5-1-2012

Simulated trading in a two-good economy on a network

Ke Shi

Rochester Institute of Technology

School of Mathematical Sciences

Applied & Computational Mathematics Program

Master's Thesis

Applicant's Name: Ke Shi

Proposed Defense Date: 05/18/2012

Advisor's Name: Bernard Brooks

Committee Member 1: Tony Harkin

Committee Member 2: Christopher Wahle

Graduate Program Director: Dr. Tamas Wiandt

Simulated Trading in a Two-Good Economy on a Network

A thesis presented

by

Ke Shi

to

The School of Mathematical Sciences

in partial fulfillment of the requirements

for the degree of

Master's of Science

in the subject of

Applied and Computational Mathematics

Rochester Institute of Technology

Rochester, New York

May 2012

Abstract

With the upturn in living standards and increasing work stress, people have started to pay more attention to how happy they feel in their daily lives. Evaluating people's happiness is becoming increasingly important in scientific fields. The focus of this thesis is to build a rational happiness function and to do simulations on several kinds of social networks related to the real world in a trading model with only two types of goods but no common currency. The purpose of the simulations is to seek the equilibrium points of the trading system. For this model, the parameters in the function and the shape of networks are key factors that affect the experimental results.

Content

Introduction

In this thesis, we derive and analyze the happiness function, the value of which describes how happy a person is. We have built our system seeking the equilibrium point at which people can't become any happier. The happiness function is based on a normal trading system with only two types of goods: firewood and chocolate bars.

Our major work is focusing on the happiness function that we created ourselves,

$$H = k \times \left(1 - e^{-a \times F - b \times B}\right) + (1 - k) \times \left(\frac{P_f \times F + P_b \times B}{M + P_f \times F + P_b \times B}\right)$$

which will be derived in section 1.2.

At each iteration of the trading algorithm, a connection in the network is chosen uniformly at random from the set of possible connections. One person in that connection is arbitrarily called person 1 and the other person 2. Before doing any simulations, we figured out the condition where a trade occurs in a nice form. The proof is shown in section 2.3.

To get better results, we show a few proofs and do some experiments on a couple basic networks to determine appropriate parameters in the happiness function before simulations.

Consider that there are different kinds of social networks in the real world. We do simulations on 4 kinds networks: complete graphs corresponding to the world in which people are free to trade

with anyone else in the system, graphs with same number of vertices but different degree sums corresponding to the worlds with same amount of people ,but the people in each network are not equally well connected, the graph consists of two complete graphs connected by several bridges corresponding to the situation that two free worlds in which only some people are allowed to trade with people in the other world and 4-regular graphs corresponding to the world in which everyone has four adjacencies.

In the simulations of those four networks, we are looking at “Average Network Happiness (ANH)” instead of the happiness of each person in the network. As we know, only one pair of connected people is picked and one of the two or neither of them gets happier at each iteration. The ANH either goes up a little or stays the same at each iteration. The ANH stays the same if the two people decide not to trade. By looking at it we could easily follow the difference made at each step.

The simulations results in a series of plots which helped us to answer which factor determines how long it takes the equilibrium to occur, how happy people could be in the network when it hits the equilibrium point and various other questions.

Chapter 1 The Trade Model

1.1 Building the trade system

Consider the scenario in which people trade goods with each other in order to increase their own happiness. We build our system seeking the point at which people can't be any happier. The happiness function is based on a normal trading system.[1] There are only two types of goods: firewood and chocolate bars. In the system, there is no common currency. People trade goods for goods by some ratio. The total amount of each kind of goods in the entire system doesn't change. The structure of the network can be diversified regarding that people in groups are connected in distinct ways in the real world.

1.2 Define the happiness function

Individuals in the network are seeking to increase their happiness through trade. The happiness function's value shows how happy a person is. The bigger the value, the happier the person. Since there is no common currency in the system, the price of firewood ($\boldsymbol{P_f}$) and the price of chocolate bars ($\boldsymbol{P_b}$) are not evaluated by currency units. Usually firewood is considered to be cheaper. We initially set the price of firewood ($\boldsymbol{P_f}$) to be 1 so that we could simply set the price of chocolate bars to be some integer bigger than 1 and the ratio of the two prices is reasonable in different cases.

We define the happiness function ourselves as the weighted average of two factors: one that measures the happiness gained through the utility of the goods, $\left(1-\mathrm{e}^{-a\times F-b\times B}\right)$, and one that measures the happiness gained through the exchange value of the goods, $\left(\frac{P_f\times F+P_b\times B}{M+P_f\times F+P_b\times B}\right)$.

Our happiness function is shown below,

$$H(F,B)=k\times\left(1-\mathrm{e}^{-a\times F-b\times B}\right)+(1-k)\times\left(\frac{P_f\times F+P_b\times B}{M+P_f\times F+P_b\times B}\right)$$

The variables:

F : The current number of firewood.

B : The current number of chocolate bars.

P_f : The price of firewood.

P_b : The price of chocolate bars.

M : The initial total value of goods.

k : A constant on [0,1].

a : A constant measuring how the number of firewood affects the utility value of the function.

b : A constant chosen randomly around $(\boldsymbol{P_b}\,/\,\boldsymbol{P_f})\times\boldsymbol{a}$ measuring how the number of chocolate bars affects the utility value of the function and b is fixed through all trading attempts.

1.2.1 The range of the function

As we can see in the utility part, $\mathrm{e}^{-a\times F-b\times B}$ goes to 0 as ***F*** or ***B*** goes to infinity. And it goes to 1 as both ***F*** and ***B*** go to 0. That is to

say $e^{-a\times F-b\times B}$ ranges from 0 to 1. Then, $1-e^{-a\times F-b\times B}$ ranges from 0 to 1 as well. Now looking at the second part, the numerator is the total exchange value of goods someone has and the denominator is the sum of the numerator and the initial total value of goods of this person. We notice that the numerator is always less than the denominator. As the numerator which is the total value of goods someone has, goes to 0, the whole part $\left(\frac{P_f\times F+P_b\times B}{M+P_f\times F+P_b\times B}\right)$ goes to 0. However, when the numerator goes to infinity so that ***M,*** which is fixed when we first build the system, is negligible and $\left(\frac{P_f\times F+P_b\times B}{M+P_f\times F+P_b\times B}\right)$ goes to 1. Therefore, both the two parts go from 0 to 1. And take a weighted average by multiplying these two parts by ***k*** and 1-***k*** which are both on [0, 1] and adding them together, we could make the happiness function range from 0 to 1. The bigger ***k*** is, the more important the utility value is considered. For instance, for someone not concerned with the future trading value, ***k*** is large while for person who doesn't care about the utility of the goods, ***k*** is small.

1.2.2 Economic explanation of *a* and *b*

Even though the happiness function is evaluated objectively and subjectively, the two parts of the function are not entirely independent. The ratio of how much a piece of firewood contributes to the gain of the utility function value and how much a

chocolate bar does should be related to the ratio of their price. The ratios are almost the same in our system. The only difference is that ***b*** is around $\frac{\mathrm{P_b}}{\mathrm{P_f}} \times a$ but not the exact number. In the real world, one thing could have different useful value to different people. For example, chocolate bars mean more to someone who personally loves chocolate than it does to others.

Chapter 2 Trading Condition

Our purpose is that everyone gets happier by trading. Eventually after some trades no one in the system could get happier. That is to say, people stop trading when the system hits an equilibrium point.

2.1 Economic equilibrium

In economics, economic equilibrium is a state of the world where economic forces are balanced and in the absence of external influences the (equilibrium) values of economic variables will not change.[2] It is the point at which quantity demanded and quantity supplied is equal. Market equilibrium, for example, refers to a condition where a market price is established through competition such that the amount of goods or services sought by buyers is equal to the amount of goods or services produced by sellers. This price is often called the equilibrium price or market clearing price and will tend not to change unless demand or supply change.

2.2 The network

People can only trade goods with people that they are connected to. Thus before any trading happens a network that determines which people could each individual trade with must be created.

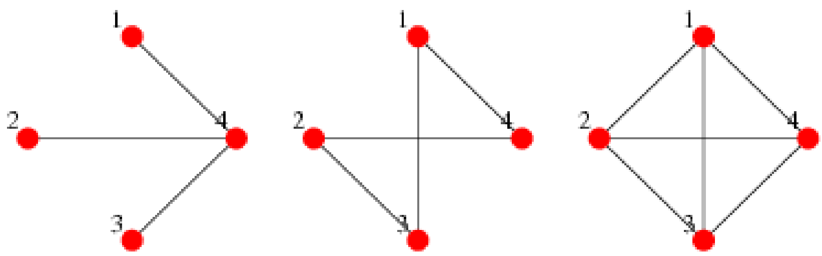
A social network can be seen as a graph consisting of vertices and edges. Everyone in the network is a vertex and there is an edge

if two people are connected. The edges are undirected. Individuals can only trade with people that are connected to them. Some people may be well connected. Those people are considered to be near the “center” of the network. However, some have few adjacencies. Some are even isolated (Such people never trade so that their happiness always remains the same).

In mathematics and computer science, an adjacency matrix is a means of representing which vertices (or nodes) of a graph are adjacent to which other vertices. It is a matrix **A** where,

$$\mathrm{a_{ij}} = \begin{cases} 1 \text{ , if i and j are adjacent} \\ 0 \text{ , if i and j are not adjaccent} \end{cases} \qquad \text{i, j are integers and i} \neq \text{j}$$

Noticing that the diagonal entries $\mathrm{a_{ii}}$ are always 0 since individuals are considered not connected to themselves. Here are some examples of networks and their corresponding adjacency matrix,



$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

2.3 The proof of trading condition

At each iteration, a connection in the network is chosen

uniformly at random from the set of possible connections. One person in that connection is arbitrarily called person 1 and the other person2. Without loss of generality person 1 trades firewood and gets chocolate bars. Thus person 2 receives firewood and gives chocolate bars. The happiness function is shown below,

$$\mathrm{H(F,B)} = \mathrm{k} \times \left(1 - \mathrm{e}^{-\mathrm{a}\times\mathrm{F}-\mathrm{b}\times\mathrm{B}}\right) + (1-\mathrm{k}) \times \left(\frac{\mathrm{P_f} \times \mathrm{F} + \mathrm{P_b} \times \mathrm{B}}{\mathrm{M} + \mathrm{P_f} \times \mathrm{F} + \mathrm{P_b} \times \mathrm{B}}\right)$$

H is a function of ***F*** and ***B*** and the gradient of **H** will point in the direction of greatest increase of happiness. So we have the gradient of happiness function of two people picked are,

$$\overrightarrow{\mathrm{G_1}} = \begin{bmatrix} \mathrm{k} \times \mathrm{a} \times \mathrm{e}^{-\mathrm{a}\times\mathrm{F_1}-\mathrm{b}\times\mathrm{B_1}} + (1-\mathrm{k}) \times \dfrac{\mathrm{M_1} \times \mathrm{P_{f_1}}}{\left(\mathrm{M_1} + \mathrm{P_{f_1}} \times \mathrm{F_1} + \mathrm{P_{b_1}} \times \mathrm{B_1}\right)^2} \\ \mathrm{k} \times \mathrm{b} \times \mathrm{e}^{-\mathrm{a}\times\mathrm{F_1}-\mathrm{b}\times\mathrm{B_1}} + (1-\mathrm{k}) \times \dfrac{\mathrm{M_1} \times \mathrm{P_{b_1}}}{\left(\mathrm{M_1} + \mathrm{P_{f_1}} \times \mathrm{F_1} + \mathrm{P_{b_1}} \times \mathrm{B_1}\right)^2} \end{bmatrix}$$

$$\overrightarrow{\mathrm{G_2}} = \begin{bmatrix} \mathrm{k} \times \mathrm{a} \times \mathrm{e}^{-\mathrm{a}\times\mathrm{F_2}-\mathrm{b}\times\mathrm{B_2}} + (1-\mathrm{k}) \times \dfrac{\mathrm{M_2} \times \mathrm{P_{f_2}}}{\left(\mathrm{M_2} + \mathrm{P_{f_2}} \times \mathrm{F_2} + \mathrm{P_{b_2}} \times \mathrm{B_2}\right)^2} \\ \mathrm{k} \times \mathrm{b} \times \mathrm{e}^{-\mathrm{a}\times\mathrm{F_2}-\mathrm{b}\times\mathrm{B_2}} + (1-\mathrm{k}) \times \dfrac{\mathrm{M_2} \times \mathrm{P_{b_2}}}{\left(\mathrm{M_2} + \mathrm{P_{f_2}} \times \mathrm{F_2} + \mathrm{P_{b_2}} \times \mathrm{B_2}\right)^2} \end{bmatrix}$$

We make the rule that the first time person 1 sells one firewood to get some amount of chocolate bars, say $\boldsymbol{P_m}$ which is the market price. The trading vector of person 1 is,

$$\overrightarrow{\mathrm{T_1}} = [-1 \quad \mathrm{P_m}]$$

And the trading vector of person 2 is,

$$\overrightarrow{\mathrm{T_2}} = [1 \quad -\mathrm{P_m}]$$

In order to make person 1 not less happy, at least we need,

$$\overrightarrow{T_1} \times \overrightarrow{G_1} = 0$$

Then we have,

$$-G_{11} + P_m \cdot G_{12} = 0$$

Therefore,

$$P_m = \frac{G_{11}}{G_{12}}$$

Now we want that person 2 could be happier by doing the trade, which requires

$$\overrightarrow{T_2} \times \overrightarrow{G_2} > 0$$

Then,

$$G_{21} - P_m \cdot G_{22} > 0$$

Furthermore we substitute in P_m,

$$G_{21} - \frac{G_{11}}{G_{12}} \cdot G_{22} > 0$$

Since G_{12} is positive, we have

$$G_{21} \cdot G_{12} - G_{11} \cdot G_{22} > 0$$

Therefore we find the condition required to do the trade,

$$\begin{vmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{vmatrix} < 0$$

The next time person 1 loses ***P_m*** chocolate bars to get a firewood so that the distance someone goes along the gradient would not be too far. Now the trading vector of person 1 is,

$$\overrightarrow{T_1} = [1 \quad -P_m]$$

And the trading vector of person 2 is,

$$\overrightarrow{T_2} = [-1 \quad P_m]$$

In order to make person 1 not less happy, at least we need,

$$\overrightarrow{T_1} \times \overrightarrow{G_1} = 0$$

Then we have,

$$G_{11} - P_m \cdot G_{12} = 0$$

Therefore,

$$P_m = \frac{G_{11}}{G_{12}}$$

Now we want person 2 could be happier by doing the trade, which requires

$$\overrightarrow{T_2} \times \overrightarrow{G_2} > 0$$

Then,

$$-G_{21} + P_m \cdot G_{22} > 0$$

Furthermore we substitute in P_m,

$$G_{21} - \frac{G_{11}}{G_{12}} \cdot G_{22} < 0$$

Since G_{12} is positive, we have

$$G_{21} \cdot G_{12} - G_{11} \cdot G_{22} < 0$$

Therefore we find the condition required to do the trade,

$$\begin{vmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{vmatrix} > 0$$

The next time person 1 sells one chocolate bar again.

Thus at each iteration a connection in the network is chosen, the

two people are arbitrarily labeled person 1 and person 2, and then person 1 tries to trade one firewood for P_m chocolate bars. We check to see if the trading conditions are met and if so, the trade is made. For the next iteration, a connection is again chosen randomly from the network and the trading conditions checked.

Chapter 3 Determine the parameters

To get better results, we are showing a few proofs and doing some experiments on a couple basic networks before simulations.

3.1 Determine *a* and *b*

In the real world, usually the utility is normally directly proportional to the market price (exchange value) of goods. In our system, that is to say,

$$\frac{a}{b} = \frac{P_f(\text{the market price of firewood})}{P_b(\text{the market price of chocolate bars})}$$

We could intuitively understand the ratios are not supposed to be exactly the same because different people have different preferences and the same item can't be equally valuable to different individuals, but why?

Assume they are the same, say $\frac{a}{b} = \frac{P_f}{P_b} = \frac{1}{3}$. Now let's look at the happiness function again,

$$H = k \times \left(1 - e^{-a\times F - b\times B}\right) + (1-k) \times \left(\frac{P_f \times F + P_b \times B}{M + P_f \times F + P_b \times B}\right)$$

When a trade occurs, for person 1,

$$F \to F - 1$$

$$B \to B + \frac{1}{3}$$

Since,

$$\frac{a}{b} = \frac{P_f}{P_b} = \frac{1}{3}$$

We have,

$$-a \times (F - 1) - b \times \left(B + \frac{1}{3}\right) = -a \times F - b \times B$$

And,

$$P_f \times (F - 1) + P_b \times (B + \frac{1}{3}) = P_f \times F + P_b \times B$$

For person 2,

$$F \to F + 1$$

$$B \to B - \frac{1}{3}$$

Since,

$$\frac{a}{b} = \frac{P_f}{P_b} = \frac{1}{3}$$

We have,

$$-a \times (F + 1) - b \times \left(B - \frac{1}{3}\right) = -a \times F - b \times B$$

And,

$$P_f \times (F + 1) + P_b \times (B - \frac{1}{3}) = P_f \times F + P_b \times B$$

The above shows, in this case everyone in the network keeps doing the trades forever but their happiness stays the same.

Therefore, we are considering if we could set $\frac{a}{b}$ approximately equal to $\frac{P_f}{P_b}$. Set person 1's $\frac{a}{b} = \frac{1}{3.2}$ and person 2's $\frac{a}{b} = \frac{1}{2.9}$.

When trade happens, for person 1,

$$F \to F - 1$$

$$\mathrm{B} \rightarrow \mathrm{B}+\frac{1}{3}$$

Since,

$$\frac{\mathrm{a}}{\mathrm{b}}=\frac{1}{3.2}<\frac{\mathrm{P_f}}{\mathrm{P_b}}=\frac{1}{3}$$

We have,

$$-\mathrm{a} \times (\mathrm{F}-1)-\mathrm{b} \times \left(\mathrm{B}+\frac{1}{3}\right)<-a \times F-b \times B$$

And,

$$\mathrm{P_f} \times (\mathrm{F}-1)+\mathrm{P_b} \times (\mathrm{B}+\frac{1}{3})=\mathrm{P_f} \times \mathrm{F}+\mathrm{P_b} \times \mathrm{B}$$

Then, $\mathrm{k} \times \left(1-\mathrm{e}^{-\mathrm{a} \times \mathrm{F}-\mathrm{b} \times \mathrm{B}}\right)$ goes up, $(1-\mathrm{k}) \times \left(\frac{\mathrm{P_f} \times \mathrm{F}+\mathrm{P_b} \times \mathrm{B}}{\mathrm{M}+\mathrm{P_f} \times \mathrm{F}+\mathrm{P_b} \times \mathrm{B}}\right)$ stays the same. So H goes up.

For person 2,

$$\mathrm{F} \rightarrow \mathrm{F}+1$$

$$\mathrm{B} \rightarrow \mathrm{B}-\frac{1}{3}$$

Since,

$$\frac{\mathrm{a}}{\mathrm{b}}=\frac{1}{2.9}>\frac{\mathrm{P_f}}{\mathrm{P_b}}=\frac{1}{3}$$

We have,

$$-\mathrm{a} \times (\mathrm{F}+1)-\mathrm{b} \times \left(\mathrm{B}-\frac{1}{3}\right)<-a \times F-b \times B$$

And,

$$\mathrm{P_f} \times (\mathrm{F}+1)+\mathrm{P_b} \times (\mathrm{B}-\frac{1}{3})=\mathrm{P_f} \times \mathrm{F}+\mathrm{P_b} \times \mathrm{B}$$

Then, $k \times \left(1 - e^{-a \times F - b \times B}\right)$ goes up, $(1-k) \times \left(\frac{P_f \times F + P_b \times B}{M + P_f \times F + P_b \times B}\right)$ stays the same. So H goes up.

We can even concludes that the trading condition can be simplified as ,

$\frac{a_1}{b_1} \leq \frac{P_f}{P_b}$ and $\frac{a_2}{b_2} \geq \frac{P_f}{P_b}$, where $\boldsymbol{k}$, $\boldsymbol{e}$, $\boldsymbol{P_f}$, $\boldsymbol{P_b}$ are constants.

3.2 Monopoly pricing

A monopoly exists when a specific person or enterprise is the only supplier of a particular commodity or owns most of one kind of supply in the whole market. (This contrasts with a monopsony which relates to a single entity's control of a market to purchase a good or service, and with oligopoly which consists of a few entities dominating an industry)[3] Monopolies are thus characterized by a lack of economic competition to produce the good or service and a lack of viable substitute goods.[4] The verb "monopolize" refers to the process by which a company gains the ability to raise prices or exclude competitors. In economics, a monopoly is a single seller. Although monopolies may be big businesses, size is not a characteristic of a monopoly. A small business may still have the power to raise prices in a small industry (or market).

A monopolist can be a price maker and decides the price of the good or product to be sold. We applied this characteristic into our simulations. In our model, if someone in the system holds over 50%

of the total amount of one type of good in the network, he raises the price of his goods himself. The next time this person is picked to do a trade, the exchange values are no longer the initial ones.

3.3 Determine *k*

The parameter *k* controls the weighted average of the utility part and the exchange value part. What value we set *k* to be is based on which of those two parts contributes more to the value of the whole happiness function.

Case 1:

Sometimes there is no monopoly in our simulation. If this happens, we can see in the function,

$$M = P_f \times F + P_b \times B$$

from the very beginning to the end. Thus the exchange value part $\left(\frac{P_f \times F + P_b \times B}{M + P_f \times F + P_b \times B}\right)$ is always $\frac{1}{2}$ and it contributes nothing to the change of the value of happiness function. In this case we set 1-***k*** to be very small.

Case 2:

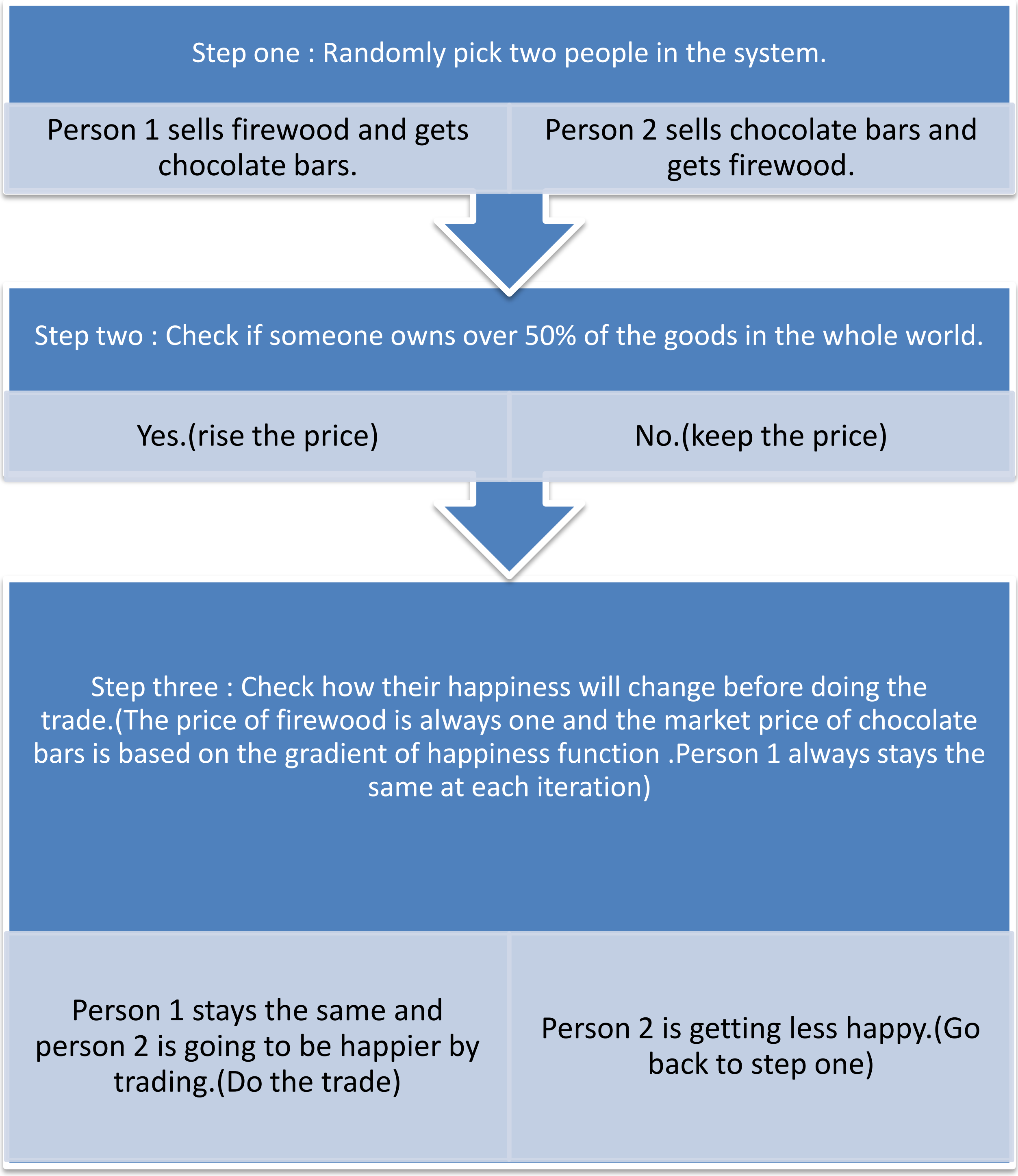
Sometimes the ratio of ***a*** and ***b*** are fixed by the ratio of $\boldsymbol{P_f}$ and $\boldsymbol{P_b}$. $\boldsymbol{P_f}$ and $\boldsymbol{P_b}$ do change, but they only change a few times after a monopoly appears. In this case, $-a \times F - b \times B$ keeps the same value through most steps of the iterations. Therefore, the utility part $\left(1 - e^{-a \times F - b \times B}\right)$ doesn't affect the value of happiness

function. We set $\boldsymbol{k}$ to be small.

Case 3:

In real world, all $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{P_f}$ and $\boldsymbol{P_b}$ differ rapidly. In this case, we should test to see what value we set $\boldsymbol{k}$ to be so that the change of **H** could be obvious. We test starting from 0.5 and begin our real simulation after $\boldsymbol{k}$ is determined.

Chapter 4 The flowchart of a trade



Chapter 5 Simulations

In real world, there are some objective factors that affect people's happiness. Now the network is set up. We made a few assumptions and did a couple of simulations to support our results. These simulations will address which factors determine how long it takes the equilibrium to occur, how happy people could be in the network when it hits an equilibrium and various other results.

Let's introduce in a new definition-"Average Network Happiness (ANH)". As we know, only one pair of people is picked and one of the two or neither of them gets happier at each step of the iteration. The ANH either goes up a little or stays the same at each step of the iteration. By looking at it we could easily follow the difference made at each step.

Before all the simulations on our own happiness function, lets introduce in a more standard utility function from economics,[5] where ***F*** and ***B*** are defined as same as the ones in our own happiness function.

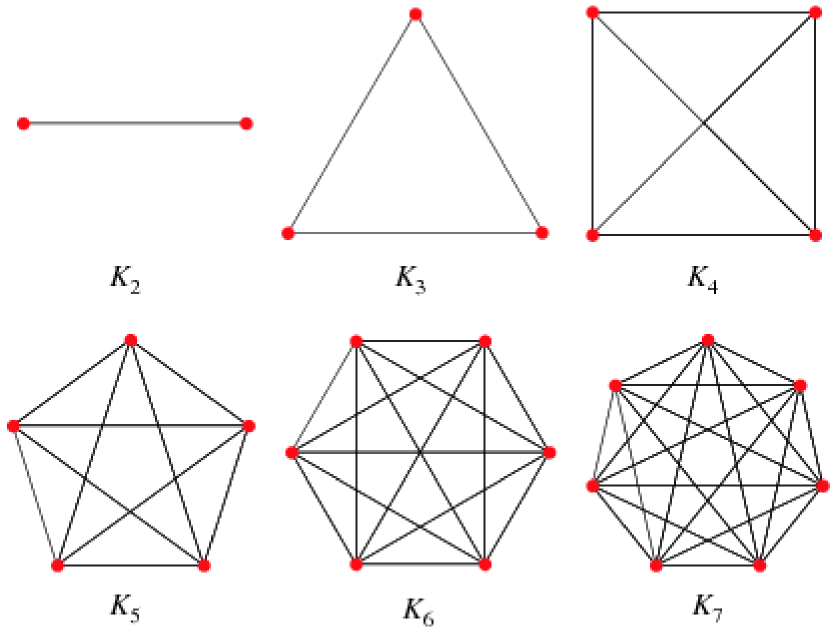
$$\mathrm{U(F, B)} = \mathrm{F}^{0.25} \times \mathrm{B}^{0.75}$$

We are about to apply several runs of our simulations using Matlab[6][7] on both the happiness function we defined ourselves and the one with the more standard utility function from economics.

5.1 Experiments on basic complete graphs

The first simulation is based on a complete graph[8]. A complete graph is a graph in which all the vertices are adjacent pairwise.

Examples of a complete graph are shown below ($\boldsymbol{K_i}$ is a complete graph with ***i*** vertices, ***i*** is an integer),[9]



5.1(a) examples of complete graphs

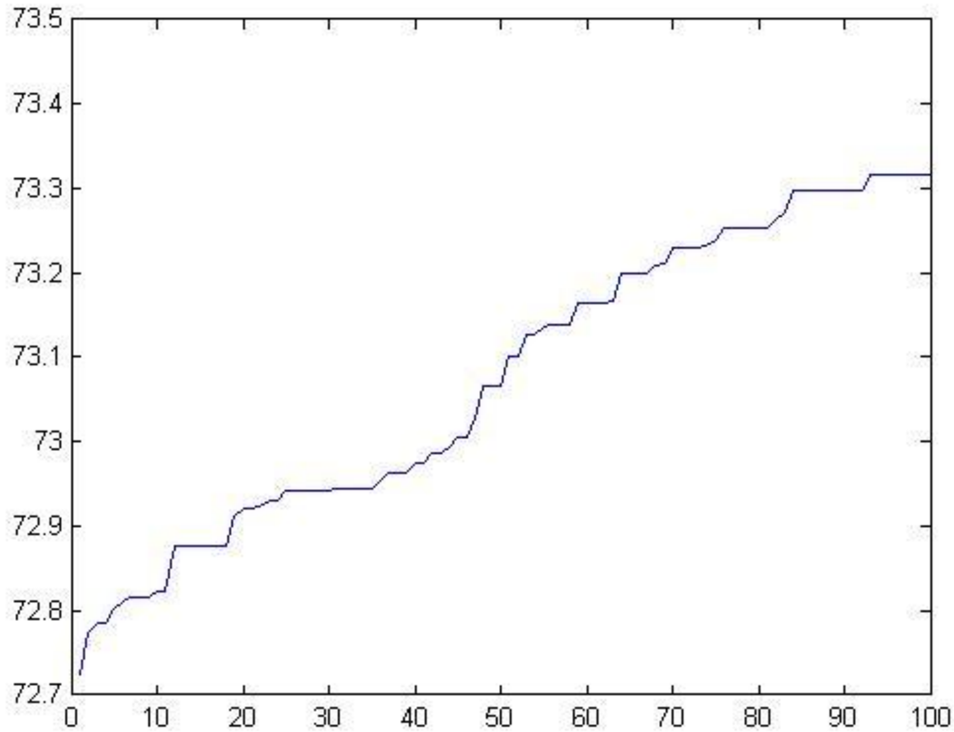
In the Matlab code, the number of firewood each person initially has is uniformly distributed between 1 and 120 and the number of chocolate bars each person initially has is uniformly distributed between 1 and 40.

First, we use the standard utility function from economics as part of the happiness function. We find that the range $\mathrm{F}^{0.25} \times \mathrm{B}^{0.75}$ is from 0 to approximately 100 and the range of exchange value part is from 0 to 1. Thus we take the weighted average of the two parts

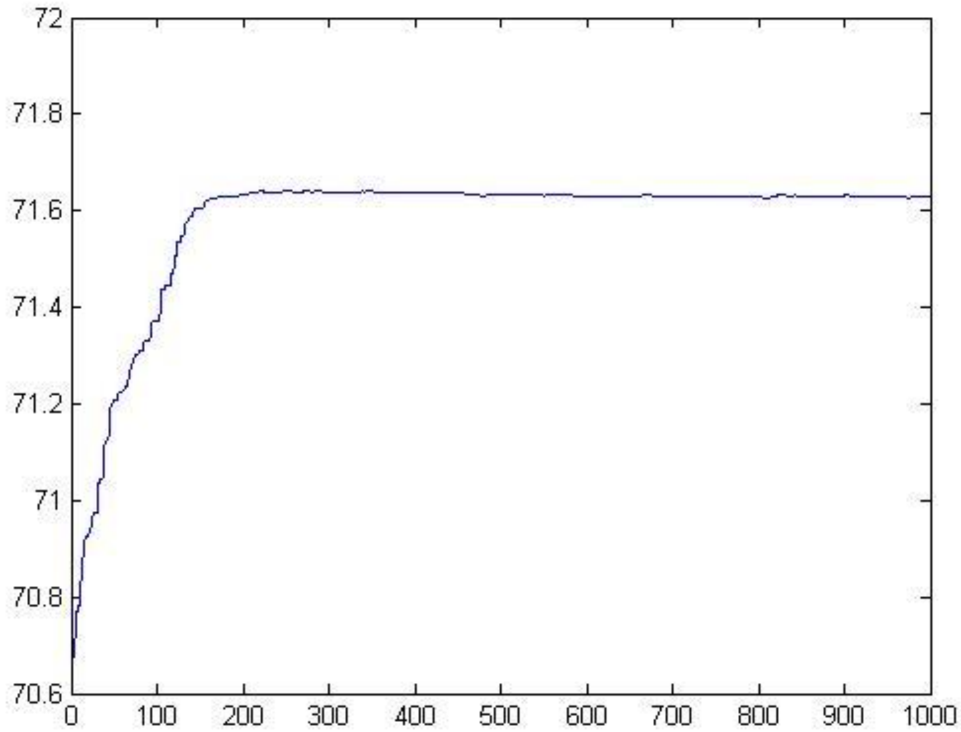
by multiplying $\left(\frac{P_f \times F + P_b \times B}{M + P_f \times F + P_b \times B}\right)$ by 100 so that the two parts are equally weighted. Then the happiness function becomes,

$$H(F, B) = F^{0.25} \times B^{0.75} + 100 \times \left(\frac{P_f \times F + P_b \times B}{M + P_f \times F + P_b \times B}\right)$$

Now apply the simulation of this happiness function on complete graphs, the ANH-T(trade attempts) plots are shown below,



5.1(b1) ANH-T plot of standard happiness function (N=100, n=8) on a complete graph

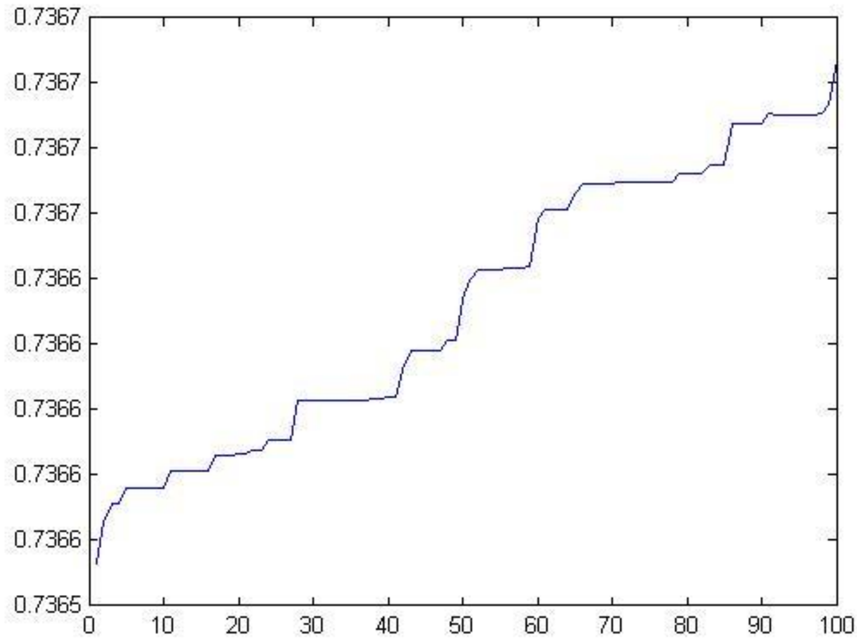


5.1(b2) ANH-T plot of standard happiness function (N=1000, n=8) on a complete graph

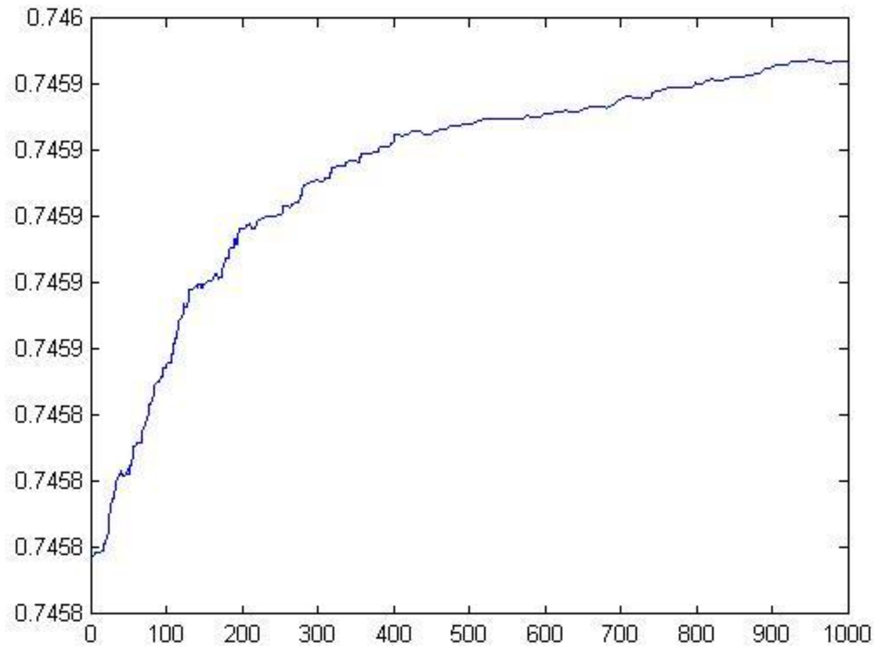
Then we are going to do simulations on our happiness function discussed in previous chapters. In the Matlab[10] code, b is $3 \times \mathrm{a} + \mathrm{b}'$ where b' is uniformly distributed between $0.001 \times \mathrm{b}$ and $0.1 \times \mathrm{b}$. For each simulation, we could apply different ***k*** , ***n*** which is the number of individuals in the whole network ,***a*** and ***N*** which is the number of iterations.

When ***n***=8, ANH is compared to the number of iterations or trade attempts, ***a***=0.04, ***k***=0.5. We did experiments with different values of ***N*** seeking to find a sufficient ***N*** that we can use for the following simulations. The ANH-T plots on a complete graph are

shown below.

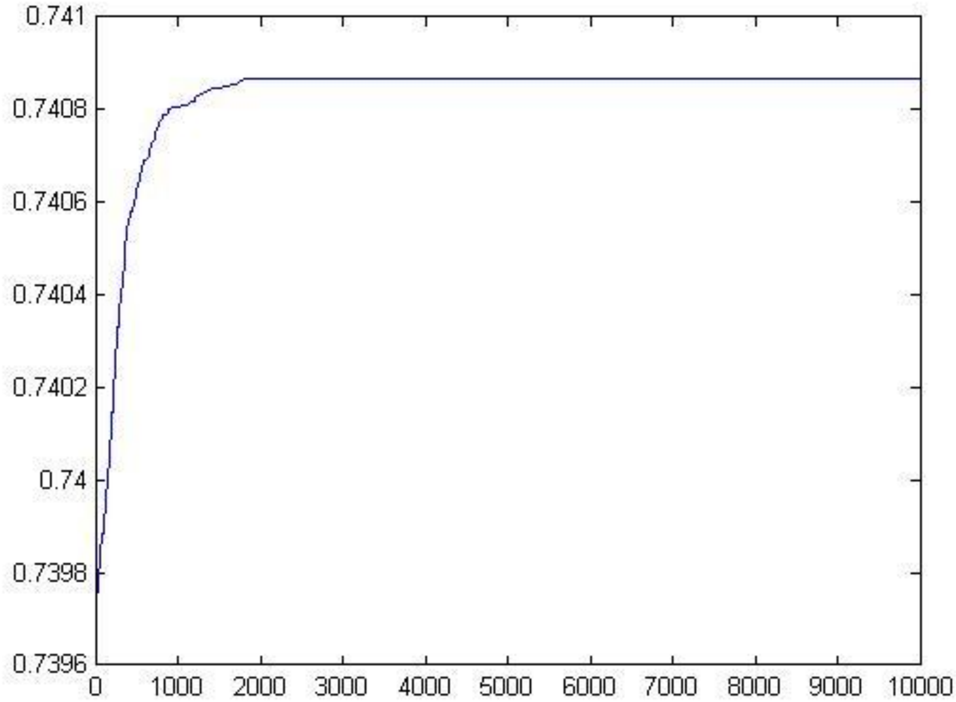


5.1(c1) ANH-T plot of original happiness function (N=100, n=8, k=0.5, a=0.04) on a complete graph



5.1(c2) ANH-T plot of original happiness function (N=1000, n=8, k=0.5, a=0.04) on a

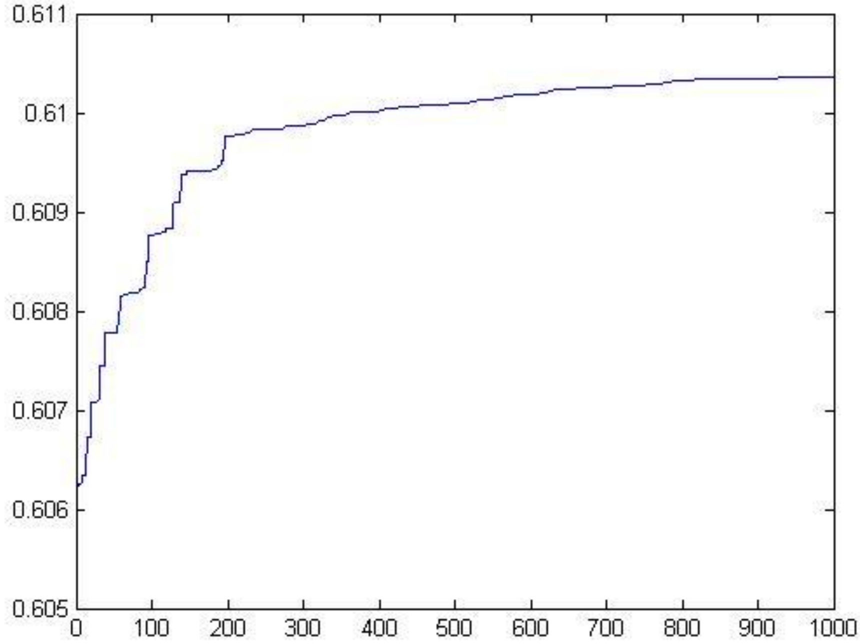
complete graph



5.1(c3) ANH-T plot of original happiness function (N=10000, n=8, k=0.5, a=0.04) on a complete graph

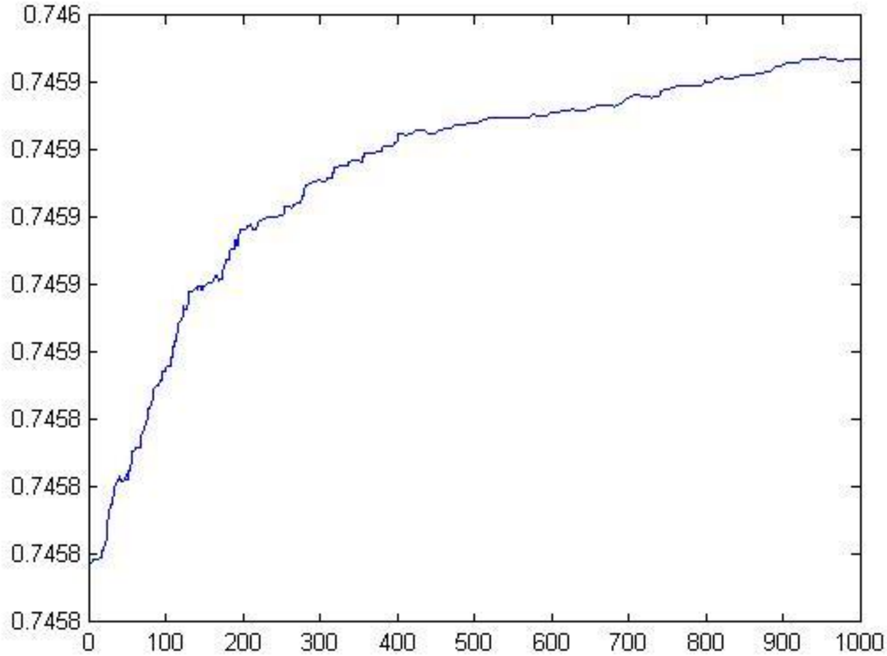
From the above plots, we see 1000 trade attempts are sufficient enough for a network with 8 individuals to find when the equilibrium occurs. Additionally we find the ANH doesn't change much through the whole process. We are considering if we could change the weights of the utility and the exchange value to make the range on ANH to be more obvious.

Thus we fix ***n***=8, ***a***=0.04 and ***N***=1000. As we change ***k***, the ANH-T plots vary.



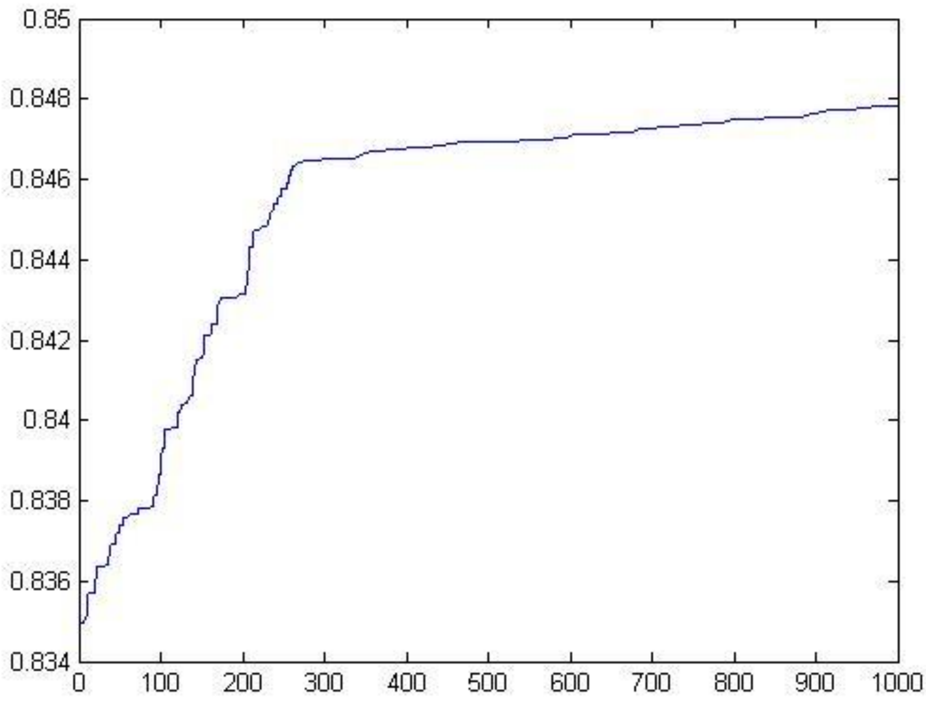
5.1(d1) ANH-T plot of original happiness function (N=1000, n=8, k=0.25, a=0.04)

on a complete graph



5.1(d2) ANH-T plot of original happiness function (N=1000, n=8, k=0.5, a=0.04)

on a complete graph



5.1(d3) ANH-T plot of original happiness function (N=1000, n=8, k=0.75, a=0.04) on a complete graph

These three plots tell the bigger the ***k*** is, the wider the range of ANH is. Therefore, we use ***k***=0.75 in the following simulations.

In conclusion, the simulations on our own happiness function work as well as those on the happiness function with standard utility function from economics except that our own happiness function converges slower. In following chapters we are going discuss simulations of our own happiness function on different types of networks.

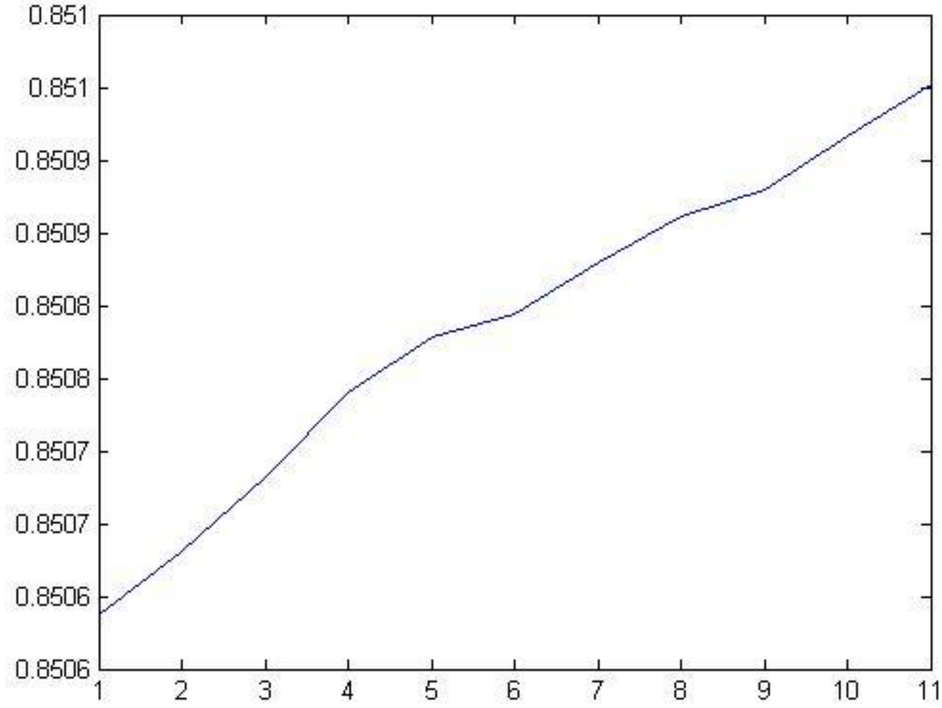
5.2 Networks with different degree sums

Once how wealthy people are is determined in a network, will they be happier if people are well connected together than rarely

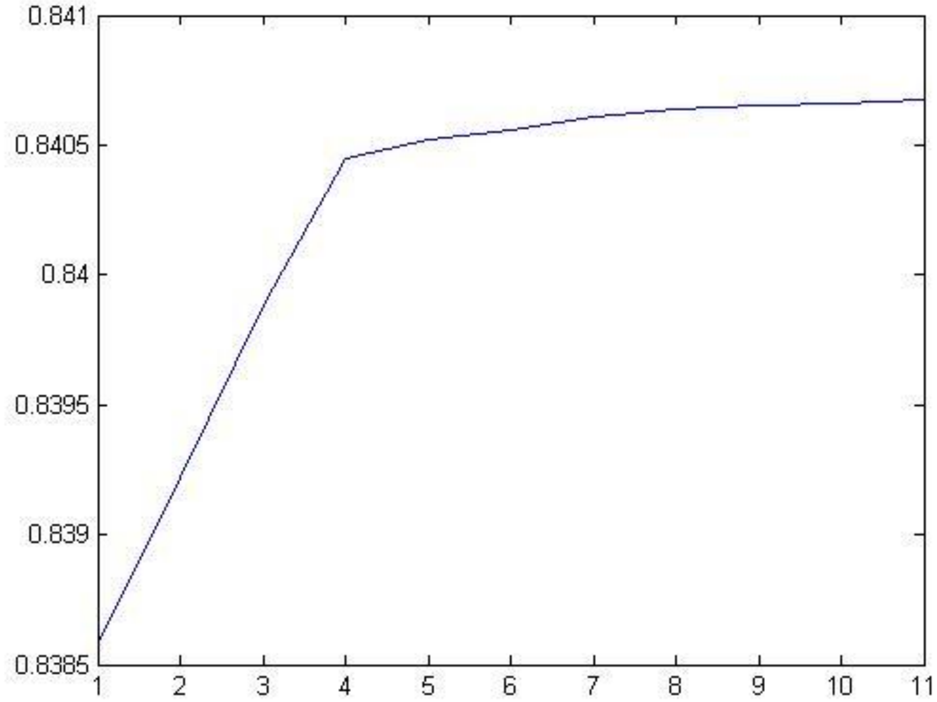
connected when equilibrium occurs? We think so.

Let's introduce in a few definitions first. In graph theory, the degree of a vertex of a graph is the number of edges incident to the vertex, with loops counted twice.[11] The degree of a vertex v is denoted $\deg(v)$. Given a graph $G=(V,E)$, the degree sum is $\sum_{v\in V}\deg(v)$. It is one way to represent how well people are connected in the network.

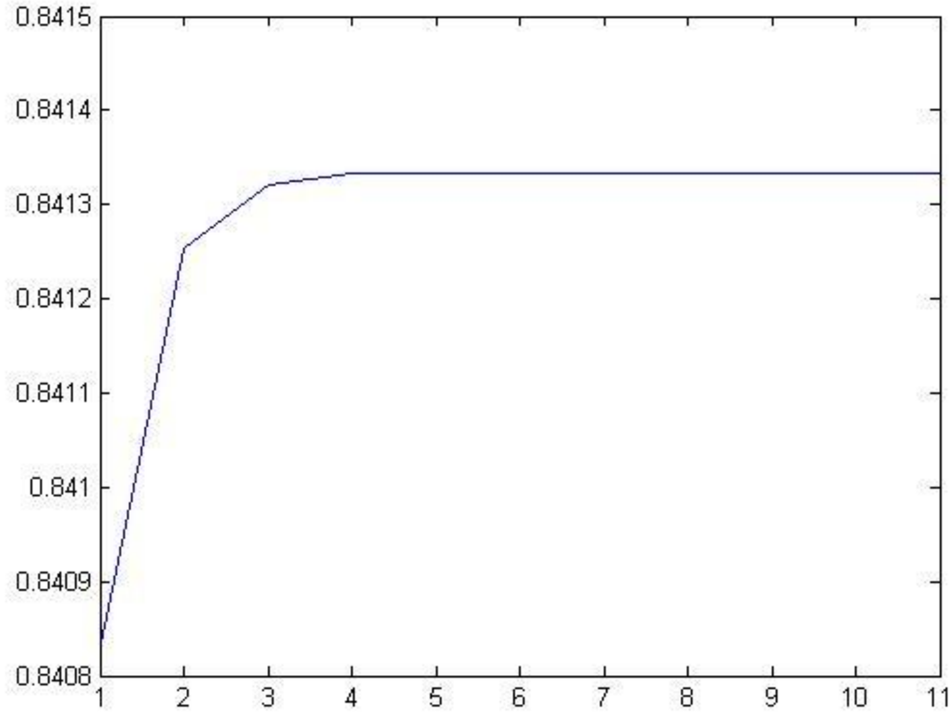
In the Matlab code, we use an index ***p*** to control the degree sum of a graph. ***p*** determines the probability that two vertices are adjacent. In the code, for each pair of individuals we generate a random integer ***X*** between 0 and 10000. If $X \le (\boldsymbol{p}-1)\times 1000$, set the corresponding element in the adjacency matrix to be 1 and all elements on diagonal are 0s. The bigger ***p*** is the better people are connected in the network. Notice that $1 \le \boldsymbol{p} \le 11$ so that the probability that two vertices are adjacent is between 0 and 1. When ***n***=16, ***k***=0.75, ***a***=0.04, ***N***=100 we get the ANH-***p*** plot, where ***x***-axis corresponds to ***p*** and ***y***-axis corresponds to ANH,



5.2(a1) ANH-p plot of original happiness function (N=100, n=16, k=0.75, a=0.04) on graphs of different degree sums



5.2(a2) ANH-p plot of original happiness function (N=1000, n=16, k=0.75, a=0.04) on graphs of different degree sums



5.2(a3) ANH-p plot of original happiness function (N=10000, n=16, k=0.75, a=0.04) on graphs of different degree sums

When ***N***=100, an equilibrium could hardly occur as we know from Fig. 5.1 The first 100 trade attempts are the ones through which the ANH grows quite fast. Figure 5.2(a) tells with more connections in a network, the ANH increases faster.

When $\boldsymbol{N} \geq 1000$, usually an equilibrium happens. Then combining Figs. 5.2(b) and 5.2(c) together, we can see the ANH is bigger at the equilibrium point with bigger ***p***. Furthermore, there is no significant change on ANH under the condition that $\boldsymbol{p} \geq 4$ where 30% of the possible bridges exist.

5.3 Complete graphs and 4-regular graphs

No doubt that the complete network is an ideal trading model.

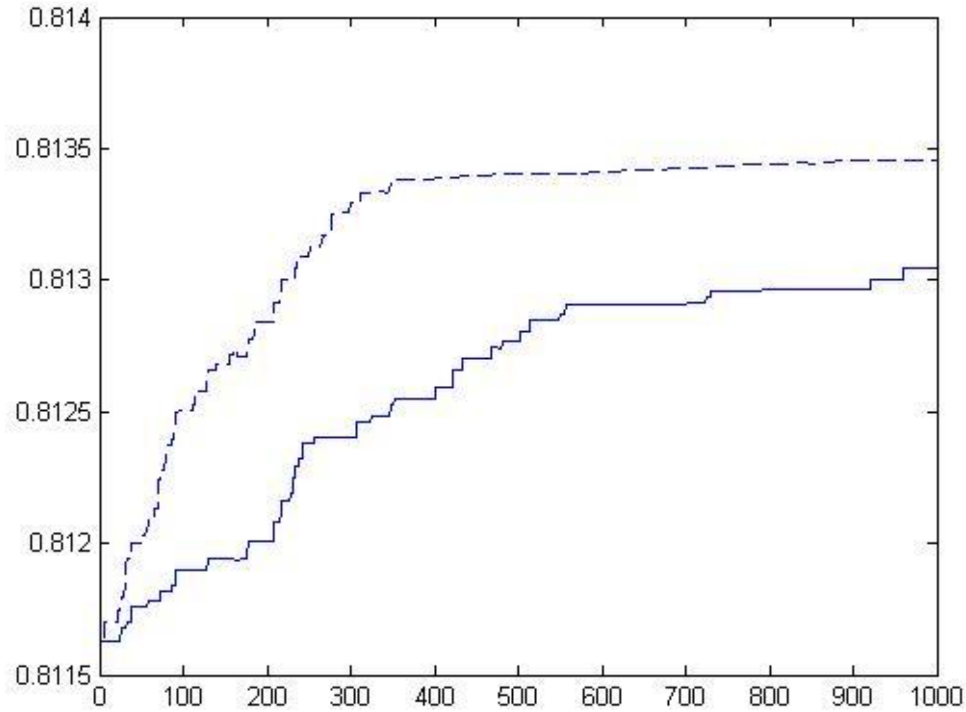
Usually the network is not that well connected. Then we start thinking if there are alternative kinds networks in which people feel free to trade with each other.

We are looking at regular graphs. In graph theory, a regular graph is a graph where each vertex has the same number of neighbors. A regular graph with vertices of degree ***k*** is called a ***k***-regular graph or regular graph of degree *k*.[12]

Our experiment is based on a 4-regular graph with 16 vertices. Its adjacency matrix is,

$$\begin{bmatrix}
0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0
\end{bmatrix}$$

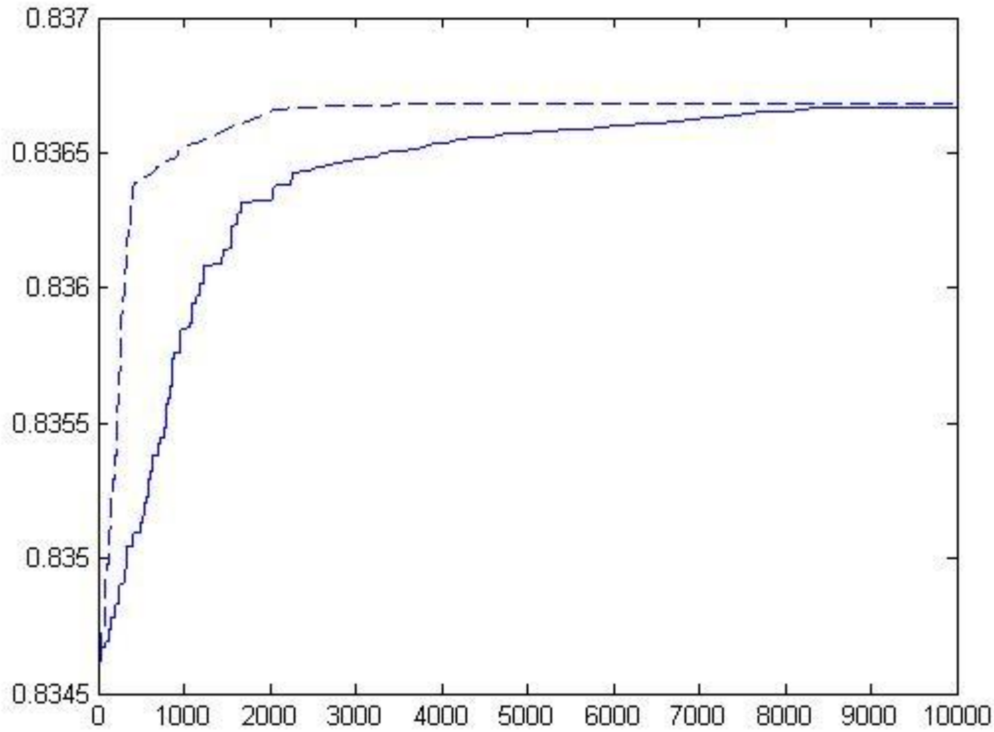
The ANH-T plot of this network will be compared with the one where people are completely connected,



5.3(a) ANH-T plots comparison between a complete graph and a 4-regular graph with same number of vertices (N=1000, n=16, k=0.75, a=0.04)

The line of dashes refers to the complete network and the other one refers to the regular network. The ANH of the complete network increases faster than the one of the regular graph, especially at the very beginning. Is it saying that people could be happier if they are better connected? The answer is NO.

We did the same experiment with more trade attempts, the plot goes as the following,



5.3(b) ANH-T plots comparison between a complete graph and a 4-regular graph with same number of vertices (N=10000, n=16, k=0.75, a=0.04)

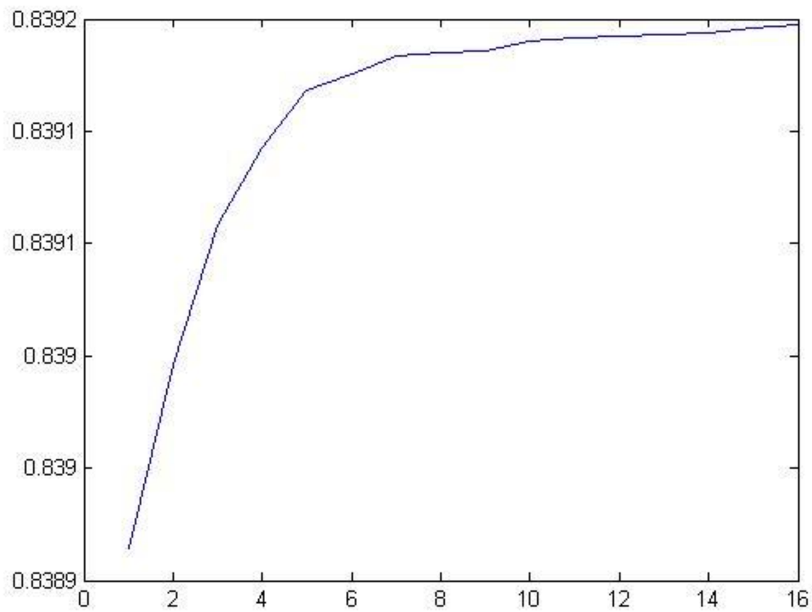
The result shows that the equilibrium of the complete network occurs much faster than the one of a regular network, especially by the first few trade attempts. However the equilibrium will happen at sooner or later. Additionally the final ANH will be the same. In either case, people will ultimately get the goods that they want, but it will take longer with less connections.

5.4 Two complete graphs connected by bridges

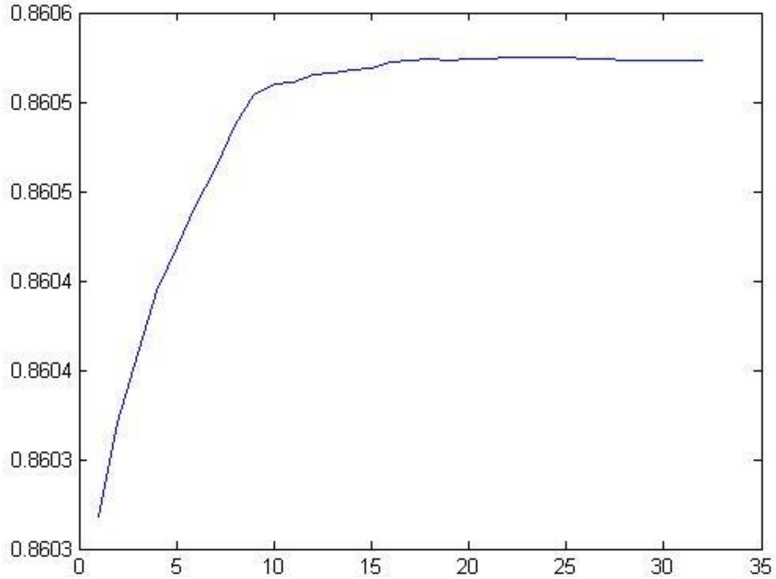
Imagine the trade between individuals in different countries in the real world. Of course people are free to trade with each other in the same country. However, only some people have the access to

trade with people overseas. Considering this situation, we did experiments on the networks which consist of two complete graphs with several bridges between them. We are focusing on the number of bridges to see how it affects the ANH.

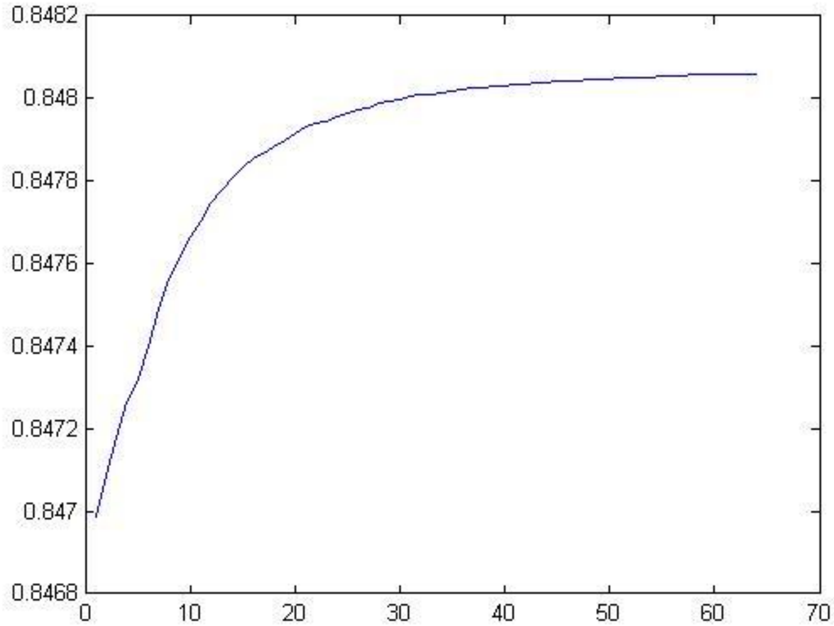
In the code, we build up two groups of individuals first. People in each group are completely connected. Then we add bridges between those two groups starting from one to the number of people in each group. For each number of bridges, try 1000 trade attempts and record the final ANH. When $\boldsymbol{n}$=32 where there are 16 people in each group figure 5.4(a1) results where the $\boldsymbol{x}$-axis corresponds to the number of bridges ($\boldsymbol{n_{bg}}$) and $\boldsymbol{y}$-axis corresponds to ANH.



5.4(a1) ANH- $\boldsymbol{n_{bg}}$ plot of original happiness function on graphs each consists of two complete graphs connected by bridges (N=1000, n=32, k=0.75, a=0.04)



5.4(a2) ANH- n_{bg} plot of original happiness function on graphs each consists of two complete graphs connected by bridges (N=1000, n=64, k=0.75, a=0.04)

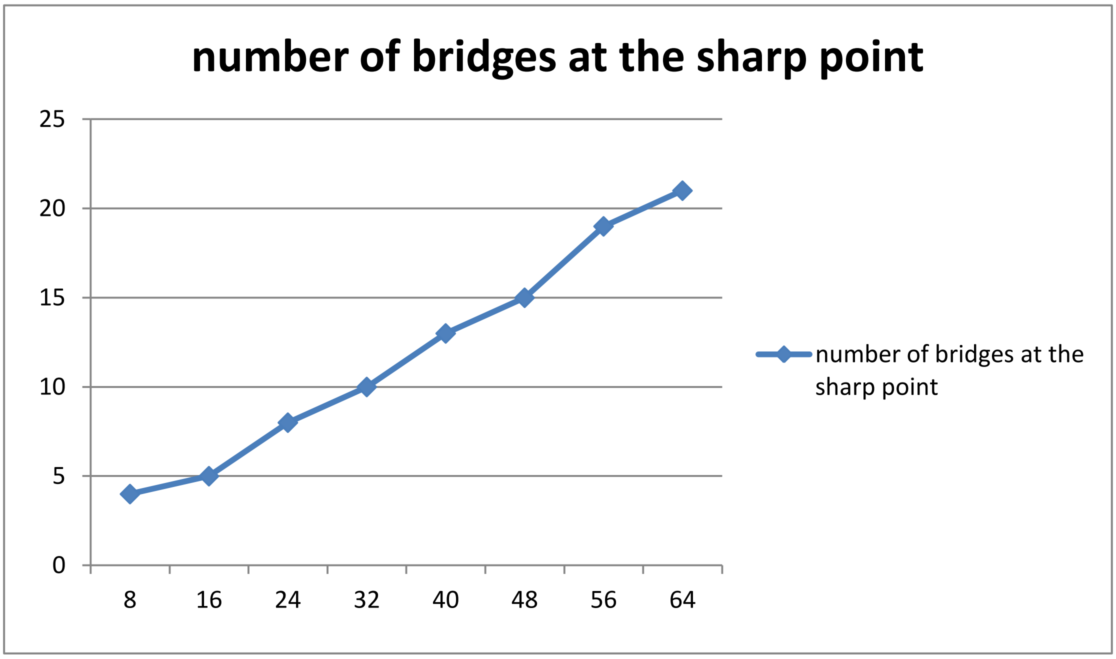


5.4(a3) ANH- n_{bg} plot of original happiness function on graphs each consists of two complete graphs connected by bridges (N=1000, n=128, k=0.75, a=0.04)

From the above three plots, we can see as the number of bridges goes up, the final ANH increases. And here we find something interesting. There is one point in each plot by which the ANH starts growing slowly. Thus we guess there is some relation between the number of bridges ($\boldsymbol{n_y}$) by which the final ANH doesn't change quite a lot and the number of people in each group, denoted $\boldsymbol{n_x}$. By doing several more experiments, we've got the data below,

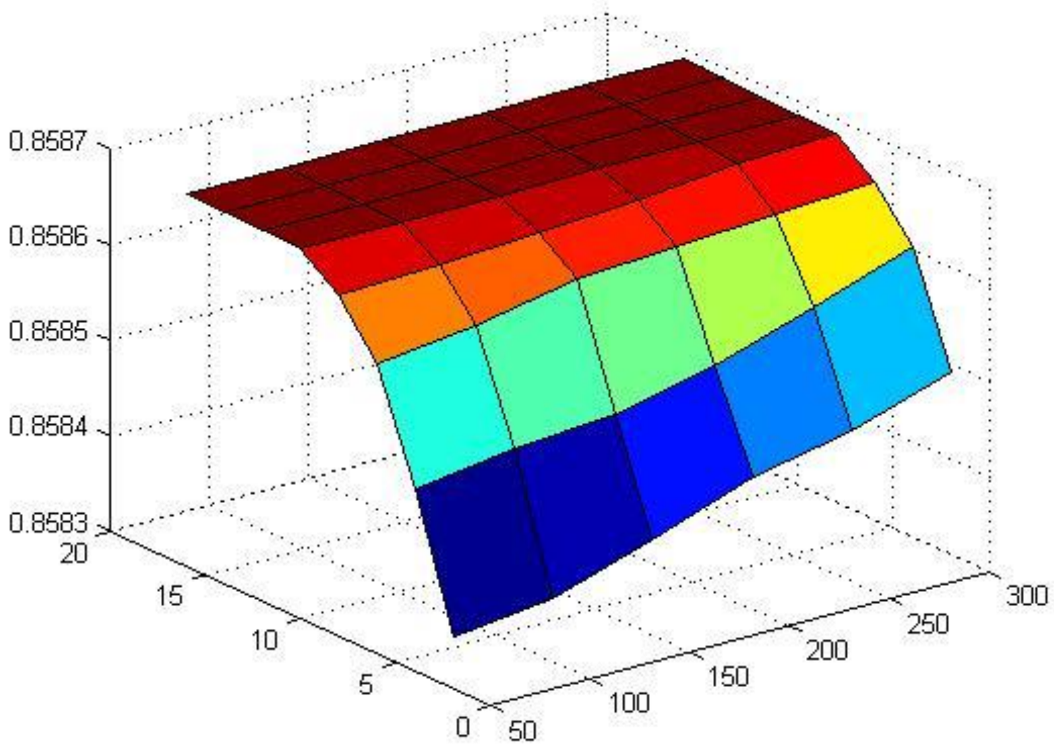
$\boldsymbol{n_x}$	$\boldsymbol{n_y}$
8	4
16	5
24	8
32	10
40	13
48	15
56	19
64	21

Here is the n_y-n_x plot,



From the plot we could tell the function from n_x to n_y is approximate to a linear function. That is to say, by knowing the number of people in each we could easily tell the exact number of necessary bridges to maximize the ANH.

After this, we consider two 16-node complete networks connected by 2, 4, 6, 8, 10, 12, 14, 16 bridges. For each case we ran 300 iterations and used the data to plot a 3 dimensional graph where y is the number of bridges, x is the iteration number (use 50, 100, 150, 200, 250, 300) and z is the average network happiness.



5.4(b) 3-D plots of original happiness function on graphs each consists of two complete graphs connected by bridges when the number of bridges and the iteration number both change (n=32, k=0.75, a=0.04)

It is visually obvious and intuitively understandable that the more bridges there are in the network, the faster the equilibrium reaches and the more steps of iteration, the larger the ANH.

Chapter 6 Conclusions

We make some conclusions about each kind of network.

First of all, after comparing the ANH-T plot of a complete graph of our original happiness function to the one of standard happiness function, we showed the simulations on our own happiness function work as well as those on the happiness function with standard utility function from economics except that our own happiness function converges slower.

We have also shown that the first 100 trade attempts are the ones through which the ANH grows fast and with more connections in a network, the ANH increases faster. Once how wealthy people are is determined in a network, the ANH at the equilibrium point is bigger with bigger ***p***. However, there is no significant change on ANH under the condition that over 30% of the possible connections exist.

From the experiment which compared the simulations on a 4-regular graph with a complete graph, we verified our initial guess that the equilibrium of the complete network occurs much faster than the one of a regular network, especially by the first few trade attempts. However the equilibrium will happen sooner or later. Additionally the final ANH will be the same.

In the situation that two complete graphs are connected by

several bridges, we were able to conclude that as the number of bridges goes up, the final ANH increases. Also there is one point corresponding to the number of people in each complete graph by which the ANH starts growing slowly. Furthermore, we showed that the relation between the number of bridges by which the final ANH doesn’t change quite a lot and the number of people in each group is approximately a linear function.

Reference

[1] “Modeling Price Pressure in Financial Markets” by Elena Asparouhova and Peter Bossaerts.

[2] Hal R. Varian, Microeconomic Analysis, Third edn. Norton, New York 1992

[3] Milton Friedman (2002). "VIII: Pokemon and the Social Responsibility of Business and Labor" (paperback). Capitalism and Freedom (40th anniversary edition ed.). The University of Chicago Press. p. 208. ISBN 0-226-26421-1.

[4] Blinder, Alan S; William J Baumol and Colton L Gale (June 2001). "11: Monopoly" (paperback). Microeconomics: Principles and Policy. Thomson South-Western. p. 212. ISBN 0-324-22115-0.

[5] Samuelson, Paul A. (1948). Economics: An Introductory Analysis McGraw-Hill

[6] Gilat, Amos (2004). MATLAB: An Introduction with Applications 2nd Edition. John Wiley & Sons. ISBN 978-0-471-69420-5.

[7] Quarteroni, Alfio; Fausto Saleri (2006). Scientific Computing with MATLAB and Octave. Springer. ISBN 978-3-540-32612-0.

[8] Gries, David; Schneider, Fred B. (1993), A Logical Approach to Discrete Math, Springer-Verlag, p. 436 .

[9] Pirnot, Thomas L. (2000), Mathematics All Around, Addison

Wesley, p. 154, ISBN 9780201308150.

[10] Ferreira, A.J.M. (2009). MATLAB Codes for Finite Element Analysis. Springer. ISBN 978-1-4020-9199-5.

[11] Diestel, Reinhard (2005), Graph Theory (3rd ed.), Berlin, New York: Springer-Verlag, ISBN 978-3-540-26183-4.

[12] Chen, Wai-Kai (1997). Graph Theory and its Engineering Applications. World Scientific. pp. 29. ISBN 9789810218591.

Appendix

Main Matlab code:

Chapter 5.1

```
function val = equipstd(n,N)
% Define the adjacent matrix A.
for i=1:N
    for j=1:N
        if i==j
            A(i,j)=0;
        else
            A(i,j)=1;
        end
    end
end

%Price of goods
for i=1:N
    Pfo(i)=1;
    Pbo(i)=3;
    %Initial number of goods for each person
    F(i)=randi(120,1);  %firewood
    B(i)=randi(40,1);%candybars
    b(i)=randi(100,1);
end
d=0.5;%dominant ratio
%Initial value of goods for each person
for i=1:length(F)
    M(i) = F(i)*Pfo(i)+B(i)*Pbo(i);
end
for i=1:length(F)
        U(i) = F(i)^0.25*B(i)^0.75;%utility function
end

for k=1:length(F)  %Happiness Function
        H(k) =
U(k)+100*((Pfo(k)*F(k)+Pbo(k)*B(k))/(M(k)+Pfo(k)*F(k)+Pbo(k)*B(k)));
end

z=1;
while z<=n;
    s = randi(N,1);
```

```
    t = randi(N,1);
%%%check if some one is dominant in the market
    for j=1:length(F)
        F1(j)=F(j)/sum(F);
        B1(j)=B(j)/sum(B);
        for i=1:length(F1)
            if F1(i)>=d
               Pf(i)=(1+(F1(i)-d)/2)*Pfo(i);% 2 is a scaler here
            else
                Pf(i)=Pfo(i);
           end
            if B1(i)>=d
                Pb(i)=(1+(B1(i)-d)/2)*Pbo(i);
            else
                Pb(i)=Pbo(i);
            end

        end
    end
%Gradient of the Happiness Function
G=[0.25*F(s)^(-0.75)*B(s)^(0.75)+100*Pf(s)*M(s)/(Pf(s)*F(s)+Pb(s)*B(s
)+M(s))^2
0.75*F(s)^(0.25)*B(s)^(-0.25)+100*Pb(s)*M(s)/(Pf(s)*F(s)+Pb(s)*B(s)+M
(s))^2;0.25*F(t)^(-0.75)*B(t)^(0.75)+100*Pf(t)*M(t)/(Pf(t)*F(t)+Pb(t)
*B(t)+M(t))^2
0.75*F(t)^(0.25)*B(t)^(-0.25)+100*Pb(t)*M(t)/(Pf(t)*F(t)+Pb(t)*B(t)+M
(t))^2];
    Pm=G(1,1)/G(1,2);%market price
    %check the trading condition
    if (mod(z,2)==1 && F(s)>=1 && B(t)>=Pm && A(s,t)==1 && det(G)<0)


        F(s) = F(s) - 1;
        B(s) = B(s) + Pm;
        F(t) = F(t) + 1;
        B(t) = B(t) - Pm;
    end
    if (mod(z,2)==0 && B(s)>=Pm && F(t)>=1 && A(s,t)==1 && det(G)>0)
        F(s) = F(s) + 1;
        B(s) = B(s) - Pm;
        F(t) = F(t) - 1;
        B(t) = B(t) + Pm;
    end
    for i=1:length(F)
        U(i) =  F(i)^0.25*B(i)^0.75;
```

```
    end

    for k=1:length(F)  %Happiness Function
        H(k)
=U(k)+100*((Pf(k)*F(k)+Pb(k)*B(k))/(M(k)+Pf(k)*F(k)+Pb(k)*B(k)));
    end
    V(z)=mean(H);
    z = z+1; %Counter
end
plot(V)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Chapter 5.2

```
function val = equi16avg(n)
%Price of goods
Pfo=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
Pbo=[3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3];
d=0.5;%dominant ratio
%Initial number of goods for each person
F=[randi(120,1) randi(120,1) randi(120,1) randi(120,1) randi(120,1)
randi(120,1) randi(120,1) randi(120,1) randi(120,1) randi(120,1)
randi(120,1) randi(120,1) randi(120,1) randi(120,1) randi(120,1)
randi(120,1)]; %firewood
B=[randi(40,1) randi(40,1) randi(40,1) randi(40,1) randi(40,1)
randi(40,1) randi(40,1) randi(40,1) randi(40,1) randi(40,1) randi(40,1)
randi(40,1) randi(40,1) randi(40,1) randi(40,1)
randi(40,1)];    %candybars

%Each person gets random number many firewood and chocolate-bars
bb=0.12;
b=[randi(100,1) randi(100,1) randi(100,1) randi(100,1) randi(100,1)
randi(100,1) randi(100,1) randi(100,1) randi(100,1) randi(100,1)
randi(100,1) randi(100,1) randi(100,1) randi(100,1) randi(100,1)
randi(100,1)];
a=0.04;
%Initial value of goods for each person
%define the proportion of the total amount of goods each person has
for i=1:length(F)
    M(i) = F(i)*Pfo(i)+B(i)*Pbo(i);
end
for i=1:length(F)
       U(i) = 1-exp(-a*F(i)-(bb+b(i)*bb*0.001)*B(i));
end
for k=1:length(F)  %Happiness Function
```

```
        H(k) = 0.75*U(k) +
0.25*((Pfo(k)*F(k)+Pbo(k)*B(k))/(M(k)+Pfo(k)*F(k)+Pbo(k)*B(k)));
end
for i=1:16
    for j=1:16
        X(i,j) = randi(10000,1);
    end
end
%%%%%doing the trade n times randomly picking  two traders
for p=1:11
    for i=1:16
        for j=1:16
            if X(i,j)<=(p-1)*1000
                A(i,j)=1;
                A(j,i)=1;
            end
            if i==j
                A(i,j)=0;
            end
        end
    end
    z=1;
    while z<=n;
        z = z+1; %Counter
        s = randi(16,1);
        t = randi(16,1);
        %check if some one is dominant in the market
        for j=1:length(F)
            F1(j)=F(j)/sum(F);
            B1(j)=B(j)/sum(B);
            for i=1:length(F1)
                if F1(i)>=d
                    Pf(i)=(1+(F1(i)-d)/2)*Pfo(i);% 2 is a scaler here
                else
                    Pf(i)=Pfo(i);
                end
                if B1(i)>=d
                    Pb(i)=(1+(B1(i)-d)/2)*Pbo(i);
                else
                    Pb(i)=Pbo(i);
                end

            end
        end
```

```
        %Gradient of the Happiness Function
        G=
[0.5*a*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+Pf(s)*M(s)/(Pf(s)*F(s)+Pb
(s)*B(s)+M(s))^2
0.5*(bb+b(i)*bb*0.001)*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+Pb(s)*M(s
)/(Pf(s)*F(s)+Pb(s)*B(s)+M(s))^2;
0.5*a*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+Pf(t)*M(t)/(Pf(t)*F(t)+Pb(
t)*B(t)+M(t))^2
0.5*(bb+b(i)*bb*0.001)*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+Pb(t)*M(t
)/(Pf(t)*F(t)+Pb(t)*B(t)+M(t))^2];
        Pm=G(1,1)/G(1,2);
        if (mod(z,2)==1 && F(s)>=1 && B(t)>=Pm && A(s,t)==1 && det(G)<0)

            F(s) = F(s) - 1;
            B(s) = B(s) + Pm;
            F(t) = F(t) + 1;
            B(t) = B(t) - Pm;
        end
        if (mod(z,2)==0 && B(s)>=Pm && F(t)>=1 && A(s,t)==1 && det(G)>0)
            F(s) = F(s) + 1;
            B(s) = B(s) - Pm;
            F(t) = F(t) - 1;
            B(t) = B(t) + Pm;
        end
        for i=1:length(F)
            U(i) = 1-exp(-a*F(i)-(bb+b(i)*bb*0.001)*B(i));
        end
        for k=1:length(F)
            H(k) = 0.75*U(k) +
0.25*((Pf(k)*F(k)+Pb(k)*B(k))/(M(k)+Pf(k)*F(k)+Pb(k)*B(k)));
        end
    end
    V(p)=mean(H);
end
plot(V)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Chapter 5.3

```
function val = equitorus16(n,kk,aa)
% Define the adjacent matrix A.
A=[0 1 0 1 1 0 0 0 0 0 0 0 1 0 0 0;%1
   1 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0;%2
   0 1 0 1 0 0 1 0 0 0 0 0 0 0 1 0;%3
   1 0 1 0 0 0 0 1 0 0 0 0 0 0 0 1;%4
```

```
    1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0;%5
    0 1 0 0 1 0 1 0 0 1 0 0 0 0 0 0;%6
    0 0 1 0 0 1 0 1 0 0 1 0 0 0 0 0;%7
    0 0 0 1 1 0 1 0 0 0 0 1 0 0 0 0;%8
    0 0 0 0 1 0 0 0 0 1 0 1 1 0 0 0;%9
    0 0 0 0 0 1 0 0 1 0 1 0 0 1 0 0;%10
    0 0 0 0 0 0 1 0 0 1 0 1 0 0 1 0;%11
    0 0 0 0 0 0 0 1 1 0 1 0 0 0 0 1;%12
    1 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1;%13
    0 1 0 0 0 0 0 0 0 1 0 0 1 0 1 0;%14
    0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 1;%15
    0 0 0 1 0 0 0 0 0 0 0 1 1 0 1 0];%16

for i=1:16
    for j=1:16
        C(i,j)=1;
        if i==j
            C(i,j)=0;
        end
    end
end
%Price of goods
for i=1:16
    Pfo(i)=1;
    Pbo(i)=3;
    %Initial number of goods for each person
    F(i)=randi(120,1);  %firewood
    B(i)=randi(40,1);%candybars
    FF(i)=F(i);
    BB(i)=B(i);
    b(i)=randi(100,1);
end
d=0.5;%dominant ratio

%Initial value of goods for each person
%define the proportion of the total amount of goods each person has
for i=1:length(F)
    F1(i)=F(i)/sum(F);
    B1(i)=B(i)/sum(B);
    M(i)  = F(i)*Pfo(i)+B(i)*Pbo(i);
end
%Utility Function
a=aa;
bb=3*a;
```

```
%%%%%doing the trade n times randomly picking  two people
z=1;
while z<=n;

    s = randi(16,1);
    t = randi(16,1);

    %%%check if someone is dominant in the market
    for j=1:length(F)
        F1(j)=F(j)/sum(F);
        B1(j)=B(j)/sum(B);
        for i=1:length(F1)
            if F1(i)>=d
               Pf(i)=(1+(F1(i)-d)/2)*Pfo(i);% 2 is a scaler here
            else
                Pf(i)=Pfo(i);
           end
            if B1(i)>=d
                Pb(i)=(1+(B1(i)-d)/2)*Pbo(i);
            else
                Pb(i)=Pbo(i);
            end

        end
    end

    %Gradient of the Happiness Function
    G=
[kk*a*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+(1-kk)*Pf(s)*M(s)/(Pf(s)*F
(s)+Pb(s)*B(s)+M(s))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+(1-kk)*Pb(
s)*M(s)/(Pf(s)*F(s)+Pb(s)*B(s)+M(s))^2;
kk*a*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+(1-kk)*Pf(t)*M(t)/(Pf(t)*F(
t)+Pb(t)*B(t)+M(t))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+(1-kk)*Pb(
t)*M(t)/(Pf(t)*F(t)+Pb(t)*B(t)+M(t))^2];
    Pm=G(1,1)/G(1,2);

    if (mod(z,2)==1 && F(s)>=1 && B(t)>=Pm && A(s,t)==1 && det(G)<0)

        F(s) = F(s) - 1;
        B(s) = B(s) + Pm;
        F(t) = F(t) + 1;
```

```
        B(t) = B(t) - Pm;
    end
    if (mod(z,2)==0 && B(s)>=Pm && F(t)>=1 && A(s,t)==1 && det(G)>0)

        F(s) = F(s) + 1;
        B(s) = B(s) - Pm;
        F(t) = F(t) - 1;
        B(t) = B(t) + Pm;
    end
    for i=1:length(F)
        U(i) = 1-exp(-a*F(i)-(bb+b(i)*bb*0.001)*B(i));
    end
    for k=1:length(F)  %Happiness Function
        H(k) = kk*U(k) +
(1-kk)*((Pf(k)*F(k)+Pb(k)*B(k))/(M(k)+Pf(k)*F(k)+Pb(k)*B(k)));
    end
    V(z)=mean(H);
%perfect graph with 16 people and same initial value
%Gradient of the Happiness Function
    G=
[kk*a*exp(-a*FF(s)-(bb+b(i)*bb*0.001)*BB(s))+(1-kk)*Pf(s)*M(s)/(Pf(s)
*FF(s)+Pb(s)*BB(s)+M(s))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*FF(s)-(bb+b(i)*bb*0.001)*BB(s))+(1-kk)*P
b(s)*M(s)/(Pf(s)*FF(s)+Pb(s)*BB(s)+M(s))^2;
kk*a*exp(-a*FF(t)-(bb+b(i)*bb*0.001)*BB(t))+(1-kk)*Pf(t)*M(t)/(Pf(t)*
FF(t)+Pb(t)*BB(t)+M(t))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*FF(t)-(bb+b(i)*bb*0.001)*BB(t))+(1-kk)*P
b(t)*M(t)/(Pf(t)*FF(t)+Pb(t)*BB(t)+M(t))^2];
    Pm=G(1,1)/G(1,2);
    if (mod(z,2)==1 && FF(s)>=1 && BB(t)>=Pm && C(s,t)==1 && det(G)<0)

        FF(s) = FF(s) - 1;
        BB(s) = BB(s) + Pm;
        FF(t) = FF(t) + 1;
        BB(t) = BB(t) - Pm;
    end

    if (mod(z,2)==0 && BB(s)>=Pm && FF(t)>=1 && C(s,t)==1 && det(G)>0)

        FF(s) = FF(s) + 1;
        BB(s) = BB(s) - Pm;
        FF(t) = FF(t) - 1;
        BB(t) = BB(t) + Pm;
    end
```

```
    for i=1:length(F)
        U(i) = 1-exp(-a*FF(i)-(bb+b(i)*bb*0.001)*BB(i));
    end

    for k=1:length(F)  %Happiness Function
        H(k) = kk*U(k) +
(1-kk)*((Pf(k)*FF(k)+Pb(k)*BB(k))/(M(k)+Pf(k)*FF(k)+Pb(k)*BB(k)));
    end
    VV(z)=mean(H);
    z = z+1; %Counter
end
plot(V)
hold on
plot (VV,'--')
hold off
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Chapter 5.4

```
function val = equi1616(n,kk,aa,Z)
%kk=k aa=a n:#of trades Z:# of people in each group
% Define the adjacent matrix A.
for i=1:2*Z
    for j=1:2*Z
        A(i,j)=0;
    end
end
for i=1:Z
    for j=1:Z
        if i==j
            A(i,j)=0;
        else
            A(i,j)=1;
            A(i+Z,j+Z);
        end
    end
end
%Price of goods
for i=1:2*Z
    Pfo(i)=1;
    Pbo(i)=3;
    %Initial number of goods for each person
    F(i)=randi(120,1);  %firewood
```

```
    B(i)=randi(40,1);%candybars
    b(i)=randi(100,1);%hedging b to be around 3*a
end
d=0.5;%dominant ratio
%Utility Function
a=aa;
bb=3*a;

%caculating the initail H of each person
for i=1:length(F)
        U(i) = 1-exp(-a*F(i)-(bb+b(i)*bb*0.001)*B(i));
end

for k=1:length(F)  %Happiness Function
        H(k) = kk*U(k) +
(1-kk)*((Pfo(k)*F(k)+Pbo(k)*B(k))/(M(k)+Pfo(k)*F(k)+Pbo(k)*B(k)));
end
for y=1:Z
    for w=1:y
        A(y,y+Z)=1;
    end
    z=1;
    while z<=n;

        s = randi(2*Z,1);
        t = randi(2*Z,1);
        %%%check if some one is dominant in the market
        for j=1:length(F)
            F1(j)=F(j)/sum(F);
            B1(j)=B(j)/sum(B);
            for i=1:length(F1)
                if F1(i)>=d
                    Pf(i)=(1+(F1(i)-d)/2)*Pfo(i);% 2 is a scaler here
                else
                    Pf(i)=Pfo(i);
                end
                if B1(i)>=d
                    Pb(i)=(1+(B1(i)-d)/2)*Pbo(i);
                else
                    Pb(i)=Pbo(i);
                end

            end
        end
```

```
        %Gradient of the Happiness Function
        G=
[kk*a*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+(1-kk)*Pf(s)*M(s)/(Pf(s)*F
(s)+Pb(s)*B(s)+M(s))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+(1-kk)*Pb(
s)*M(s)/(Pf(s)*F(s)+Pb(s)*B(s)+M(s))^2;
kk*a*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+(1-kk)*Pf(t)*M(t)/(Pf(t)*F(
t)+Pb(t)*B(t)+M(t))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+(1-kk)*Pb(
t)*M(t)/(Pf(t)*F(t)+Pb(t)*B(t)+M(t))^2];
    Pm=G(1,1)/G(1,2);
        if (mod(z,2)==1 && F(s)>=1 && B(t)>=Pm && A(s,t)==1 && det(G)<0)

            F(s) = F(s) - 1;
            B(s) = B(s) + Pm;
            F(t) = F(t) + 1;
            B(t) = B(t) - Pm;
        end
        if (mod(z,2)==0 && B(s)>=Pm && F(t)>=1 && A(s,t)==1 && det(G)>0)

            F(s) = F(s) + 1;
            B(s) = B(s) - Pm;
            F(t) = F(t) - 1;
            B(t) = B(t) + Pm;
        end


        for i=1:length(F)
            U(i) = 1-exp(-a*F(i)-(bb+b(i)*bb*0.001)*B(i));
        end

        for k=1:length(F)  %Happiness Function
            H(k) = kk*U(k) +
(1-kk)*((Pf(k)*F(k)+Pb(k)*B(k))/(M(k)+Pf(k)*F(k)+Pb(k)*B(k)));
        end
        z = z+1; %Counter
    end
    V(y)=mean(H);
end
plot(V)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function val = equi3d(kk,aa,Z)
%kk=k aa=a n:#of trades Z:# of people in each group
% Define the adjacent matrix A.
```

```
for i=1:2*Z
    for j=1:2*Z
        A(i,j)=0;
    end
end
for i=1:Z
    for j=1:Z
        if i==j
            A(i,j)=0;
        else
            A(i,j)=1;
            A(i+Z,j+Z);
        end
    end
end
%Price of goods
for i=1:2*Z
    Pfo(i)=1;
    Pbo(i)=3;
    %Initial number of goods for each person
    F(i)=randi(120,1);  %firewood
    B(i)=randi(40,1);%candybars
    b(i)=randi(100,1);%hedging b to be around 3*a
end
d=0.5;%dominant ratio
%define the proportion of the total amount of goods each person has
for i=1:length(F)
    F1(i)=F(i)/sum(F);
    B1(i)=B(i)/sum(B);
    M(i) = F(i)*Pfo(i)+B(i)*Pbo(i);
end
a=aa;
bb=3*a;

%caculating the initail H of each person
for i=1:length(F)
        U(i) = 1-exp(-a*F(i)-(bb+b(i)*bb*0.001)*B(i));
end

for k=1:length(F)  %Happiness Function
        H(k) = kk*U(k) +
(1-kk)*((Pfo(k)*F(k)+Pbo(k)*B(k))/(M(k)+Pfo(k)*F(k)+Pbo(k)*B(k)));
end
```

```
for y=2:2:Z
    for w=1:y
        A(y,y+Z)=1;
    end
    for n=50:50:300;
    z=1;
    while z<=n;

        s = randi(2*Z,1);
        t = randi(2*Z,1);
        %%%check if some one is dominant in the market
        for j=1:length(F)
            F1(j)=F(j)/sum(F);
            B1(j)=B(j)/sum(B);
            for i=1:length(F1)
                if F1(i)>=d
                    Pf(i)=(1+(F1(i)-d)/2)*Pfo(i);% 2 is a scaler here
                else
                    Pf(i)=Pfo(i);
                end
                if B1(i)>=d
                    Pb(i)=(1+(B1(i)-d)/2)*Pbo(i);
                else
                    Pb(i)=Pbo(i);
                end

            end
        end
        %Gradient of the Happiness Function
        G=
[kk*a*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+(1-kk)*Pf(s)*M(s)/(Pf(s)*F
(s)+Pb(s)*B(s)+M(s))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*F(s)-(bb+b(i)*bb*0.001)*B(s))+(1-kk)*Pb(
s)*M(s)/(Pf(s)*F(s)+Pb(s)*B(s)+M(s))^2;
kk*a*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+(1-kk)*Pf(t)*M(t)/(Pf(t)*F(
t)+Pb(t)*B(t)+M(t))^2
kk*(bb+b(i)*bb*0.001)*exp(-a*F(t)-(bb+b(i)*bb*0.001)*B(t))+(1-kk)*Pb(
t)*M(t)/(Pf(t)*F(t)+Pb(t)*B(t)+M(t))^2];
    Pm=G(1,1)/G(1,2);
        if (mod(z,2)==1 && F(s)>=1 && B(t)>=Pm && A(s,t)==1 && det(G)<0)

            F(s) = F(s) - 1;
            B(s) = B(s) + Pm;
            F(t) = F(t) + 1;
```

```
            B(t) = B(t) - Pm;
        end

        if (mod(z,2)==0 && B(s)>=Pm && F(t)>=1 && A(s,t)==1 && det(G)>0)

            F(s) = F(s) + 1;
            B(s) = B(s) - Pm;
            F(t) = F(t) - 1;
            B(t) = B(t) + Pm;
        end


        for i=1:length(F)
            U(i) = 1-exp(-a*F(i)-(bb+b(i)*bb*0.001)*B(i));
        end

        for k=1:length(F)  %Happiness Function
            H(k) = kk*U(k) +
(1-kk)*((Pf(k)*F(k)+Pb(k)*B(k))/(M(k)+Pf(k)*F(k)+Pb(k)*B(k)));
        end

        z = z+1; %Counter
        x=int8(n/50);
        yy=int8(y/2);
    end
    V(yy,x)=mean(H);
   end
end
Y=[2 4 6 8 10 12 14 16];
X=[50 100 150 200 250 300];
surf(X,Y,V)
```