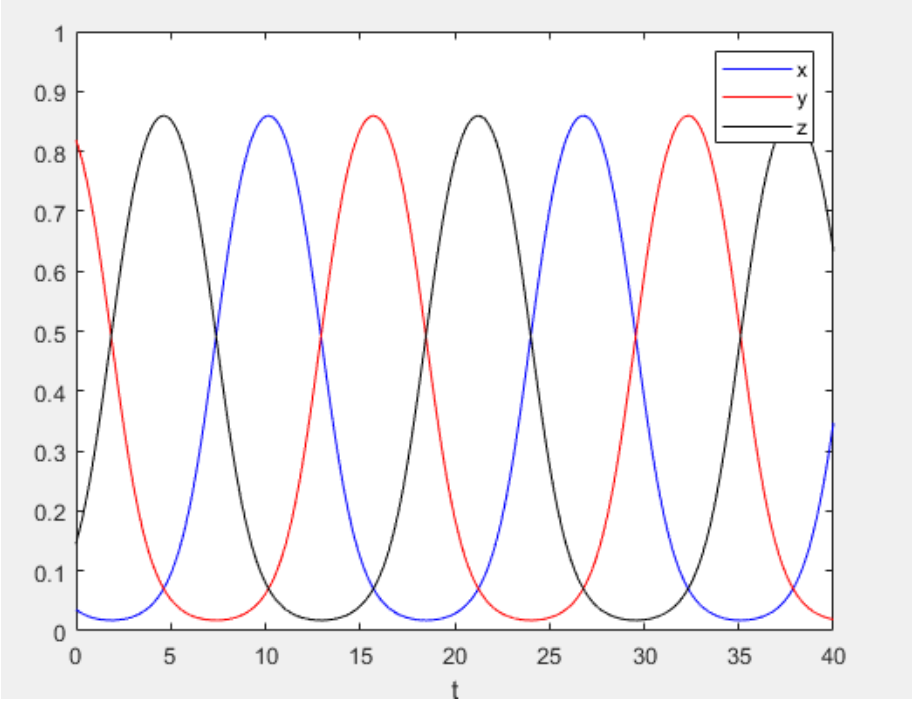
FIT3139 Assignment 2

Xinyu Ma
28652703

13/06/20

Part 1a



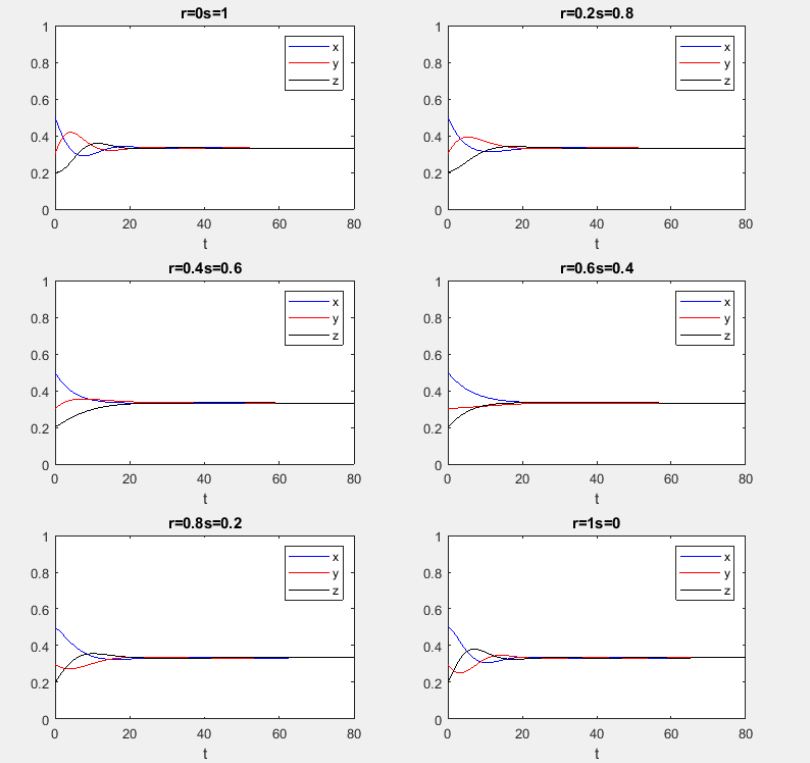

```
clc;clear;close all;
r=-1;
s=1;
x0=rand;
y0=(1-x0)*rand;
z0=1-x0-y0;
h=0.01;
max_T=40;
[x,y,z,t] = RK2_RPS(s,r,x0,y0,z0,h,max_T);
plot(t,x,'b-',t,y,'r-',t,z,'black-')
title('r=-2,s=-2')
legend('x','y','z')
xlabel('t')
axis([0 max_T 0 1])
```

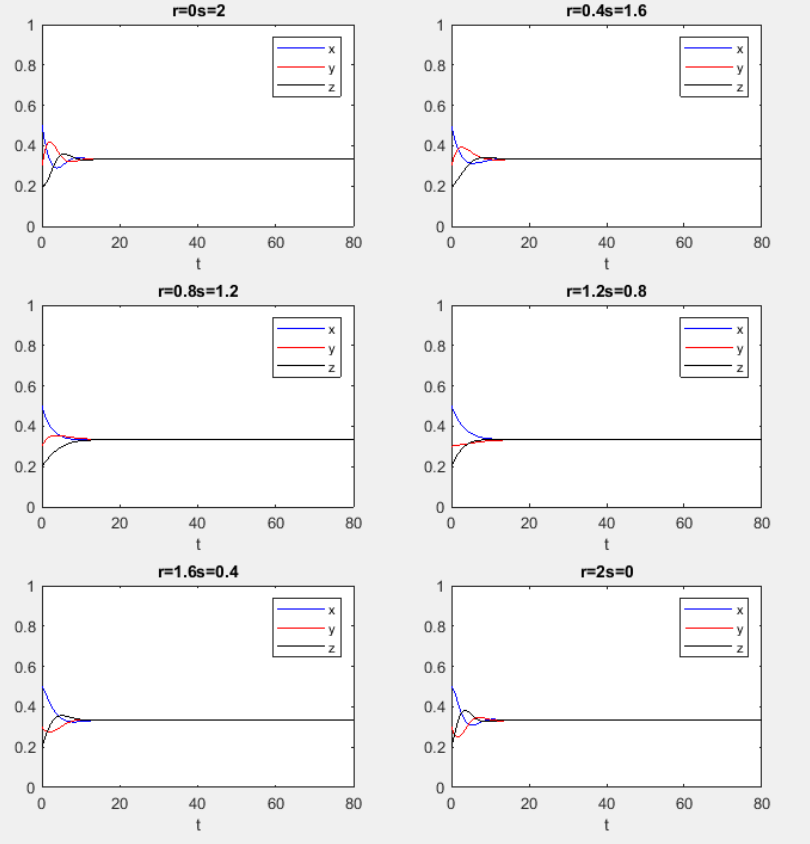
When r + s = 0, after the initial value is given, for any variables in x, y, z, the system shows a periodic change trend. The period of all three is the same, but the phase is different.

In order to ensure that the sum of the entire system remains unchanged, the increment of one variable is just offset by the decrease of the sum of the other two variables, which can be obtained by adding the three differential equations.

Part 1b

r + s > 0:


r=0s=1
r=0.2s=0.8
r=0.4s=0.6
r=0.6s=0.4
r=0.8s=0.2
r=1s=0
x
y
z
t


r=0s=2
r=0.4s=1.6
r=0.8s=1.2
r=1.2s=0.8
r=1.6s=0.4
r=2s=0
x
y
z
t

When r + s > 0, the three variables finally converge to the equal share.

After comparison, we found that the larger the value of r + s, the faster the convergence rate. According to the observation, the convergence rate is proportional to the value of r + s.

At the same time, while ensuring that r + s remains unchanged, changing the values of r and s will not affect the speed of convergence.

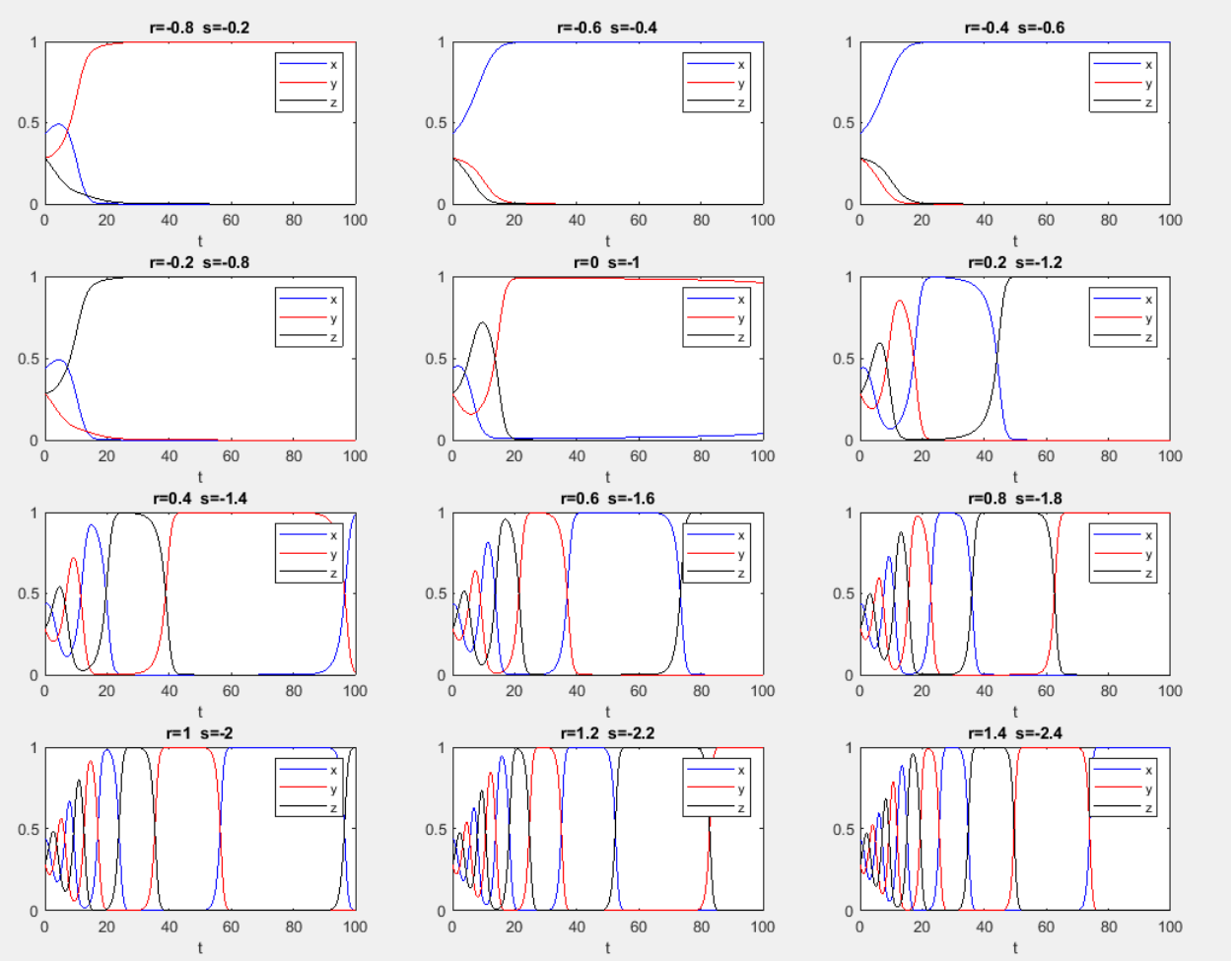
When the values of r and s are closer, the changing trend of the three variables is more stable.

It can be considered that r + s = 0 is the limit of r + s > 0. At this time, the convergence time will become infinity; that is, the image will fluctuate periodically.

Code for r + s > 0:

```
for k=1:5
    figure(k)
    for i=1:6
        subplot(3,2,i)
        r=0+(i-1)*0.2*k;
        s=1*k-(i-1)*0.2*k;
        % x0=rand;
        % y0=(1-x0)*rand;
        % z0=1-x0-y0;
        x0=0.5;
        y0=0.3;
        z0=0.2;
        h=0.01;
        max_T=80;
        [x,y,z,t] = RK2_RPS(s,r,x0,y0,z0,h,max_T);
        plot(t,x,'b-',t,y,'r-',t,z,'black-')
        legend('x','y','z')
        xlabel('t')
        axis([0 max_T 0 1])
        title(['r=' num2str(r) 's=' num2str(s)])
    end
end
```

r + s < 0:



When r⩽0, it will eventually converge to the situation of "one winner, two losers". Under the condition of ensuring that r + s remains unchanged, the status of winners and losers will change with the initial value and r value, and no pattern is found here.

When r > 0, although there will be a "winner" temporarily. But when the time axis is lengthened, we will find that the status of the loser and the winner will change every time, and as the number of replacements increases, the time interval between the next position replacement will become longer and longer. At the same time, in the same replacement, the larger the value of r, the shorter the position replacement time.

The larger the absolute value of r + s, the longer the time for the long-term winner to arrive when r<0. Similarly, for the extreme case of r + s = 0, the long-term winner will never come, and the three will always show periodic fluctuations in the "competition"

Winner: refers to a variable whose value is very close to or equal to 1

Loser: refers to a variable whose value is very close to or equal to 0

Long-term winner: refers to a variable whose value is limited to 1

Code for r + s < 0:

```
for k=1:5
    figure(k)
    for i=1:12
        subplot(4,3,i)
        r=0+(i-5)*0.2*k;
        s=-1*k-(i-5)*0.2*k;
        % x0=rand;
        % y0=(1-x0)*rand;
        % z0=1-x0-y0;
        x0=1/3+0.1;
        y0=1/3-0.05;
        z0=1/3-0.05;
        h=0.01;
        max_T=100;
        [x,y,z,t] = RK2_RPS(s,r,x0,y0,z0,h,max_T);
        plot(t,x,'b-',t,y,'r-',t,z,'black-')
        legend('x','y','z')
        xlabel('t')
        axis([0 max_T 0 1])
        title(['r=' num2str(r) '  s=' num2str(s)])
    end
end
```

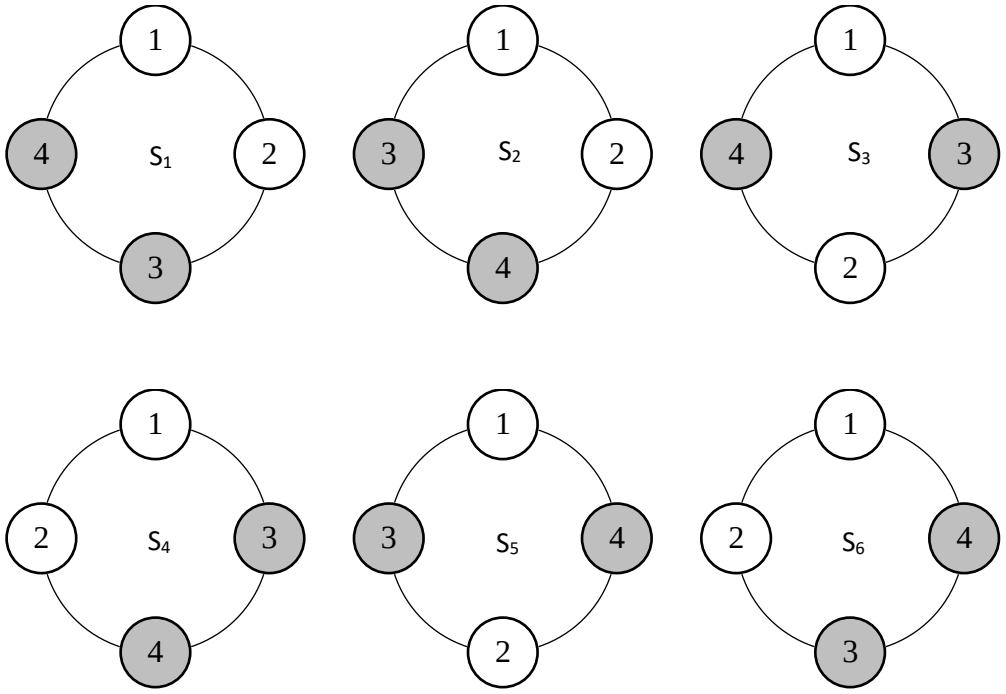
Part 2a

Specify explicitly the transition Matrix of the MC for n = 4. Explain how the transition probabilities are computed and how the states are labeled.

I will represent each agent with a circle, these agents have two categories: white and grey. White → "0" and Grey → "1".

We need two "0"s and two "1"s for simplified Schelling model with n = 4, we name each agent as A_1, A_2, A_3, A_4.

This question focuses on the state of the system itself rather than the exchange process, so I will try to label the different relative positions as one state. For this system, there should be six states as below:

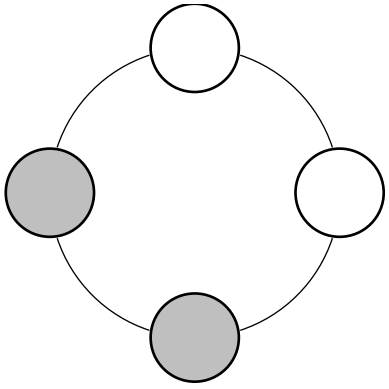


Record the sequence of these six states clockwise:

S_1	S_2	S_3	S_4	S_5	S_6
1-2-3-4	1-2-4-3	1-3-2-4	1-3-4-2	1-4-2-3	1-4-3-2

The probability of transferring between different states is then calculated separately, for a total of 36. Here are a few examples of the analytical calculations:

We can see that S_1, S_2, S_4, S_6 are in the “happiest” state, and no agent pair has the “desire” to switch positions, so these states are the absorbing states. Correspondingly, $p_{11} = p_{22} = p_{44} = p_{66} = 1$.



From $S_3 \rightarrow S_1$, only when A_2 and A_3 match successfully, the transfer can be successful. There are 6 match results, so $p_{31}=1/6$.

From $S_3 \rightarrow S_5$, the position of A_3 and A_4 are exchanged, but in S_3, even if they match A_3 and A_4, their positions will not change, so $p_{35}=0$.

From $S_3 \rightarrow S_2$, only when A_1 and A_3 match successfully, the transfer can be successful(A 90^o counterclockwise rotation is required), so $p_{32}=1/6$.

From $S_3 \rightarrow S_3$ is transferring to itself, the transfer will only be successful if it matches A_1 A_2 or A_3 A_4, so $p_{33}=1/3$.

Apply the above calculation to all transfer paths to obtain the transition matrix P_4:

	S_1	S_2	S_3	S_4	S_5	S_6
S_1	1	0	0	0	0	0
S_2	0	1	0	0	0	0
S_3	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{6}$	0	$\frac{1}{6}$
S_4	0	0	0	1	0	0
S_5	$\frac{1}{6}$	$\frac{1}{6}$	0	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{6}$
S_6	0	0	0	0	0	1

$= P_4$

Show the canonical form of the Markov chain for n = 4. Make sure to specify clearly how states are re-labeled or re-ordered if necessary.

Absorbing states: S_1, S_2, S_4, S_6

Transient states: S_3, S_5

We place transient states in the first two rows and two columns of the matrix, absorbing states in the last four rows and columns, then we can get canonical form of the Markov chain.

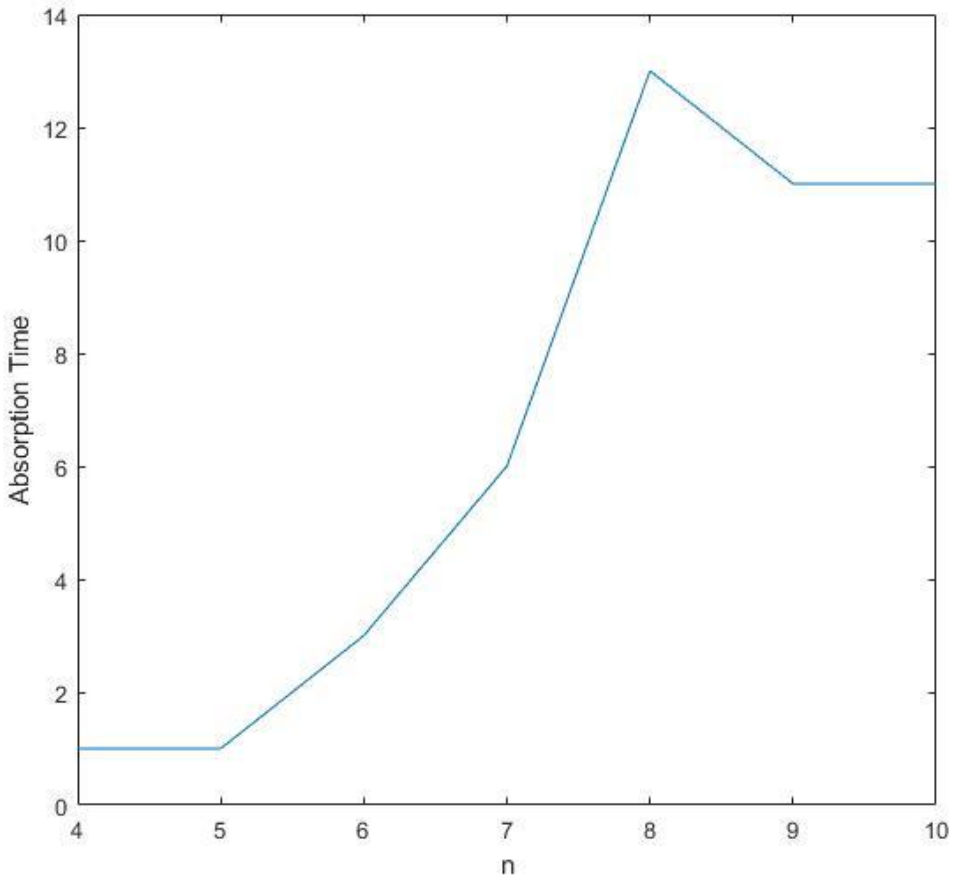
	S_3	S_5	S_1	S_2	S_4	S_6
S_3	$\frac{1}{3}$	0	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$
S_5	0	$\frac{1}{3}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$
S_1	0	0	1	0	0	0
S_2	0	0	0	1	0	0
S_4	0	0	0	0	1	0
S_6	0	0	0	0	0	1

	TR.	ABS.
TR.	**Q**	**R**
ABS.	**0**	**I**

$\mathbf{Q} = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{pmatrix}$, $\lim_{n\to\infty} \mathbf{Q}^n = \lim_{n\to\infty} \begin{pmatrix} \frac{1}{3^n} & 0 \\ 0 & \frac{1}{3^n} \end{pmatrix} = \mathbf{0}$, so this is an absorbing transient matrix.

Using Montecarlo simulations show how the absorption time varies with n = 4, 5, . . . 10.

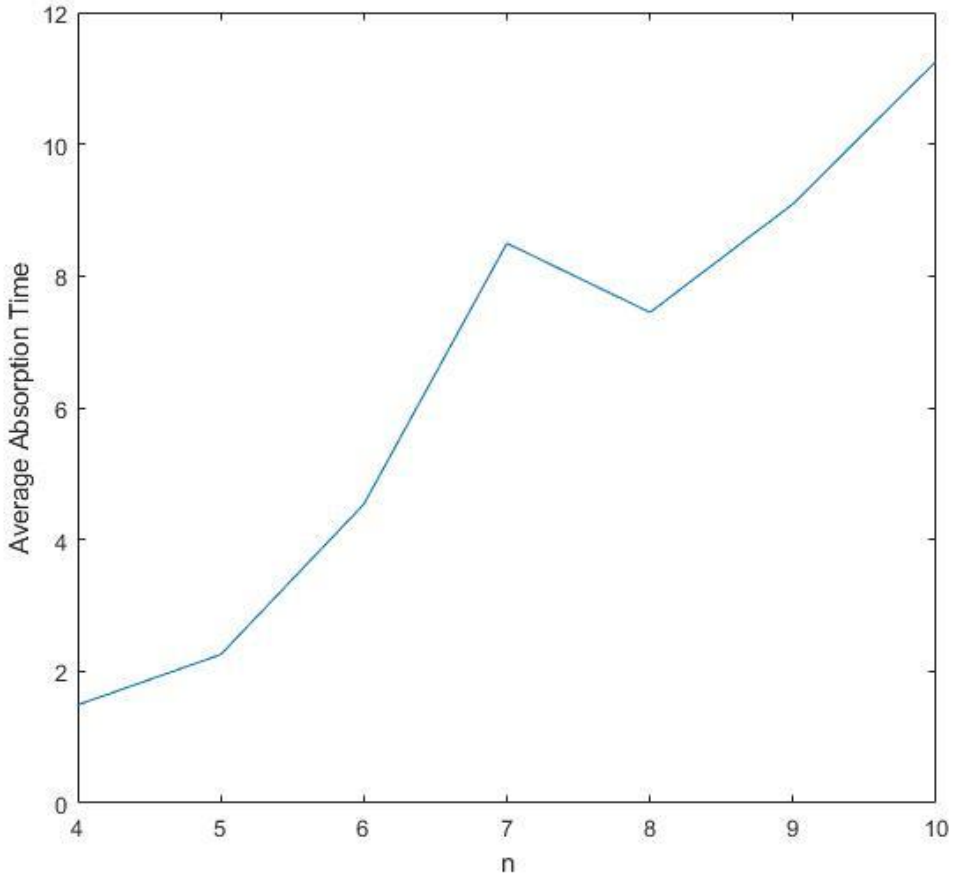
We conducted a simulation for n = 4, 5,... 10 respectively. When n is large, the number of states arranged is extremely large, and it is difficult to list them one by one, so the method of randomly generating the initial state is selected for simulation. The absorption time changes with n as shown in the figure.



n	4	5	6	7	8	9	10
AT	1	1	3	6	13	11	11

The absorption time of n=8 is longer than n=9 and 10, this conclusion is counterintuitive. The reason might be that the simulation is performed only once, which is more accidental. For example, the system may randomly generate an

initial state with a high "happiness level", and only a small amount of exchange is needed to reach the absorption state. Therefore, in order to ensure the generality of the results, we conducted 20,000 simulations (each simulation recorded the absorption time of n = 4, 5,... 10), and then averaged the absorption times of the 20,000 times to obtain the following Figure:



n	4	5	6	7	8	9	10
Average AT	1.504	2.515	5.031	9.340	7.725	9.378	11.647

After simulating 20,000 times to find the average value, we found that when n=8, the average absorption time is smaller than when n=7, 9, 10, which is a noteworthy phenomenon. The simulation procedure is as follows:

```
m=20000;

AT=zeros(m,10);
```

```
for i=1:m
    for n=4:10
        white=1:n/2;
        in=randperm(n);
        while calculate(in,n,white)==0
            in=randperm(n);
        end

        T=n*10;
        [~,unhappy] = Simplified_Schelling(n,white,in,T);
        a=find(unhappy==0,1);
        if ~isempty(a)
            AT(i,n)=a-1;
        else
            AT(i,n)=T;
        end
    end
end
plot(4:10,mean(AT(:,4:10)))
```

Numerically approximate the absorption times for n = 4 and n = 5 and show that they agree with the Montecarlo simulations.

To find the absorption time, we have to list the fundamental matrix corresponding to it:

$$\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1} = \begin{pmatrix} 3/2 & 0 \\ 0 & 3/2 \end{pmatrix}$$

$$t = \mathbf{N}c = \begin{pmatrix} 3/2 & 0 \\ 0 & 3/2 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 3/2 \\ 3/2 \end{pmatrix}$$

Starting from S_3 or S_3 with n=4, the average time required to reach the absorption state is 3/2, so when calculating the average value of the absorption time of all

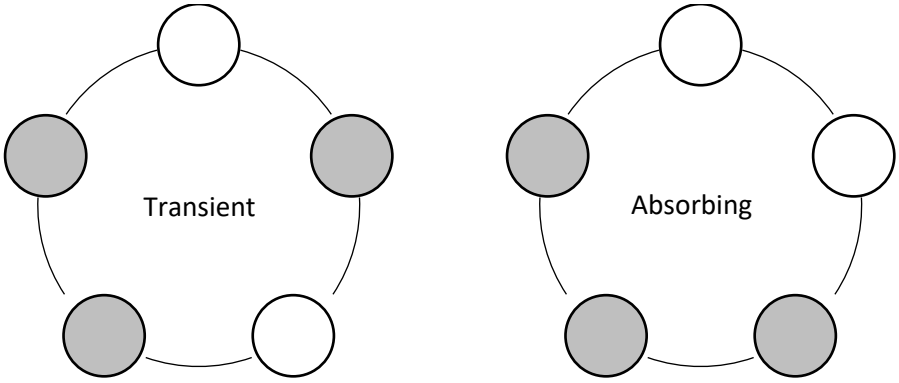
non-absorption states, the initial state can be considered as each non-absorption state when all occur with equal probability, the average absorption time is 3/2=1.5, which is very close to the result of Monte Carlo simulation 1.504.

Using the same analysis method, it can be concluded that the number of possible states when n=5 is 24, and the transition matrix is a square matrix of order 24, of which the non-absorption state is 12 (already listed in order in the table).

Code of looking for non-absorbing states:

```
n=5;
S=perms(2:n);
L=size(S,1);
white=1:n/2;
Qs=zeros(1,n);
k=1;
for i=1:L
    if calculate([1 S(i,:)],n,white)~=0
        Qs(k,1:n)=[1 S(i,:)];
        k=k+1;
    end
end
```

Since the number of "unhappy" agents in these 12 states is 3, there is no transition between these states, it is only possible to transform into an absorption state or remain unchanged. Due to this feature, the transition matrix Q of the non-absorbing state must be a diagonal matrix. The left side of the figure below shows the unsteady state distribution, and the right side shows the absorption state distribution.



All unsteady states have been found through computer programs. Number these states:

TR1	1	3	2	4	5
TR2	1	3	2	5	4
TR3	1	3	4	2	5
TR4	1	3	5	2	4
TR5	1	4	2	3	5
TR6	1	4	2	5	3
TR7	1	4	3	2	5
TR8	1	4	5	2	3
TR9	1	5	2	3	4
TR10	1	5	2	4	3
TR11	1	5	3	2	4
TR12	1	5	4	2	3

There are 4 kinds of changes from the left picture state to the right picture state (considering the result after rotation),and there are 6 possible matching ways to

keep the original state unchanged, so the probability of keeping the original state unchanged is P = 6/10 = 3/5, so we can get the diagonal matrix Q when n=5.

$$\mathbf{Q} = \begin{pmatrix} 3/5 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 3/5 \end{pmatrix}$$

All the elements on the diagonal of the matrix Q are 3/5, so the corresponding fundamental matrix:

$$\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1} = \begin{pmatrix} 5/2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 5/2 \end{pmatrix}$$

$$t = \mathbf{N}c = \begin{pmatrix} 5/2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 5/2 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} 5/2 \\ \vdots \\ 5/2 \end{pmatrix}$$

Starting from the non-absorption state of n=5, the average time required to reach the absorption state is 5/2, which is very close to the Monte Carlo simulation result of 2.515.

Part 2b

Specify the full transition matrix for n = 4, compute the stationary distribution numerically and show that it is in agreement with Montecarlo simulations. What can you conclude from this model?

First calculate the transition matrix after the disturbance ϵ is added. For the absorption states S_1, S_2 S_4, and S_6 in the original system, there will also be a probability of transition to other states. Take S_1 as an example:

From S_1 → S_2: A_3 matches A_4 and exchange each other, $p_{12} = \epsilon/6$

Similarly, $p_{13} = p_{14} = p_{15} = p_{16} = \epsilon/6$

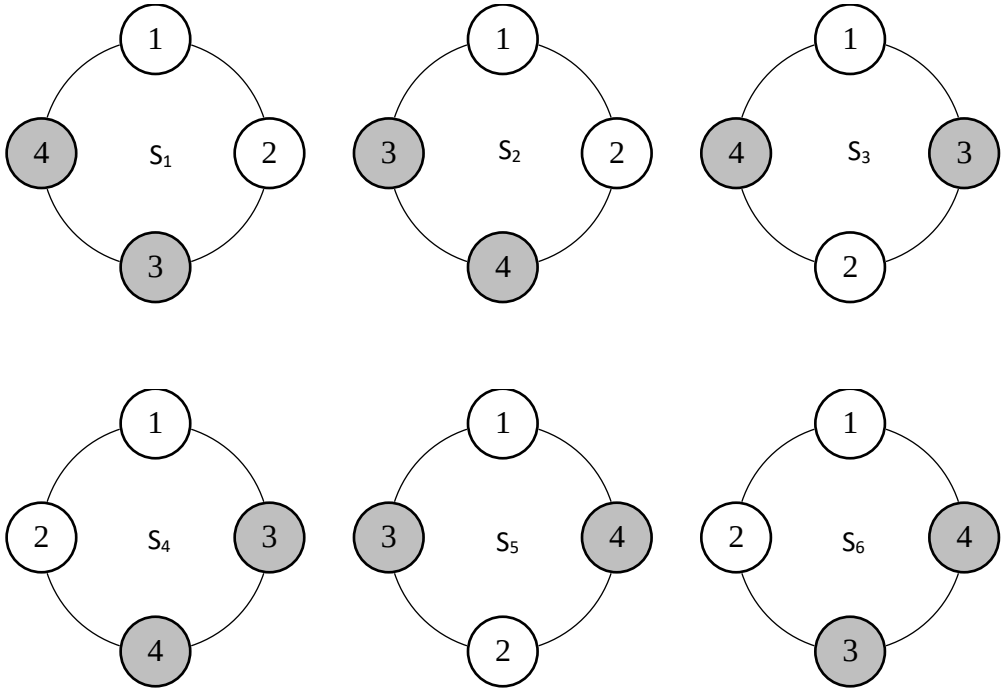
For non-absorbing state S_3 and S_5, take S_3 as an example:

From S_3 → S_5: A_3 matches A_4 and exchange each other, $p_{35} = \epsilon/6$

From S_3 → S_1: A_2 matches A_3 and exchange each other, $p_{31} = (1 - \epsilon)/6$

Similarly, $p_{32} = p_{34} = p_{36} = (1 - \epsilon)/6$

From $S_3 \rightarrow S_3$: $p_{33} = 1 - p_{31} - p_{32} - p_{34} - p_{35} - p_{36} = (2 + 3\epsilon)/6$



Full transition matrix:

	S_1	S_2	S_3	S_4	S_5	S_6
S_1	$\frac{6-5\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$
S_2	$\frac{\epsilon}{6}$	$\frac{6-5\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$
S_3	$\frac{1-\epsilon}{6}$	$\frac{1-\epsilon}{6}$	$\frac{2+3\epsilon}{6}$	$\frac{1-\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{1-\epsilon}{6}$
S_4	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{6-5\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$
S_5	$\frac{1-\epsilon}{6}$	$\frac{1-\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{1-\epsilon}{6}$	$\frac{2+3\epsilon}{6}$	$\frac{1-\epsilon}{6}$
S_6	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{6-5\epsilon}{6}$

Obviously, this is an ergodic chain.

The following table shows the stationary distribution results obtained by numerical calculation and Monte Carlo simulation under different values of ϵ. In the numerical calculation, through the *eigs* function in *matlab*, we get the eigenvector corresponding to the transition matrix (need to be transposed) with the eigenvalue of 1.

```
RP5=[6-5*e,e,e,e,e,e;
    e,6-5*e,e,e,e,e;
    1-e,1-e,2+3*e,1-e,e,1-e;
    e,e,e,6-5*e,e,e;
    1-e,1-e,e,1-e,2+3*e,1-e;
    e,e,e,e,e,6-5*e]/6;
[V,D]=eigs(RP5');
np=[V(:,1)/sum(V(:,1))]'
```

The Monte Carlo simulation was performed 10,000 times, each simulation generated a different initial state, and recorded the final state after the state was transferred 100 times according to the rules. Finally, the final state of the 10,000 simulation results is counted to obtain the frequency.

```
in=randperm(n);
T=100;
m=10000;
last=zeros(m,length(in));
S=zeros(1,6);
for i=1:m
[last(i,:),~,~] = Simplified_Schelling2(n,white,in,e,T);
if isequal(last(i,:),[1 2 3 4])
    S(1)=S(1)+1;
end
if isequal(last(i,:),[1 2 4 3])
    S(2)=S(2)+1;
end
if isequal(last(i,:),[1 3 2 4])
    S(3)=S(3)+1;
```

```
    end
    if isequal(last(i,:),[1 3 4 2])
        S(4)=S(4)+1;
    end
    if isequal(last(i,:),[1 4 2 3])
        S(5)=S(5)+1;
    end
    if isequal(last(i,:),[1 4 3 2])
        S(6)=S(6)+1;
    end
    end
    mp=S/sum(S)
```

The function used here has been modified based on the function of part a, and the judgment conditions have been changed according to the requirements of part b:

```
function [last,status,unhappy] = Simplified_Schelling2(n,white,in,e,T)
……
    r=rand;
    if or(and(unhappy(t+1)>=unhappy(t),r>=e),...
           and(unhappy(t+1)<unhappy(t),r<e))
        status(t+1,:)=status(t,:);
        unhappy(t+1)=unhappy(t);
    end
……
end
```

ϵ	Methods	S_1	S_2	S_3	S_4	S_5	S_6
0	Numerical	-	-	-	-	-	-
	Montecarlo	0.2468	0.2502	0	0.2533	0	0.2497
0.05	Numerical	0.2436	0.2436	0.0128	0.2436	0.0128	0.2436
	Montecarlo	0.2407	0.2464	0.0129	0.2380	0.0144	0.2476
0.1	Numerical	0.2368	0.2368	0.0263	0.2368	0.0263	0.2368
	Montecarlo	0.2286	0.2372	0.0230	0.2447	0.0286	0.2379
0.2	Numerical	0.2222	0.2222	0.0556	0.2222	0.0556	0.2222
	Montecarlo	0.2259	0.2221	0.0568	0.2175	0.0572	0.2205
0.5	Numerical	0.1667	0.1667	0.1667	0.1667	0.1667	0.1667
	Montecarlo	0.1626	0.1689	0.1728	0.1708	0.1669	0.1580
0.8	Numerical	0.0833	0.0833	0.3333	0.0833	0.3333	0.0833
	Montecarlo	0.0801	0.0831	0.3292	0.0828	0.3453	0.0795
1	Numerical	0	0	0.5	0	0.5	0
	Montecarlo	0	0	0.5025	0	0.4975	0

It can be found that the calculation results of the two methods are not much different. In this model, the larger the value of ϵ, the smaller the probability of the absorption state appearing in the original model (the model in part a), and the greater the probability of the occurrence of non-absorption state. When ϵ=1, the original non-absorbing state S_3 and S_5 become the absorbing state, and the original non-absorbing state no longer appears. The status of the two has changed.

Strangely, when ϵ=0, the model of the first question, its transition matrix has four eigenvalues of 1, so it cannot be calculated using the numerical method. However, the other extreme case when ϵ=1 has only one eigenvalue and is

eventually absorbed by S_3 and S_5. They both have absorbing chains in common and are no longer ergodic chains.

Repeat this analysis in the case where agents do not live on a cycle, but on a simple linear structure; i.e., the agents on both ends only have one neighbour.

The system structure changed from a ring to a chain, and the total number of states when n=4 changed from six to twenty-four. Of course, this is the number of permutations after considering different agent numbers. If the arrangement is not considered in the chain model, only the combination is considered, then there are six states. Although the number of states has not changed after the "de-numbering process", the calculation method of the two is actually different. In the computer, the state codes become '0011,0101,0110,1001,1010,1100' six kinds, respectively numbered L_1, L_2, L_3, L_4, L_5, L_6, it is easy to calculate. The number of unhappy people *u* in these six states is 0, 4, 2, 2, 4, 0, respectively.

On the original basis, the transition matrix (in the numerical method) and the calculation function of the unhappy agent (in the Monte Carlo simulation) need to be modified respectively. The modified part of the custom function is as follows:

```
function unhappy = calculate2(status,n,white)
……
if ss(1)~=ss(2)
    unhappy=unhappy+1;
end
……
if ss(n)~=ss(n-1)
    unhappy=unhappy+1;
end
end
```

An example of the calculation process of the transition probability between representative states: number the positions with P_1, P_2, P_3, P_4:

From L_1 → L_2, u will rise, so the transition condition is that P_2 and P_3 match and exchange, $p_{12} = \epsilon/6$.

Similarly, $p_{13} = p_{14} = p_{15} = \epsilon/6$.

From $L_1 \rightarrow L_6$, we cannot get it through one exchange, so $p_{16} = 0$, similarly, $p_{25} = p_{34} = p_{43} = p_{52} = p_{61} = 0$.

From $L_2 \rightarrow L_1$, u will fall, the transition condition is that P_2 and P_3 match and exchange, $p_{21} = (1 - \epsilon)/6$.

As for the transfer to itself, we only need to subtract the sum of the remaining probabilities by 1, for example:

From $L_1 \rightarrow L_1$, $p_{33} = 1 - p_{12} - p_{13} - p_{14} - p_{15} - p_{16} = (3 - 2\epsilon)/3$.

Full transition matrix:

	L_1	L_2	L_3	L_4	L_5	L_6
L_1	$\frac{3-2\epsilon}{3}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	0
L_2	$\frac{1-\epsilon}{6}$	$\frac{1+2\epsilon}{3}$	$\frac{1-\epsilon}{6}$	$\frac{1-\epsilon}{6}$	0	$\frac{1-\epsilon}{6}$
L_3	$\frac{1-\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{2}{3}$	0	$\frac{\epsilon}{6}$	$\frac{1-\epsilon}{6}$
L_4	$\frac{1-\epsilon}{6}$	$\frac{\epsilon}{6}$	0	$\frac{2}{3}$	$\frac{\epsilon}{6}$	$\frac{1-\epsilon}{6}$
L_5	$\frac{1-\epsilon}{6}$	0	$\frac{1-\epsilon}{6}$	$\frac{1-\epsilon}{6}$	$\frac{1+2\epsilon}{3}$	$\frac{1-\epsilon}{6}$
L_6	0	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{\epsilon}{6}$	$\frac{3-2\epsilon}{3}$

The numerical method and the Monte Carlo simulation method are used to obtain the results of stationary distribution. The calculation and simulation ideas are the same as the first question.

Compared with the first question, the code deletes the input related to the number, and only enters the category information of each position:

```
liner_states=[0,0,1,1;0,1,0,1;0,1,1,0;1,0,0,1;1,0,1,0;1,1,0,0];
n=4;
in=liner_states(randi(6),:);
T=100;
m=10000;
```

```
last=zeros(m,length(in));
S=zeros(1,6);
for i=1:m
[last(i,:),~,~] = Simplified_Schelling2L(n,in,e,T);
for j=1:6
    if isequal(liner_states(j,:),last(i,:))
        S(j)=S(j)+1;
    end
end
end
mp2=S/sum(S)
```

The main part of the simulation function *Simplified_Schelling2L* is the same as *Simplified_Schelling2*, but the function that calculates the number of unhappy people is adjusted, and the part related to the number is deleted:

```
function [last,status,unhappy] = Simplified_Schelling2L(n,in,e,T)
……
unhappy(1)=calculate2L(in,n);
status=[in;zeros(T,n)];

for t=1:T
……
    unhappy(t+1)=calculate2L(status(t+1,:),n);
……
end
last=status(T+1,:);
end

function unhappy = calculate2L(status,n,white)
ss=ones(1,n);
for i=1:length(white)
w=find(status==white(i));
ss(w)=0;
```

```
end
……
End
```

The calculation results are shown in the table below:

ϵ	Methods	L_1 (u=0)	L_2 (u=4)	L_3 (u=2)	L_4 (u=2)	L_5 (u=4)	L_6 (u=0)
0	Numerical	-	-	-	-	-	-
	Montecarlo	0.4963	0	0	0	0	0.5037
0.05	Numerical	0.4524	0.0128	0.0348	0.0348	0.0128	0.4524
	Montecarlo	0.4515	0.0110	0.0342	0.0359	0.0138	0.4536
0.1	Numerical	0.4091	0.0263	0.0646	0.0646	0.0263	0.4091
	Montecarlo	0.4061	0.0295	0.0641	0.0671	0.0287	0.4045
0.2	Numerical	0.3333	0.0556	0.1111	0.1111	0.0556	0.3333
	Montecarlo	0.3342	0.0564	0.1105	0.1125	0.0552	0.3312
0.5	Numerical	0.1667	0.1667	0.1667	0.1667	0.1667	0.1667
	Montecarlo	0.1660	0.1666	0.1698	0.1671	0.1665	0.1640
0.8	Numerical	0.0556	0.3333	0.1111	0.1111	0.3333	0.0556
	Montecarlo	0.0524	0.3425	0.1115	0.1058	0.3291	0.0587
1	Numerical	-	-	-	-	-	-
	Montecarlo	0	0.5088	0	0	0.4912	0

It can be seen that in a stationary distribution, as ϵ increases, the probability of S_1 and S_6 in the state of u=0 decreases, while the probability of S_2 and S_4 in the state of u=4 increases. This means that if these agents have a higher probability of

exchanging incorrectly, then they are more likely to place themselves in a lower "happiness" situation.

Note that when $\epsilon=0.2$ and $\epsilon=0.8$, the results of numerical calculations show that in a stationary distribution, the frequency of occurrence of u=0 and u=4 states are exchanged. The two extreme cases of $\epsilon=0$ and $\epsilon=1$ are absorbing chains, and the states they finally absorb are the states with the highest happiness index (u=0) and the lowest happiness index (u=4). Since there are two eigenvalues of 1, there is no numerical calculation result. This may explain that $\epsilon=0$ and $\epsilon=1$ represent the two extremes, and the more "orderly", the more "happy".

Discuss reasonable extensions of this model that would allow for richer, and perhaps more realistic dynamics, while keeping tractability at hand. What can you conclude from this exercise?

We can extend the ring model in part b to preserve the possibility of swapping by mistake. After that, consider an “assimilation” case: for each agent whose surrounding neighbour category is different from it, there will be a small probability η, let this agent change its category.

This “assimilation” process occurs simultaneously after the exchange, and there is no order. For example, A1 with category “0” is affected by the assimilation of surrounding neighbours. After the current time changes to category “1”, his neighbour A2(must be category “1”) will not immediately realize that the category of A1 has changed. Therefore, when A2 judges whether it will be “assimilated”, it still regards A1 as category “0”. Taking n=4 as an example, 10,000 simulations were carried out using Monte Carlo, and the time for each simulation was 100. The number of different categories in the initial state will no longer be limited by half of each. The final result shows that in 10,000 simulations, there are 4,825 times converged to the state [0,0,0,0,0,0], and 4,781 times converged to the state [1,1,1,1,1,1]. The rest 394 results converged to different other states. Similarly, trying n=5,6,7,8,9,10 is also a similar situation: in most cases, it converges to the same state of all agents.

This may indicate that, in a particular social circle of friends, if people interact regularly and have the possibility to accept the assimilation of others in certain aspects (such as hobbies, learning status, etc.). After a long period, this circle of

friends is likely to become a circle with common characteristics. In other words, birds of a feather flock together.