+i;
    if (in(lattice,x1,y3) && allowed(lattice, x1, y3, x, y))
        r = [r ; [x1,y3]];
    end
    if (in(lattice,x2,y3) && allowed(lattice, x2, y3, x, y))
        r = [r ; [x2,y3]];
    end
end
r = [first ; r ; last];
end
function [xnew,ynew] = moveto(lattice,dummylattice, x, y, color,prop)
xnew = 0;
```

```
ynew = 0;
flag = 0;
o = 1;
l_size = length(lattice);
maxo = abs(max([(x-l_size),(l_size-x),(y-l_size),(l_size-y)]));
while (~flag)
    neigh = neighbourhood(lattice, x,y,o);
    l_size = size(neigh,1);
    dist = randperm(l_size);
    for k=1:l_size
        xnew = neigh(dist(k), 1);
        ynew = neigh(dist(k), 2);
        f = opposite_frac(lattice,xnew,ynew,color);
        if (prop > f)
            if(dummylattice(xnew,ynew)~=3)
                flag = 1;
                continue;
            end
        end
    end
    o = o + 1;
    if ( o > maxo )
        xnew = 0;
        ynew = 0;
        flag = 1;
        continue;
    end
end
end
```

4.1.5(a) MATLAB code for running the Schelling dynamics with blind moves for the case when fixed agents activated after half time are present in the system

```
%Rachneet Kaur
%Activated switching agents activated after half time and moves are blind
function [contact susep] = schelling1d_blind(lattice,dummylattice,
toler,l_size)
figure
disp(lattice);
max = 500;
visit= zeros(l_size);
contact = zeros(max,1);
tem = zeros(max,1);
```

```
susep = zeros(max,1);
u_move = 0.2;
h_move = 1e-04;
p_switch=0.05;
for t=1:max
    s=0;
    s1=0;
    c=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j,lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
            end
        end
    end
    contact(t)=2*s/c;
    tem(t) = 2*s1/c;
    susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
    xdist = randperm(l_size);
    ydist= randperm(l_size);
    for i=1:l_size
        for j=1:l_size
            x = xdist(mod(i+j,l_size)+1);
            y = ydist(j);
            if ( visit(x,y) == 0 && lattice(x,y) ~= 0
             && dummylattice(x,y)~=3)
                fr = opposite_frac(lattice,x,y,lattice(x,y));
                if (fr > toler)
                    if ( rand(1)< u_move )
                        [newx,newy] = moveto(lattice);
                        if (newx ~= 0)
                            visit(x,y) = 1;
                            visit(newx,newy) = 1;
                            tmp = lattice(x,y);
                            lattice(x,y) = lattice(newx,newy);
                            lattice(newx,newy) = tmp;
                        end
                    end
                else
                    if ( rand(1)< h_move )
```

```
                    [newx,newy] = moveto(lattice);
                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                        tmp = lattice(x,y);
                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            end

        end
        if(t>250)
            if ( dummylattice(x,y)== 3 && visit(x,y)==0)
                visit(x,y)=1;
                if (rand(1)< p_switch)
                    if(lattice(x,y)==1)
                        lattice(x,y)=2;
                    else
                        lattice(x,y)=1;
                    end
                end
            end
        end
    end
end

    disp(lattice);
    visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with activated switching agents after half time');
pause(0.015);
end
function flag = in(lattice, x, y)
flag = 0;
l_size = length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
```

```
end
function [agent1,frac] = n_frac(lattice, x, y)
n = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
frac = 0;
agent1 = 0;
count = 0;
for k=1:8
    n_x = x + n(k, 1);
    n_y= y + n(k, 2);
    if (in(lattice, n_x, n_y))
        if (lattice(n_x,n_y) ~= 0)
            count = count+1;
            if (lattice(n_x,n_y) == 1)
                agent1 = agent1 + 1;
            end
        end
    end
end
if (agent1 > 0)
    frac = agent1 / count;
end
end
function fr = opposite_frac(lattice,x,y,color)
[c,fr] = n_frac(lattice, x, y);
if (color == 1)
    fr = 1 - fr;
end
end
function [nx,ny] = moveto(lattice)
ka=0;
l_size= length(lattice);
for i=1:l_size
    for j=1:l_size
        if(lattice(i,j)==0)
            ka=ka+1;
            vac(ka,1)=i;
            vac(ka,2)=j;
        end
    end
end
r=randi([1,ka]);
nx = 0;
ny = 0;
```

```
flag = 0;
while (~flag)
    nx=vac(r,1);
    ny=vac(r,2);
    flag = 1;
    continue;
end
end
```

4.1.5(b) MATLAB code for running the Schelling dynamics with non blind moves for the case when fixed agents activated after half time are present in the system

```
%Rachneet Kaur
%Active switching agents ( activated after half time )
function [contact susep] = schelling1d(lattice,dummylattice,toler,l_size)
figure
disp(lattice);
max = 500;
u_move = 0.2;
h_move = 1e-04;
p_switch = 0.05;
visit = zeros(l_size);
contact = zeros(max,1);
tem = zeros(max,1);
susep = zeros(max,1);
for t=1:max
    s=0;
    s1=0;
    c=0;
    for i=1:l_size
        for j=1:l_size
            if (lattice(i,j)~=0)
                c=c+1;
                temp = opposite_frac(lattice,i,j,lattice(i,j));
                s=s+temp;
                s1=s1+(temp*temp);
            end
        end
    end
    contact(t)=2*s/c;
    tem(t) = s1/c;
    susep(t) = (tem(t)-(contact(t)*contact(t)))/ toler;
    xdist = randperm(l_size);
```

```
ydist = randperm(l_size);
for i=1:l_size
    for j=1:l_size
        x = xdist(mod(i+j, l_size)+1);
        y = ydist(j);
        if ( visit(x,y) == 0 && lattice(x,y) ~= 0
         && dummylattice(x,y)~=3)
            fr= opposite_frac(lattice,x,y,lattice(x,y));
            if (fr> toler)
                if ( rand(1) < u_move )
                    [newx,newy] = moveto(lattice,dummylattice,x,y,
                    lattice(x,y),toler);
                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                       tmp = lattice(x,y);
                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            else
                if ( rand(1) < h_move )
                    [newx,newy] = moveto(lattice,dummylattice,x,y,
                    lattice(x,y),toler);
                    if (newx ~= 0)
                        visit(x,y) = 1;
                        visit(newx,newy) = 1;
                       tmp = lattice(x,y);
                        lattice(x,y) = lattice(newx,newy);
                        lattice(newx,newy) = tmp;
                    end
                end
            end
        end
        if (t>250)
            if ( dummylattice(x,y)== 3 && visit(x,y)==0)
                visit(x,y)=1;
                if (rand(1)< p_switch)
                    if(lattice(x,y)==1)
                        lattice(x,y)=2;
                    else
                        lattice(x,y)=1;
                    end
```

```
                    end
                end
            end
        end
    end
    disp(lattice);
    visit = zeros(l_size);
end
end
function disp(lattice)
imagesc(lattice);
colormap([1 1 1; 1 0 0; 0 0 1]);
title('Segregation model with switching agents activated after half time');
pause(0.015);
end
function flag = in(lattice, x, y)
flag = 0;
l_size= length(lattice);
if (x >= 1 && y >= 1 && x <= l_size && y <= l_size)
    flag = flag+1;
end
end
function flag = allowed(lattice, x1, y1, x0,y0)
flag = 0;
if (x1 ~= x0 || y1 ~= y0)
    if (lattice(x1, y1) == 0)
        flag = flag+1;
    end
end
end
function [agent1,frac] = n_frac(lattice, x, y)
neigh = [-1 -1; 0 -1; 1 -1; 1 0; 1 1; 0 1; -1 1; -1 0];
frac = 0;
agent1 = 0;
count = 0;
for k=1:8
    ngbr_r = x + neigh(k, 1);
    ngbr_c = y + neigh(k, 2);
    if (in(lattice, ngbr_r, ngbr_c))
        if (lattice(ngbr_r,ngbr_c) ~= 0)
            count = count+1;
            if (lattice(ngbr_r,ngbr_c) == 1)
                agent1 = agent1 + 1;
```

```
            end
        end
    end
end
if (agent1 > 0)
    frac = agent1 / count;
end
end
function fr = opposite_frac(lattice,x,y,color)
[c,fr] = n_frac(lattice, x, y);
if (color == 1)
    fr = 1 - fr;
end
end
function r = neighbourhood(lattice, x, y, o)
h = o*2;
l = o*2-1;
x1 = x - o;
y1 = y - o;
x2 = x1 + 2*o;
y2 = y1 + 2*o;
r = [];
first = [];
last = [];
for i=0:h
    x3 = x1+i;
    if (in(lattice,x3,y1) && allowed(lattice, x3, y1, x, y))
        first = [first ; [x3,y1]];
    end
    if (in(lattice,x3,y2) && allowed(lattice, x3, y2, x, y))
        last = [last ; [x3,y2]];
    end
end
for i=1:l
    y3 = y1+i;
    if (in(lattice,x1,y3) && allowed(lattice, x1, y3, x, y))
        r = [r ; [x1,y3]];
    end
    if (in(lattice,x2,y3) && allowed(lattice, x2, y3, x, y))
        r = [r ; [x2,y3]];
    end
end
r = [first ; r ; last];
```

```
end
function [nx,ny] = moveto(lattice,dummylattice, x, y, color, delta)
nx = 0;
ny = 0;
flag = 0;
o = 1;
l_size = length(lattice);
maxo = abs(max([(x-l_size),(l_size-x),(y-l_size),(l_size-y)]));
while (~flag)
    neigh = neighbourhood(lattice, x,y,o);
    l_size = size(neigh,1);
    dist = randperm(l_size);
    for k=1:l_size
        nx = neigh(dist(k), 1);
        ny = neigh(dist(k), 2);
        f = opposite_frac(lattice,nx,ny,color);
        if (delta > f)
            if(dummylattice(nx,ny)~=3)
                flag = 1;
                continue;
            end
        end
    end
    o = o + 1;
    if ( o > maxo )
        nx = 0;
        ny = 0;
        flag = 1;
        continue;
    end
end
end
```

4.1.6 MATLAB code to plot contact density against tolerance for f =0, 0.25 and 0.5 (Fig. 1.2, 1.3, 1.4)

```
function fig2
clear all
close all
clc
toler=0.1:0.1:1
f=[0.25 0.50];
f1=0.25;
```

```
[grid1 dummygrid1] = generateRandomGrid(f1);
for j=1:10
[contact1 susep1]=schelling1a(grid1, dummygrid1,toler(j))
c(j)=contact1(500)
end
figure
plot(toler,c);
xlabel('Tolerance');
ylabel('Contact density')
title('At f=0');
for i=1:2
for j=1:10
[grid1 dummygrid1] = generateRandomGrid(f(i));
[contact3 susep3]=schelling1c(grid1, dummygrid1,toler(j)) ;
c(j)=contact3(500);
end
figure
plot(toler,c);
xlabel('Tolerance');
ylabel('Contact density')
if (i==1)
title('At f=0.25');
else
title('At f=0.50');
end
end
end
```

4.1.7 MATLAB code to plot contact density against tolerance and occupation density for f =0, 0.25 and 0.5 (Fig. 1.5, 1.6, 1.7)

```
function fig5
clear all
close all
clc
toler=0.1:0.1:1
f=[0.25 0.50];
f1=0.25;
vacancy = [0.40 0.35 0.30 0.25 0.20 0.15 0.10 0.05]
%vacancy = [0.40 0.35]
for i=1:8
occ(i)= 1-vacancy(i);
end
```

```
for i=1:8
 %for i=1:2
[grid1 dummygrid1] = generaterandomgrid(f1,30,vacancy(i));
for j=1:10
[contact1 susep1]=schelling1a(grid1, dummygrid1,toler(j),30)
c(i,j)=contact1(500)
end
end
[X,Y]= meshgrid(occ,toler)
figure
mesh(X,Y,c');
xlabel('Occupation density');
ylabel('Tolerance')
zlabel('Contact density')
title('At f=0');
for k=1:2
    for i=1:8
[grid1 dummygrid1] = generaterandomgrid(f(k),30,vacancy(i));
for j=1:10
[contact3 susep3]=schelling1c(grid1, dummygrid1,toler(j),30) ;
c(i,j)=contact3(500);
end
    end
    [X,Y]= meshgrid(occ,toler)
figure
mesh(X,Y,c');
xlabel('Occupation density');
ylabel('Tolerance')
zlabel('Contact density')
if (k==1)
title('At f=0.25');
else
title('At f=0.50');
end
end
end
```

4.1.8 MATLAB code to plot contact density against tolerance and f (Fig. 1.8)

```
function fig8
clear all
close all
clc
```

```
toler=0.1:0.1:1
f=[0.1 0.2 0.3 0.4 0.50];
f1=0.25;
f2=[0 0.1 0.2 0.3 0.4 0.50];
[grid1 dummygrid1] = generaterandomgrid(f1,30,0.15);
for j=1:10
[contact1 susep1]=schelling1a(grid1, dummygrid1,toler(j),30)
c(j,1)=contact1(500)
end
for i=1:5
for j=1:10
[grid1 dummygrid1] = generaterandomgrid(f(i),30,0.15);
[contact3 susep3]=schelling1c(grid1, dummygrid1,toler(j),30) ;
c(j,i+1)=contact3(500);
end
end
figure
[X,Y]=meshgrid(toler,f2);
mesh(X,Y,c');
xlabel('Tolerance');
ylabel('fraction of fixed agents')
zlabel('Contact density')
title('Contact density against tolerance and f');
end
```

4.1.9 MATLAB code to plot contact density against f (Fig. 1.9)

```
function fig9
clear all
close all
clc
toler=0.3;
f=0:0.1:0.8
f1=0.25;
N=30;
[grid1 dummygrid1] = generaterandomgrid(f1,30,0.15);
[contact1 susep1]=schelling1a(grid1, dummygrid1,toler,N);
c(1)=contact1(500);
for j=2:9
[grid2 dummygrid2] = generaterandomgrid(f(j),30,0.15);
[contact3 susep3]=schelling1c(grid2, dummygrid2,toler,N) ;
c(j)=contact3(500);
end
```

```
c
figure
plot(f,c);
xlabel('f (fraction of fixed switching agents) ----->');
ylabel('Contact density ---->')
title('Transition of contact dens. along f at fixed tol. = 0.3');
end
```

4.1.10 MATLAB code to generate a random initial grid (used as a function in the above codes)

```
function [grid dummygrid] = generaterandomgrid(f,N,vacancies)
agent1=0.5;
indexes = randperm(N);
for i=1:N
 x = indexes(i);
 for j=1:N
 y = indexes(j);
 if (rand(1) > vacancies)
 if (rand(1)>f)
 if (rand(1) > agent1)
 grid(x,y) = 2;
 dummygrid(x,y)=2;
 else
 grid(x,y) = 1;
 dummygrid(x,y)=1;
 end
 else
 dummygrid(x,y)=3;
 if (rand(1) > agent1)
 grid(x,y) = 2;
 else
 grid(x,y) = 1;
 end
 end
 else
grid(x,y)=0;
dummygrid(x,y)=0;
end
end
end
end
```

4.2 MATLAB codes for determined annular allocation of fixed agents

4.2.1 MATLAB code for initialization of fixed agents inside annular ring

```
%The initial grid in case of an annulus.
function [lattice dummylattice] = Initial_mesh_ann(N,r1,delta)
H=N/2;
vacant_frac = 0.15;
agent1=0.5;
r2=r1+ delta;
dist = randperm(N);
for i=1:N
    x = dist(i);
    for j=1:N
        y = dist(j);
        if ((((x-H)*(x-H))+ ((y-H)*(y-H))) >= r1*r1)
         && ((((x-H)*(x-H))+ ((y-H)*(y-H))) <r2*r2)
            dummylattice(x,y)=3;
            if (rand(1) > agent1)
                lattice(x,y) = 2;
            else
                lattice(x,y) = 1;
            end
        else
            if (rand(1) > vacant_frac)
                if (rand(1) > agent1)
                    lattice(x,y) = 2;
                    dummylattice(x,y)=2;
                else
                    lattice(x,y) = 1;
                    dummylattice(x,y)=1;
                end
            else
                lattice(x,y)=0;
                dummylattice(x,y)=0;
            end
        end

    end
end
end
```

4.2.2 MATLAB code for initialization of fixed agents on the outer circle

```
%Tthe initial grid in case switching agents are outside the circle
function [lattice dummylattice] = Initial_mesh_outerofcircle(N,r1)
H=N/2;
vacant_frac = 0.15;
agent1=0.5;
dist = randperm(N);
for i=1:N
    x = dist(i);
    for j=1:N
        y = dist(j);
        if (rand(1) > vacant_frac)
            if ((((x-H)*(x-H))+ ((y-H)*(y-H))) >= r1*r1)
                dummylattice(x,y)=3;
                if (rand(1) > agent1)
                    lattice(x,y) = 2;
                else
                    lattice(x,y) = 1;
                end
            else
                if (rand(1) > agent1)
                    lattice(x,y) = 2;
                    dummylattice(x,y)=2;
                else
                    lattice(x,y) = 1;
                    dummylattice(x,y)=1;
                end
            end
        else
            lattice(x,y)=0;
            dummylattice(x,y)=0;
        end
    end
end
end
```

4.2.3 MATLAB code for plot of Contact density against $\frac{\delta}{r}$ for deactivated switching agents placed in the annular ring (Fig. 2.1)

```
%Rachneet Kaur
%IIT Delhi
function main_plot3
clear all
close all
```

```
clc
toler=0.3;
N=100;
xpara = 0:0.05:0.5;
r2=N*xpara;
ypara =0:0.1:1;
c=zeros(11,1);
for i=10:11
    for j=1:11
        delta(j)=ypara(j)*r2(i);
        r1(j)=r2(i)-delta(j);
        if((i==1 &&j==1)||(i~=1 && j~=1))
            [mesh1 dummymesh1] =Initial_mesh_ann(N,r1(j),delta(j));
            [contact3 susep3]=schelling1b(mesh1, dummymesh1,toler,N) ;
            c(j)= contact3(500);
        end
    end
    if(i==1)
        figure
        plot (ypara(i),c(i));
        xlabel('delta/outer radius');
        ylabel('Contact density')
        title(r2(i));
    else
        figure
        plot (ypara,c');
        title(r2(i));
        xlabel('delta/outer radius');
        ylabel('Contact density')
    end
end
end
```

4.2.4 MATLAB code for plot of Contact density against $\frac{\delta}{r}$ for deactivated switching agents placed in the annular ring (Fig. 2.2)

```
%Rachneet Kaur
%IIT Delhi

function main_plot1
clear all
close all
clc
```

```
toler=0.3;
N=100;
xpara = 0:0.05:0.5;
r2=N*xpara;
ypara =0:0.1:1;
c=zeros(11,1);
for i=1:11
    for j=1:11
        delta(j)=ypara(j)*r2(i);
        r1(j)=r2(i)-delta(j);
        if((i==1 &&j==1)||(i~=1 && j~=1))
            [mesh1 dummymesh1] =Initial_mesh_ann(N,r1(j),delta(j));
            [contact3 susep3]=schelling1c(mesh1, dummymesh1,toler,N) ;
            c(j)= contact3(500);
        end
    end
    if(i==1)
        figure
        plot (ypara(i),c(i));
        xlabel('delta/outer radius');
        ylabel('Contact density')
        title(r2(i));
    else
        figure
        plot (ypara,c');
        title(r2(i));
        xlabel('delta/outer radius');
        ylabel('Contact density')
    end
end
end
```

4.2.5 MATLAB code for plot of Contact density against $\frac{r}{N}$ for activated switching agents placed on outer of circle (Fig. 2.5)

```
function main_plot2
clear all
close all
clc
toler=0.3;
N=100;
xpara = 0:0.05:0.5;
r=N*xpara;
```

```
c=zeros(11,1);
for i=6:11
    [mesh1 dummymesh1] =Initial_mesh_outerofcircle(N,r(i));
    [contact3 susep3]=schelling1c(mesh1, dummymesh1,toler,N) ;
    c(i)= contact3(500);
end
figure
plot (xpara,c');
title('contact density for various outer radius values' );
xlabel('radius/grid size');
ylabel('Contact density')
end
```

4.2.6 MATLAB code for plot of Contact density against $\frac{r}{N}$ for deactivated switching agents placed in the annular ring (Fig. 2.6)

```
function main_plot11
clear all
close all
clc
toler=0.3;
N=100;
xpara = 0:0.05:0.5;
r2=N*xpara;
ypara =0:0.1:1;
c=zeros(11,1);
for i=1:11
    for j=1:11
        delta(j)=ypara(j)*r2(i);
        r1(j)=r2(i)-delta(j);
        if((i==1 &&j==1)||(i~=1 && j~=1))
            [mesh1 dummymesh1] =Initial_mesh_ann(N,r1(j),delta(j));
            [contact3 susep3]=schelling1b(mesh1, dummymesh1,toler,N) ;
            c(i,j)= contact3(500);
        end
    end
end
figure
[X Y] = meshgrid(xpara,ypara)
mesh(X,Y,c')
title('3D Plot of contact density for deactivated agents inside annulus \
versus r/N and \delta/r')
xlabel('r/N')
```

```
ylabel('\delta/r')
zlabel('Contact density for activated agents')
end
```

4.2.7 MATLAB code for plot of Contact density against $\frac{r}{N}$ for activated switching agents placed in the annular ring (Fig. 2.7)

```
%Rachneet Kaur
%IIT Delhi
function main_plot11
clear all
close all
clc
toler=0.3;
N=100;
xpara = 0:0.05:0.5;
r2=N*xpara;
ypara =0:0.1:1;
c=zeros(11,1);
for i=1:11
    for j=1:11
        delta(j)=ypara(j)*r2(i);
        r1(j)=r2(i)-delta(j);
        if((i==1 &&j==1)||(i~=1 && j~=1))
            [mesh1 dummymesh1] =Initial_mesh_ann(N,r1(j),delta(j));
            [contact3 susep3]=schelling1c(mesh1, dummymesh1,toler,N) ;
            c(i,j)= contact3(500);
        end
    end
end
figure
[X Y] = meshgrid(xpara,ypara)
mesh(X,Y,c')
title('3D Plot of contact density for activated agents inside annulus
versus r/N and \delta/r')
xlabel('r/N')
ylabel('\delta/r')
zlabel('Contact density for activated agents')
end
```

4.2.8 MATLAB code for plotting contact density against tolerance and $\frac{\delta}{r}$ for deactivated switching agents placed in the annular ring

```
%Rachneet Kaur
```

```
%IIT Delhi
function main_plot32
clear all
close all
clc
toler=[ 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9];
N=50;
xpara = 0.25;
r2=N*xpara;
ypara =0:0.1:1;
c=zeros(8,11);
for i=1:8
    for j=1:11
        delta(j)=ypara(j)*r2;
        r1=r2-delta(j);
        if((i==1 &&j==1)||(i~=1 && j~=1))
            [mesh1 dummymesh1] =Initial_mesh_ann(N,r1,delta(j));
            [contact3 susep3]=schelling1b(mesh1, dummymesh1,toler(i),N) ;
            c(i,j)= contact3(500);
        end
    end
end
[X Y]= meshgrid(toler,ypara)
figure
mesh(X,Y,c')
xlabel('Tolerance');
ypara('\delta/r')
zlabel('contact density')
title('contact density at segregated stage for deactivated agents
against tolerance and \delta/r');
end
```

4.2.9 MATLAB code for plotting contact density against tolerance and $\frac{\delta}{r}$ for activated switching agents placed in the annular ring

```
%Rachneet Kaur
%IIT Delhi
function main_plot32
clear all
close all
clc
toler=[ 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9];
N=50;
```

```
xpara = 0.25;
r2=N*xpara;
ypara =0:0.1:1;
c=zeros(8,11);
for i=1:8
    for j=1:11
        delta(j)=ypara(j)*r2;
        r1=r2-delta(j);
        if((i==1 &&j==1)||(i~=1 && j~=1))
            [mesh1 dummymesh1] =Initial_mesh_ann(N,r1,delta(j));
            [contact3 susep3]=schelling1c(mesh1, dummymesh1,toler(i),N) ;
            c(i,j)= contact3(500);
        end
    end
end
[X Y]= meshgrid(toler,ypara)
figure
mesh(X,Y,c')
xlabel('Tolerance');
ypara('\delta/r')
zlabel('contact density')
title('contact density at segregated stage for activated agents
against tolerance and \delta/r');
end
```

4.3 MATLAB codes for gaussian annular allocation of fixed agents

4.3.1 MATLAB code for plot of contact density versus fraction of deactivated and activated agents for fixed tol. (Fig. 3.1)

```
% when certain fraction of agents inside the annulus (defined by gaussian
% distribution) are switching agents
function imitate_gaussian
clear all
close all
clc
toler=0.33;
N=100;
xpara = 0.3;
r1=N*xpara;
ypara =0.25;
delta=ypara*r1;
```

```
f= [0 0.08 0.1 0.2 0.30  0.4 0.5 0.6 0.7 0.8 0.9 1];
c=zeros(12,1);
d=zeros(12,1);
for i=7:7
    [mesh1 dummymesh1] =initialmesh_ann_imitate_gaussian(N,r1,delta,f(i));
    [contact2 susep2]=schelling1b(mesh1, dummymesh1,toler,N) ;
    [contact3 susep3]=schelling1c(mesh1, dummymesh1,toler,N) ;
    c(i)= contact3(500)
    d(i)= contact2(500)
end
figure
plot (f,c','*-b',f,d','o-k');
legend('With activated switching agents',
'With deactivated switching agents');
title('contact dens. for diff. fractions of switching agents in annulus');
xlabel('fraction of switching agents inside annulus');
ylabel('Contact density at segreagted stage');
end
```

4.3.2 MATLAB code for plot of contact density at segregated stage versus tolerance for fixed fraction of activated/deactivated switching agents = 0.5 (Fig. 3.3)

```
function imitate_gaussian_tolerchange
clear all
close all
clc
toler = [ 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1];
N=100;
xpara = 0.3;
r1=N*xpara;
ypara =0.25;
delta=ypara*r1;
f= 0.5;
c=zeros(10,1);
d=zeros(10,1);
[mesh1 dummymesh1] =initialmesh_ann_imitate_gaussian(N,r1,delta,f);
for j=1:10
[contact2 susep2]=schelling1b_blind(mesh1, dummymesh1,toler(j),N) ;
[contact3 susep3]=schelling1c_blind(mesh1, dummymesh1,toler(j),N) ;
c(j)= contact3(500)
d(j)= contact2(500)
end
f
```

```
c'
d'
figure
plot (toler,c','*-b',toler,d','o-k');
legend('With activated switching agents','With
deactivated switching agents');
title('contact dens. for diff. tolrances');
xlabel('tolerances');
ylabel('Contact density at segreagted stage');
end
```

4.3.3 MATLAB code to initialize a 100 X 100 grid with fixed agents allocated on a gaussian ring (Fig. 3.2)

```
function [lattice dummylattice]= initialmesh_ann_imitate_gaussian(N,r1,
delta,switch_ann)
vacant_frac = 0.15;
H=N/2;
agent1=0.5;
r2=r1+ delta;
sites=floor(switch_ann*N*N*(1-vacant_frac))
dummylattice=zeros(N);
i=1;
while(i<=sites)
    theta = 2*pi*rand(1);
    d = delta*randn(1) +r1;
    x=H+d*cos(theta);
    y=H+d*sin(theta);
    n_x = floor(x);
    n_y=floor(y);
    if ((n_x >=1 && n_x <=N)&&(n_y >=1 && n_y <=N) )
        if(dummylattice(n_x,n_y)==0)
            i=i+1;
            dummylattice(n_x,n_y)=3;
            if (rand(1) > agent1)
                lattice(n_x,n_y) = 2;
            else
                lattice(n_x,n_y) = 1;
            end
        end
    end
end
te = 1- (switch_ann)*(1-vacant_frac);
```

```
new_vacant= vacant_frac/te;
dist = randperm(N);
for i=1:N
    x0 = dist(i);
    for j=1:N
        y0 = dist(j);
        if(dummylattice(x0,y0)~=3)
            if (rand(1) > new_vacant)
                if (rand(1) > agent1)
                    lattice(x0,y0) = 2;
                    dummylattice(x0,y0)=2;
                else
                    lattice(x0,y0) = 1;
                    dummylattice(x0,y0)=1;
                end
            else
                lattice(x0,y0)=0;
                dummylattice(x0,y0)=0;
            end
        end
    end
end
end
```

4.3.4 MATLAB code for 3D plot of contact density at segregated stage versus tolerance and fraction of deactivated/ activated switching agents (Fig. 3.5, 3.6)

```
    function chap3contact_f_tol
    toler = [ 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1]
    N=100;
    xpara = 0.3;
    r1=N*xpara;
    ypara =0.25;
    delta=ypara*r1;
    f= [0.2 0.3 0.4 0.5 0.6 0.7 0.8 ]
    for i=1:7
    [mesh1 dummymesh1] =initialmesh_ann_imitate_gaussian(N,r1,delta,f(i))
    for j=1:10
    [contact2 susep2]=schelling1b_blind(mesh1, dummymesh1,toler(j),N) ;
    [contact3 susep3]=schelling1c_blind(mesh1, dummymesh1,toler(j),N) ;
    c(i,j)= contact3(500)
    d(i,j)= contact2(500)
    end
    end
```

```
    figure
    [X,Y]= meshgrid(f,toler)
    mesh (X,Y,c')
    title('Contact dens. for diff. tolrances and fraction of activated
    switching agents');
    xlabel('f');
    ylabel('Tolerance')
    zlabel('Contact density at segreagted stage');
    figure
    [X,Y]= meshgrid(f,toler);
    mesh (X,Y,d);
    title('Contact dens. for diff. tolrances and fraction of deactivated
    switching agents');
    xlabel('f');
    ylabel('Tolerance')
    zlabel('Contact density at segreagted stage');
    end
```

4.3.5 MATLAB code for 3D plot of contact density at segregated stage versus fraction and $\frac{\delta}{r}$ for deactivated/ activated switching agents (Fig. 3.7, 3.8)

```
function chap3contact_f_deltabyr
clear all
close all
clc
toler = 0.4
N=50;
xpara = 0.3;
r1=N*xpara;
ypara =[0.1 0.2 0.25 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1];
delta=ypara*r1;
f= [0.2 0.3 0.4 0.5 0.6 0.7 0.8 ]
for j=1:11
    for i=1:7
[mesh1 dummymesh1] =initialmesh_ann_imitate_gaussian(N,r1,delta(j),f(i))
 [contact2 susep2]=schelling1b_blind(mesh1, dummymesh1,toler,N) ;
[contact3 susep3]=schelling1c_blind(mesh1, dummymesh1,toler,N) ;
        c(i,j)= contact3(500)
        d(i,j)= contact2(500)
    end
end
figure
[X,Y]= meshgrid(f,ypara)
```

```
mesh (X,Y,c')
title('Contact dens. for diff. fraction of activated switching agents
and \delta/r');
xlabel('f');
ylabel('\delta/r')
zlabel('Contact density at segreagted stage');
figure
[X,Y]= meshgrid(f,ypara);
mesh (X,Y,d');
title('Contact dens. for diff. fraction of deactivated switching agents
and \delta/r');
xlabel('f');
ylabel('\delta/r')
zlabel('Contact density at segreagted stage');
end
```

4.3.6 MATLAB code for 3D plot of contact density at segregated stage versus tolerance and $\frac{\delta}{r}$ for deactivated/ activated switching agents (Fig. 3.9,3.10)

```
function chap3contact_tol_deltabyr
clear all
close all
clc
toler = [ 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9]
N=50;
xpara = 0.3;
r1=N*xpara;
ypara =[0.1 0.2 0.25 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1];
delta=ypara*r1;
f= 0.4
for i=1:9
    for j=1:11
[mesh1 dummymesh1] =initialmesh_ann_imitate_gaussian(N,r1,delta(j),f)
        [contact2 susep2]=schelling1b_blind(mesh1, dummymesh1,toler(i),N) ;
        [contact3 susep3]=schelling1c_blind(mesh1, dummymesh1,toler(i),N) ;
        c(i,j)= contact3(500)
        d(i,j)= contact2(500)
    end
end
figure
[X,Y]= meshgrid(toler,ypara)
mesh (X,Y,c')
title('Contact dens. for diff. tolerances of activated switching agents
```

```
 and \delta/r');
xlabel('tolerance');
ylabel('\delta/r')
zlabel('Contact density at segreagted stage');
figure
mesh (X,Y,d');
title('Contact dens. for diff. tolerances of deactivated switching agents
 and \delta/r');
xlabel('tolerance');
ylabel('\delta/r')
zlabel('Contact density at segreagted stage');
end
```

Concluding Remarks

The report discusses different variants of Schelling segregation models.
In chapter 1, the model with introduction of fixed switching agents at random locations is discussed and the plots are recreated as in [3].
Chapter 2 discusses the effect of fixed agents inside an annular ring and in Chapter 3, fixed agents are placed in a gaussian ring and the effects are noted.
Next I plan to study and compare the results with Cammaorota and Coragama's technique for first order phase transitions.

Bibliography

1. Schelling, Thomas C. "Models of segregation." The American Economic Review (1969): 488-493.

2. Gauvin, Laetitia, Jean Vannimenus, and J-P. Nadal. "Phase diagram of a Schelling segregation model." The European Physical Journal B 70.2 (2009): 293-304.

3. Hazan, Aurelien, and Julien Randon-Furling. "A Schelling model with switching agents: decreasing segregation via random allocation and social mobility." The European Physical Journal B 86.10 (2013): 1-9.