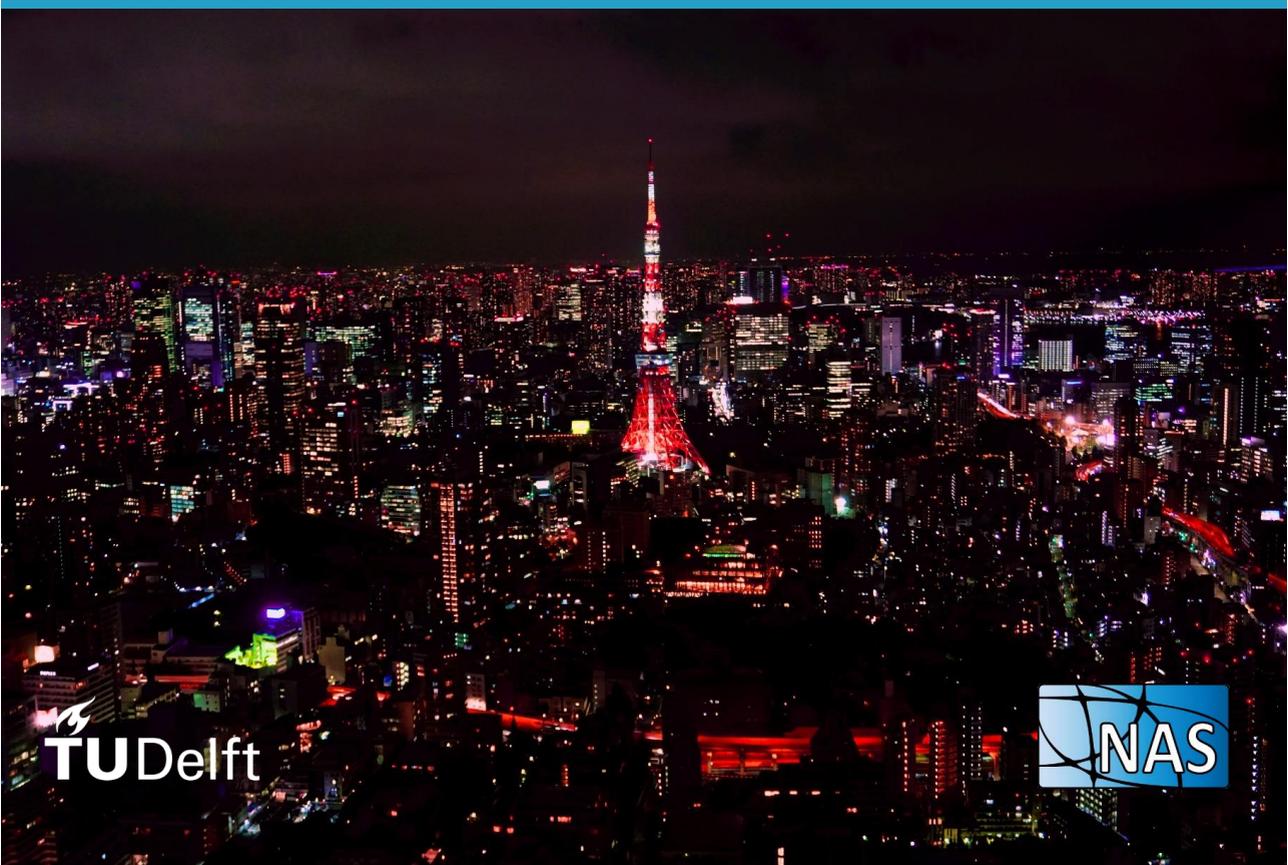


# Non-Consensus Opinion Models with Byzantine Nodes

XINHAN LIU





# **NON-CONSENSUS OPINION MODELS WITH BYZANTINE NODES**

by

**Xinhan LIU**

to obtain the degree of Master of Science  
in Electrical Engineering  
Track Wireless Communication and Sensing  
at the Delft University of Technology,  
to be defended publicly on Wednesday February 23, 2022 at 10:00 AM.

Student number: 5137985  
Project duration: March, 2021 – February, 2022  
Thesis committee: Prof.dr.ir. Rob Kooij, TU Delft, supervisor  
Dr. J.L.A. Dubbeldam, TU Delft  
Ir. Massimo Achterberg, TU Delft, daily supervisor





# PREFACE

With this thesis project, "Non-Consensus Opinion Models with Byzantine Nodes", I finished the Master of Science degree in Electrical Engineering at the Delft University of Technology. The project has been carried out at the Network Architecture and Services (NAS) group. With the completion of this thesis report, my master time is coming to an end. I will always cherish the time in TU Delft. In these two years, I have experienced a lot and grown a lot. Many people have given me a lot of help and encouragement in this process, and I would like to express my most sincere gratitude to them.

Firstly, I would like to offer my great thanks to my supervisor, Prof. Rob Kooij. Rob has given me the general direction of my research and many useful advice. He always encouraged me and was exciting with every little progress I made. I was very unreliable like a "Bunny Rabbit" when I first started my project. With Rob's help I gradually learned how to work and got some good results. What I learned from Rob will benefit me for life.

Secondly, I sincerely appreciate my daily supervisor ir. Massimo Achterberg. Massimo gave me a lot of good advice and showed me how an excellent PhD student should work. Massimo always answered my emails in a timely manner and gave me professional and thoughtful answers to my questions. He worked with Rob to ensure the successful completion of my thesis project.

I would also like to thanks all my friends, especially Anqi Chen and Yichang Han for their continued encouragement in the Covid-19 period. Special thanks to Dukong Ma, Yazhou Xi, Shih-Lao and Mr. Chuiniu to help me through the boring time.

Finally, the most sincere thanks are given to my parents for raising me and supporting me in my life and studies. They have given me the most selfless love. I will work and live well to repay them for their efforts.

Thank you to all the people I have mentioned and not mentioned who have helped and accompanied me. Any future success I may have is inseparable from you. I hope you all have a wonderful life!

*Xinhan Liu  
Delft, February 2022*



# ABSTRACT

Opinion dynamics models study how the interaction among people influences the opinion formation process. In most opinion dynamics models, only one opinion could exist in the steady state, which is different from the real-life opinion formation process. In 2009, Shao *et al.* introduced a non-consensus opinion (NCO) model, which allows different opinions to coexist in the steady state. This thesis extends the NCO model by introducing a special type of nodes, Byzantine nodes, to play the role of dishonest people. The Byzantine NCO model is more in line with the real-world opinion formation process because it considers that people who express opinions are not always honest. I build an NCO model simulation algorithm and use this algorithm to perform simulations on three different network models: small-scale graphs, the Erdős-Rényi random graph and the scale-free network. In Byzantine node selection, three different strategies are proposed, according to the degree of the selected nodes. I find a new steady state for the NCO model: the cyclic steady state. The cyclic behaviour of the NCO and Byzantine NCO model is discussed, and some networks with a long cycle period are given. I also introduced a general method to generate networks with extremely long cycle periods. The other properties of the Byzantine NCO model, such as the probability of cyclic behavior, the final opinion distribution and the convergence time are researched. By performing simulations on the network models, I find that the introduction of Byzantine nodes could help the system to reach a steady state with a more balanced opinion ratio. The introduction of Byzantine nodes could decrease the critical threshold of the NCO model and promote the coexistence steady state. A mechanism in which Byzantine nodes influences the convergence time by influencing the steady state is suggested.



# CONTENTS

<b>Preface</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview of previous opinion dynamics models	2
1.2 Contribution	2
1.3 Thesis outline	3
<b>2 Byzantine NCO Model</b>	<b>5</b>
2.1 NCO Model	5
2.2 Byzantine NCO Model	6
2.3 Network Science	8
2.3.1 Graph Theory	8
2.3.2 Network Models	9
2.4 Simulation Method	10
<b>3 Cyclic Behavior</b>	<b>15</b>
3.1 Typical Steady States of Byzantine NCO Model	15
3.1.1 Fixed Steady States	16
3.1.2 Cyclic steady states	17
3.2 Cyclic cases with long cycle period	19
3.2.1 Combined Network	27
3.3 Occurrence of cyclic behavior	30
<b>4 Opinion Distribution at Steady State</b>	<b>37</b>
4.1 Final Opinion Fraction	37
4.1.1 Exhaustive Research on Graphs with $N=7, L=10$	39
4.1.2 Simulations on ER and SF networks	41
4.2 Critical Threshold	44
<b>5 Convergence Time of the Byzantine NCO model</b>	<b>51</b>
5.1 Convergence Time	51
5.2 Exhaustive Research on Graphs with $N=7, L=10$	53
5.2.1 The Convergence Time for graphs with Different Number of Byzantine Nodes	53
5.3 Research on ER and SF Networks	56
<b>6 Conclusions and Future Work</b>	<b>61</b>
6.1 Conclusions	61
6.2 Future Research	62
<b>A Appendix A</b>	<b>67</b>



# 1

## INTRODUCTION

On November 3, 2020, as the U.S. election was held as scheduled, a big show started, which drew people's attention all over the world. The U.S. election is not only a voting campaign but also a sociological experiment. Scientists have put different ways to predict the election result, and many have proven effective. In the social dynamics area [1], we regard it as an opinion dynamics question, which researches opinion formation process based on complex networks and statistical physics [2], [3]. The design of the opinion dynamics models is driven by human behavior and includes many elements, such as individual predisposition, the influence of other people (social networks playing a crucial role in this respect), and many others. Different models have been developed, encompassing different elements.

Many opinion dynamics models are based on complex networks. Network science is widely used in the analysis of complex systems [2]–[8]. A complex network is a graph consisting of vast numbers of nodes and intricate connections between nodes. Unlike those with simple designs, complex networks exhibit higher flexibility and more diverse features that often appear in real-world networks, making it possible to use complex networks to analyze real-world problems. Many complex systems that are researched today, such as the Internet, neural networks, brain networks, and social networks, can be described by using complex networks [7]. Complex networks are an excellent tool to study social models, where the nodes and links in the network are used to indicate individuals (agents) and the relation between them.

Shao *et al.* (2009) proposed a Non-Consensus Opinion (NCO) model, which is an opinion dynamics model that shows some of the same properties as real social networks [9]. This thesis researches and expands the NCO model by introducing a new type of node, the Byzantine node, to the model. The NCO model assumes that all people are honest, but we always meet rascals who lie and make trouble in real life. In this thesis, we extend the NCO model to Byzantine NCO model. In the Byzantine-NCO model, the Byzantine node is introduced, a new type of node that plays the role of a liar in a crowd.

## 1.1. OVERVIEW OF PREVIOUS OPINION DYNAMICS MODELS

Most of the traditional opinion models are based on spin models such as Sznajd model [10], the voter model [11], the majority rule model [12], and social impact model [13]. A drawback of conventional spin models is that they usually result in an ordered steady state (consensus steady state), while in real life, different opinions tend to coexist in the steady state [2], [14].

In 2009, Shao *et al.* proposed the Non-Consensus Opinion model (NCO model), which can be used to research the opinion dynamics of a group of people [9]. The NCO model is based on a complex network, which uses nodes in the network to represent people and links to show the relationship between those people. Unlike models based on spin systems, the NCO model allows for non-consensus steady states, which is more in line with real life. Shao's *et al.* study illustrates an important fact that if the number of people holding the minority opinion is sufficiently large, the minority opinion holders could form a stable cluster, which the other opinion cannot invade. Shao *et al.* also finds that the NCO model shows similar properties as the invasion percolation process [15], which means this opinion dynamics model is closely related to some well-known physical phenomena.

Shao's *et al.* classic NCO model solves the problem of different opinions not coexisting in other opinion models; however, it still cannot ideally mimic the opinion formation process. In 2011, Qian *et al.* proposed an inflexible contrarian opinion (ICO) model by introducing some stubborn agents who never change their opinion under any circumstances [16]. Qian's study makes the NCO model more relevant to real-life social networks. In 2013, Qian *et al.* added a weighting factor to the NCO model and proposed the NCO model [14]. The weighting factor  $W$  represents the importance of a person's opinion in decision-making, where a large  $w$  makes one's opinion hard to change.

## 1.2. CONTRIBUTION

The main contributions of this thesis are:

1. Extension of the types of nodes in the NCO model by introducing Byzantine nodes. Byzantine nodes are used to model dishonest people in social networks to match real-life opinion formation dynamics closely.
2. Discussion of the issue of the cyclic behavior of the NCO model, which has been ignored in past studies. The introduction of Byzantine nodes gives rise to many graph patterns with a long cycle period and a general method to generate long-length cyclic graphs.
3. Analysis of the convergence time of the NCO model and discussed the effect of different initial parameter settings on the convergence time.
4. Investigation of the relationship between the final opinion fraction and Byzantine node selection strategies. Description of the balancing effect of Byzantine nodes

on different opinions in the opinion network.

5. Research and explanation of the effect of the introduction of Byzantine nodes on the critical threshold. Proposition that the introduction of Byzantine nodes has an inhibiting effect on the formation of large homogeneous opinion clusters.

### 1.3. THESIS OUTLINE

The thesis is organized as follows. Chapter 2 introduces the NCO model, Byzantine NCO model and provides a basic introduction to network science. After that, I provide a computational method to simulate the NCO dynamics. In Chapter 3 I present the typical steady-state behavior of the NCO model and introduce a new type of steady state, the cyclic steady state. Then I gave some cyclic cases with long cycle period and a general way to generate networks with extremely long cycle period. Besides these, I also measure the probability of occurrence of cyclic behavior for different initial states. In Chapter 4, I discuss how the final opinion fraction of the Byzantine NCO model changes as the number of Byzantine nodes increases and compare the magnitude of the effect of different Byzantine node selection strategies on the final opinion fraction. In Chapter 5, I present the effect of the introduction of Byzantine nodes on the convergence time of the Byzantine NCO model and discuss the relationship between the Hamming distance between the initial state and the convergence state and the convergence time.



# 2

## BYZANTINE NCO MODEL

### 2.1. NCO MODEL

The Non-Consensus Opinion(NCO) model is developed to research the opinion formation process. Most of the previous opinion models lead to a consensus state, which is not realistic, since in the real opinion dynamics process, competitions always end with a coexistence result. Consistent with reality the NCO model allows a non-consensus stable state [16].

The NCO model is based on the assumption that in the opinion formation process, the agent's opinion is influenced by his current opinion and the opinion of his friends represented as the node's neighbours [9]. More specifically, a node will change its opinion if and only if the number of its neighbours holding a different opinion is greater than the number of nodes holding the same opinion plus one (the node's own opinion). There are two opinions, denoted as positive opinion  $\sigma_+$  and negative opinion  $\sigma_-$ . The agents could only hold one opinion at a time.

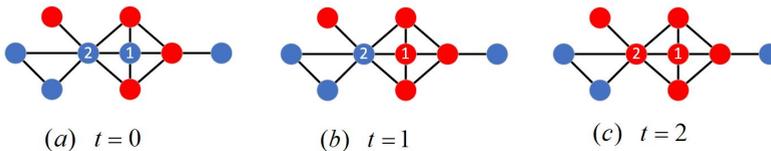


Figure 2.1: Dynamics of the NCO model on a network with  $N = 9$  nodes. The node's color denotes the opinion of the node, red means the node holds a positive opinion  $\sigma_+$ , blue means the node has a negative opinion  $\sigma_-$ .

(a) At  $t = 0$ , each node is randomly assigned an opinion, four hold  $\sigma_+$  (red), and five hold  $\sigma_-$  (blue). Node 1 will change its opinion because it has 3  $\sigma_+$  neighbor nodes and 1  $\sigma_-$  neighbor node, the local opinion ratio is  $\sigma_+ : \sigma_- = 3 : 2$ . (b) At  $t = 1$ , as node 1 changing its opinion, the opinion held by node 2 becomes a local minority opinion, and node 2 will also change its opinion. (c) At  $t = 2$ , all nodes hold the local majority opinion, and the system reaches a steady state.

Fig. 2.1 gives an example of the dynamics of the NCO model. At  $t = 0$ , nine nodes are randomly assigned two different kinds of opinion, node 1 has three  $\sigma_+$  neighbors and one negative neighbor, so for node 1, the local majority opinion is the  $\sigma_+$  opinion, and the opinion of node 1 will convert to a  $\sigma_+$  opinion. At  $t = 1$ , as node 1 changes its opinion, the local opinion ratio of node 2 becomes 4 to 2, so node 2 will also change its opinion. At  $t = 2$ , the opinions of all nodes become local majority opinions. The system reaches a steady state [9].

The example given in Fig. 2.1 finally reach a fixed steady state with a opinion ratio of  $\sigma_+ : \sigma_- = 5 : 3$ . Once the system reaches a fixed steady state, the states of the nodes in the network will no longer change. The steady state is only related to the initial state for a given network, and the same steady state may originate from different initial states, so the NCO model is deterministic.

## 2.2. BYZANTINE NCO MODEL

The NCO model assumes that all nodes are honest, but people are not always honest in real life, and some may deliberately hide their true opinions. To simulate this situation, I introduce Byzantine nodes to the NCO model.

### BYZANTINE NODES

Byzantine nodes are a special type of nodes that declare opinions contrary to their true opinions. In real social dynamic processes, Byzantine nodes can be used to represent the bad people who try to mess up in elections, or spies who infiltrate the decision-making level of a country.

As is shown in Fig. 2.2, like normal nodes, Byzantine nodes decide their opinion based on local majority opinion.

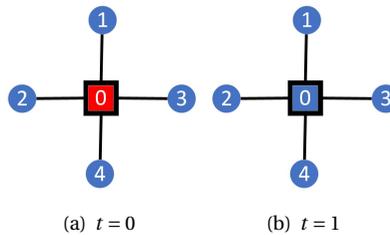


Figure 2.2: A demonstration of the behavior of a Byzantine node. Normal nodes are depicted as circles, while Byzantine nodes are depicted as squares. At  $t = 0$ , Byzantine node 0 has 4  $\sigma_-$  neighbors, the local opinion ratio is  $\sigma_- : \sigma_+ = 4 : 1$ , then the Byzantine node 0 changes its opinion.

The difference is that Byzantine nodes do not reveal their true opinions, they lie about their opinions. As shown in Fig. 2.3, the Byzantine nodes make their neighbors misjudge their own local majority opinion.

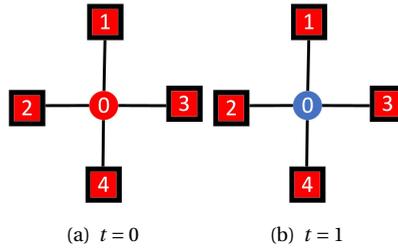


Figure 2.3: A demonstration of the behavior of a Byzantine node. At  $t = 0$ , node 0 has 4  $\sigma_+$  Byzantine neighbors, the real local opinion ratio is  $\sigma_- : \sigma_+ = 0 : 5$ . But due to the lying nature of Byzantine nodes, node 0 misjudges the local opinion ratio as  $\sigma_- : \sigma_+ = 4 : 1$ , then the Byzantine node 0 changes its opinion.

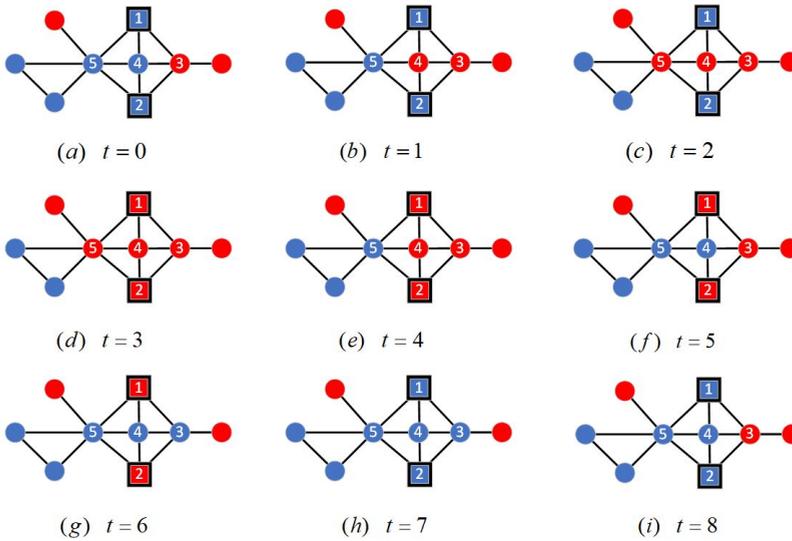


Figure 2.4: Dynamics of the Byzantine NCO model on a network with  $N = 9$  nodes. The node's color denotes the opinion of the node, red means the node holds a positive opinion  $\sigma^+$ , blue means the node has a negative opinion  $\sigma^-$ . At  $t = 0$  to  $t = 6$ , the normal nodes 3,4,5 change their opinion according to their local majority opinion. At  $t = 7$ , the Byzantine nodes 1, 2 change their opinion, then node 3's local majority opinion changes from negative to positive due to the lying nature of Byzantine nodes. At  $t = 8$ , the graph shows the same state as initial state. The graph will repeat the previous dynamics.

Fig. 2.4 shows the dynamics of the Byzantine NCO model on a network with  $N = 9$  nodes, where node 1 and node 2 are Byzantine nodes and the rest of the nodes are normal nodes. At  $t = 0$  every node is randomly assigned an opinion. Node 4 has three negative neighbor nodes and one positive neighbor node, but since node 1 and node 2 are Byzantine nodes, they will lie to node 4, who will mistakenly believe that he has three positive neighbor nodes and one negative neighbor node. Then node 4 will think the local majority opinion is positive and change its opinion to positive. At  $t = 1$ , as node 4 comprising, node 5 will also change its opinion. At  $t = 2$ , as both nodes 4 and

5 change their opinions, nodes 1 and 2 will find that the local majority opinion has become positive; they will also change their opinions. At  $t = 3$ , because node 1 and 2 changed their opinion and pretended they held the negative opinion, node 5 will be misled and change its opinion. At  $t = 4$  to  $t = 7$ , the state of each node in the network will continue to change. At  $t = 8$ , as the opinion of node 3 changes, the state of the network changes to the same state as at  $t = 0$ , and the same process will repeatedly happen without stopping, and the system will never reach a stable state. I refer to the phenomenon occurring in this example as cyclic behavior. The length of the cycle in this example is 8. The cyclic state is a special kind of steady state. Unlike the fixed steady state, the cyclic steady state does not converge to a particular state but keeps changing between several states. The cyclic behavior of Byzantine NCO model is a research focus of this thesis.

## 2.3. NETWORK SCIENCE

Before introducing the simulation method, I first introduce the following essential knowledge of network science. Network science is a science that studies the global and local properties of networks. This part introduces some knowledge about graph theory and some fundamental metrics to study network properties I used in this project [17].

### 2.3.1. GRAPH THEORY

#### ADJACENCY MATRIX

The adjacency matrix is a matrix notation representation of the graph  $G$ . We use the matrix elements to illustrate the presence of links between nodes [17]. For a network with  $N$  nodes, its adjacency matrix (Eq. (2.1)) is an  $N \times N$  matrix, where the element  $a_{ij}$  (Eq. (2.2)) in the matrix denotes the connection between node  $i$  and node  $j$ .

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,N} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N,1} & a_{N,2} & \cdots & a_{N,N} \end{bmatrix} \quad (2.1)$$

where,

$$a_{i,j} = \begin{cases} 1, & \text{if node } i \text{ is connected to node } j \\ 0, & \text{otherwise} \end{cases} \quad (2.2)$$

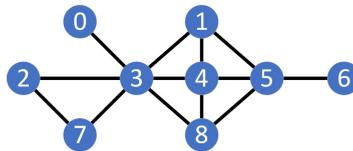


Figure 2.5: Example graph with  $N = 9$ ,  $L = 13$

The adjacency matrix of the example graph with  $N = 9$ ,  $L = 13$  shown in Fig. 2.5 is given as

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (2.3)$$

For undirected networks, like Fig. 2.5, the adjacency matrix is symmetric, but for directed networks, the adjacency matrix is asymmetric. The values of the diagonal elements of this adjacency matrix are all 0, which means that there are no self-loops in this network.

### DEGREE

The degree  $d_i$  of node  $i$  is defined as the number of neighbours for node  $i$  in the network [18]. The degree of node  $i$  follows from the adjacency matrix as

$$d_i = \sum_{j=1}^N a_{ij} \quad (2.4)$$

The degree is one of the most straightforward measures of the importance of a node. The larger the degree of a node, the more neighbouring nodes it has and thus the more nodes it can directly influence. There are many other measures of node importance, such as closeness, betweenness, and eigenvector centrality [19], [20], which are not used in this project.

### 2.3.2. NETWORK MODELS

Network science research starts with studying some simple networks with specific simple properties, such as regular networks and purely random networks. As technology advances and application scenarios become more complex, networks with simple structures are increasingly incompetent to meet the needs of network research. The study of network science has entered the era of complex networks, which refers to networks with many nodes and complex structures, such as small-world and scale-free networks. In this project, I mainly used the Erdős–Rényi random graphs [21] and scale-free networks [22].

#### ERDŐS–RÉNYI RANDOM GRAPH

In a random graph  $G(N, p)$ , there are  $N$  nodes in the graph, and every pair of nodes is connected with a probability  $p$ . The probability that the node in the graph has degree  $k$

is

$$\Pr(\text{deg}(v) = k) = \binom{N-1}{k} p^k (1-p)^{N-1-k} \quad (2.5)$$

which follows a binomial distribution. The mean value of the degree of the nodes for this type of graph is

$$\langle k \rangle = (N-1)p \quad (2.6)$$

For ER graphs, there is a threshold  $p_{\text{connected}} = \frac{\ln(N)}{N}$  [21]. If  $p$  is smaller than the threshold, then there is a high probability that the network is disconnected. If  $p$  is larger than the threshold, then there is a high probability that the network is connected [23].

### SCALE-FREE NETWORK

The degree distribution of the SF network follows a power-law distribution which is denoted as

$$\Pr(D = k) \propto k^{-\lambda} \quad (2.7)$$

Barabási and Albert proposed a way to generate power-law network in 1999 [24], [25], which are known as the Barabasi-Albert (BA) networks. The specific algorithm is as follows:

- 1) Starting from a small network  $G_0$  (this network has  $n_0$  nodes and  $E_0$  edges), new nodes are added into  $G_0$  one by one.
- 2) Suppose the original network already has  $n$  nodes ( $s_1, s_2, \dots, s_n$ ). Every time when a new node  $s_{n+1}$  is added,  $m$  ( $m < n$ ) links from this new node are connect to the original  $n$  nodes.
- 3) The newly added node are connected to existing nodes according to the degree of the nodes. For a node  $s_i$  of degree  $d_i$ , the probability of a new node establishing a link with it is

$$p_i = \frac{d_i}{\sum_{j=1}^n d_j} \quad (2.8)$$

In this thesis we define a SF network with three parameters  $N$ ,  $\lambda$  and  $k_{\text{min}}$ , where  $N$  is the number of nodes in the network,  $\lambda$  is the exponent of the degree distribution and  $k_{\text{min}}$  is the minimum value of the nodes' degree.

## 2.4. SIMULATION METHOD

The simulation of Byzantine NCO model dynamics is the process of deriving the state of each node in the network at the next time slot based on the current state of each node in the network using the knowledge of network science. Here I use state vectors  $\vec{v}_s$  (2.9) to represent the state of individual nodes in the network and Byzantine vectors  $\vec{v}_B$  (2.11) to describe whether the node is a Byzantine node or a normal node.

$$\vec{v}_s = [s_1 \quad s_2 \quad \dots \quad s_N]^T \quad (2.9)$$

where,

$$s_i = \begin{cases} 1, & \text{if node } i \text{ holds a positive opinion} \\ -1, & \text{if node } i \text{ holds a negative opinion} \end{cases} \quad (2.10)$$

$$\vec{v}_B = [b_1 \quad b_2 \quad \cdots \quad b_N]^T \quad (2.11)$$

where,

$$b_i = \begin{cases} 1, & \text{if node } i \text{ is a normal node} \\ -1, & \text{if node } i \text{ is a Byzantine node} \end{cases} \quad (2.12)$$

At a certain time  $t = t_0$ , given the state vector  $\vec{v}_s(t_0)$  and the Byzantine vector  $\vec{v}_B(t_0)$ , we know that the state that each node declared can be represented as the declared vector  $\vec{v}_d$  (2.13) which is the Hadamard product of  $\vec{v}_s(t_0)$  and  $\vec{v}_B(t_0)$

$$\vec{v}_d = \vec{v}_s \circ \vec{v}_B = [b_1 \cdot s_1 \quad b_2 \cdot s_2 \quad \cdots \quad b_N \cdot s_N]^T \quad (2.13)$$

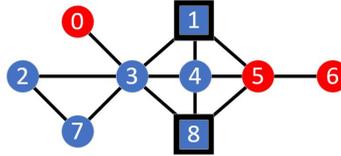


Figure 2.6: Example graph with  $N = 9$ ,  $N_B = 2$

Fig. 2.6 gives a example graph with  $N = 9$  nodes,  $L = 13$  links and  $N_B = 2$  Byzantine nodes. The  $\vec{v}_s$ ,  $\vec{v}_B$  and  $\vec{v}_d$  of the example graph are

$$\vec{v}_s(t_0) = [1 \quad -1 \quad -1 \quad -1 \quad -1 \quad 1 \quad 1 \quad -1 \quad -1]^T \quad (2.14)$$

$$\vec{v}_B(t_0) = [1 \quad -1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad -1]^T \quad (2.15)$$

$$\vec{v}_d(t_0) = [1 \quad 1 \quad -1 \quad -1 \quad -1 \quad 1 \quad 1 \quad -1 \quad 1]^T \quad (2.16)$$

At each time slot, the nodes in the graph adjust their opinions according to the opinions of their neighbor nodes. To get the state of each node in the network at the next moment, we need to know how each node is connected. The adjacency matrix provides us with this information. Each column of the adjacency matrix includes information about the

neighboring nodes of the node corresponding to that column. We can derive a local majority opinion for each node in the graph based on the adjacency matrix, the state vector, and the declared vector denoted as

$$\vec{v}_{LMO} = \text{sign}(A \cdot \vec{v}_d + \vec{v}_s) = [o_1 \quad o_2 \quad \cdots \quad o_N]^T \quad (2.17)$$

where,

$$o_i = \begin{cases} 1, & \text{if the local majority opinion for node } i \text{ is positive opinion} \\ -1, & \text{if the local majority opinion for node } i \text{ is negative opinion} \\ 0, & \text{If the number of positive and negative opinions is equal} \end{cases} \quad (2.18)$$

where  $A$ ,  $\vec{v}_s$  and  $\vec{v}_d$  are the adjacency matrix, the state vector and the declared vector, respectively. The local majority opinion of the example graph shown in Fig. 2.6 is

$$\begin{aligned} & \vec{v}_{LMO}(t_0) \\ &= \text{sign}(A \cdot \vec{v}_d(t_0) + \vec{v}_s(t_0)) \\ &= \text{sign} \left( \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ -1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ -1 \\ -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \right) = \begin{bmatrix} 0 \\ -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \end{aligned} \quad (2.19)$$

The symbols of the elements in the local majority opinion vector show the local majority opinion of the nodes in the network. The state of each node at the next time slot  $t_0 + 1$  can be inferred from the state of each node at the current time slot  $t_0$  and the local majority opinion.

$$s_i(t_0 + 1) = \begin{cases} 1, & \text{if } o_i \text{ is } 1 \\ -1, & \text{if } o_i \text{ is } -1 \\ s_i(t_0), & \text{if } o_i \text{ is } 0 \end{cases} \quad (2.20)$$

The state vector  $\vec{v}_{s,t_0+1}$  of the example graph Fig. 2.6 at the next time slot is

$$\vec{v}_s(t_0 + 1) = [1 \quad -1 \quad -1 \quad -1 \quad 1 \quad 1 \quad 1 \quad -1 \quad -1]^T \quad (2.21)$$

This is the simulation method of the dynamics of the Byzantine NCO model, by which

we can get all the states of the network from the initial to the convergence by iteration. For the fixed steady state, the steady state condition is

$$\vec{v}_s(t_{i+1}) = \vec{v}_s(t_i) \quad (2.22)$$

For the cyclic steady state with cycle period  $C$ , the steady state condition is

$$\begin{aligned} \vec{v}_s(t_{i+C}) &= \vec{v}_s(t_i) \\ \text{and} \\ \vec{v}_s(t_{i+j}) &\neq \vec{v}_s(t_i), \forall j \in [1, C-1] \end{aligned} \quad (2.23)$$

where  $C$  is the cycle length.



# 3

## CYCLIC BEHAVIOR

Imagine that Country A discovers that Country B has launched an intercontinental nuclear missile at itself, and the leaders of Country A have one hour to decide whether to launch a nuclear counterattack. Some of the leaders of Country A support an eye for an eye, while others do not want to drag humanity into destruction. Unfortunately, the leader of country A has a spy from country B. From a spy's point of view, the best way to interfere with A's decision is to keep the two factions in a constant tug-of-war, unable to make a decision in favor of their country within an hour. If the spy had studied the Byzantine NCO model he would have thought of using cyclic behavior to achieve this purpose.

A significant contribution of this project is the research on the cyclic behaviour of the NCO model. In this chapter, I will introduce and discuss typical steady states of Byzantine NCO models, give some examples of the Byzantine NCO model with a very long cycle period, and provide a general method for generating graphs with extremely long cycle periods.

### 3.1. TYPICAL STEADY STATES OF BYZANTINE NCO MODEL

In traditional NCO models, researchers believe that the opinion network eventually reaches a fixed steady state. In the fixed steady state, the opinion of each node in the network becomes fixed, and the opinion network shows a state of consensus (all nodes are of the same opinion) or coexistence (both different opinions exist). This perception holds in most cases, but in rare cases, the opinion of some nodes in the opinion network does not reach a fixed state for some specific initial state, and the whole network constantly oscillates between several states. I refer to this phenomenon as cyclic behaviour.

### 3.1.1. FIXED STEADY STATES

The fixed steady state is the most common steady state. When the NCO model is in a fixed steady state, the opinion of each node in the network will be fixed, and no more changes will occur. The NCO model reaches a fixed steady state if and only if all nodes in the network satisfy that the number of the neighbor nodes that hold the same opinion as this node plus one is more than the number of nodes that have the opposite opinion, the condition is denoted as

$$\begin{aligned} \forall n \in S_{-1} : 1 + \delta(n)_{S_{-1}} &\geq \delta(n)_{S_{+1}} \\ \forall n \in S_{+1} : 1 + \delta(n)_{S_{+1}} &\geq \delta(n)_{S_{-1}} \end{aligned} \quad (3.1)$$

where  $S_{-1}$ ,  $S_{+1}$  are the sets of nodes holding positive and negative opinions, and  $\delta(n)_{S_{-1}}$ ,  $\delta(n)_{S_{+1}}$  are the number of nodes connect to node  $n$  in this two sets.

For the Byzantine NCO model, since Byzantine nodes are always lying, this statement will become: the Byzantine NCO model reaches a fixed steady state if and only if all nodes in the network satisfy that the number of the neighbor nodes that declare to hold the same opinion as this node plus one (the node's own opinion) is more than the number of nodes that declare to have the opinion contrary to this node, denoted as

$$\begin{aligned} \forall n \in S_{-1} : 1 + \delta(n)_{S_{d,-1}} &\geq \delta(n)_{S_{d,+1}} \\ \forall n \in S_{+1} : 1 + \delta(n)_{S_{d,+1}} &\geq \delta(n)_{S_{d,-1}} \end{aligned} \quad (3.2)$$

where  $S_{-1}$ ,  $S_{+1}$  are the sets of nodes holding positive and negative opinions,  $S_{d,-1}$  and  $S_{d,+1}$  are sets of nodes declaring a positive or negative opinions, and  $\delta(n)_{S_{d,-1}}$ ,  $\delta(n)_{S_{d,+1}}$  are the number of nodes connect to node  $n$  in  $S_{d,-1}$  and  $S_{d,+1}$  sets.

If the state of the network at a specific time slot does not satisfy this condition, then the network either has not yet converged or is in a cyclic steady state.

According to the presence of the two different opinions, the steady states can be divided into two categories:

1. The consensus steady state, where all the nodes in the network are holding the same opinion.
2. The coexistence steady state, where both of the two different opinions are existing in the network.

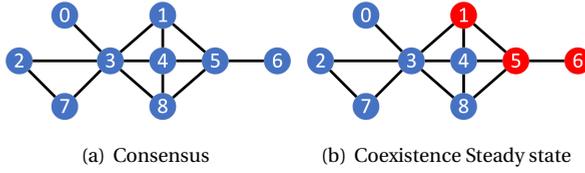


Figure 3.1: A demonstration of consensus and coexistence steady state

### 3.1.2. CYCLIC STEADY STATES

Cyclic behaviour refers to the phenomenon that in the dynamics of the NCO model, several different states appear cyclically so that the state of the system can never converge to a fixed state, like Fig. 3.2. and Fig. 3.3. Here we regard this behavior of the Byzantine NCO model as a special kind of steady state, namely a cyclic steady state.

The cycle period  $C$  of the cyclic steady state is defined as the smallest positive integer that satisfies any state  $s_t$  in the cycle is the same as the state after  $C$  iterations  $s_{t+C}$ , denoted as

$$s_t = s_{t+C} \quad \forall s_t \text{ in the cycle} \quad (3.3)$$

Cyclic behavior occurs in both the NCO and Byzantine NCO model, with the difference that the NCO model has a cycle with length of only two (I have not found examples with cycle periods longer than two, but do not exclude the possibility that they exist). The Byzantine NCO model can exhibit cyclic behavior with an extended cycle period. Cyclic behavior can be viewed as another kind of steady state.

#### CYCLIC STEADY STATES OF THE NCO MODEL

The NCO model exhibits cyclic behavior (with a cycle length of 2) if we can separate the nodes into four groups: two oscillatory groups  $S_{O_-}$ ,  $S_{O_+}$  and two static groups  $S_{-1}$ ,  $S_{+1}$ . The size of groups  $S_{O_-}$ ,  $S_{O_+}$  can not be 0, but the size of two static groups  $S_{-1}$ ,  $S_{+1}$  can. During the oscillations, nodes in the static groups do not change their opinion, whereas nodes in the oscillatory groups keep flipping their opinion at every time step. The constraints are as follows:

$$\begin{aligned} \forall n \in S_{+1} : \delta(n)_{S_{-1}} &> \delta(n)_{S_{+1}} + |\delta(n)_{S_{O_+}} - \delta(n)_{S_{O_-}}| + 1 \\ \forall n \in S_{-1} : \delta(n)_{S_{+1}} &> \delta(n)_{S_{-1}} + |\delta(n)_{S_{O_+}} - \delta(n)_{S_{O_-}}| + 1 \\ \forall n \in S_{O_+} : 1 + \delta(n)_{S_{O_+}} &\geq \delta(n)_{S_{O_-}} + |\delta(n)_{S_{+1}} - \delta(n)_{S_{-1}}| \\ \forall n \in S_{O_-} : 1 + \delta(n)_{S_{O_-}} &\geq \delta(n)_{S_{O_+}} + |\delta(n)_{S_{+1}} - \delta(n)_{S_{-1}}| \end{aligned} \quad (3.4)$$

where  $\delta(n)_{S_{-1}}$ ,  $\delta(n)_{S_{+1}}$ ,  $\delta(n)_{S_{O_+}}$ ,  $\delta(n)_{S_{O_-}}$  are the number of nodes connect to node  $n$  in these 4 groups.

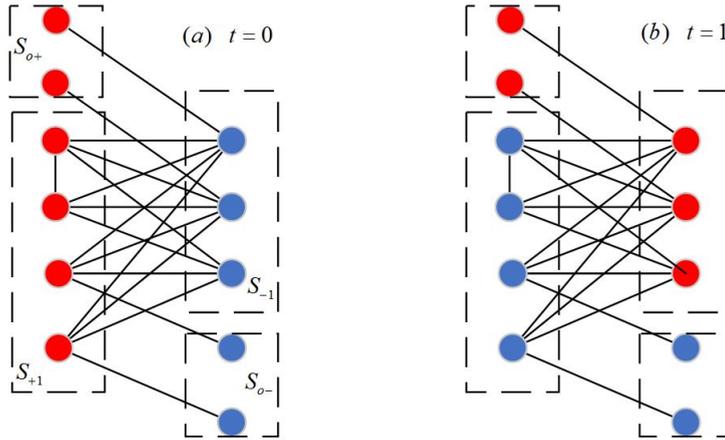


Figure 3.2: A demonstration of the cyclic steady states in the NCO model.  $S_{-1}$ ,  $S_{+1}$  are the static groups, and  $S_{O-}$ ,  $S_{O+}$  are oscillatory groups.

A demonstration of the cyclic steady states in the NCO model is shown in Fig 3.2. Group  $S_{+1}$  and  $S_{-1}$  are two oscillatory groups that hold different views,  $S_{O+}$  and  $S_{O-}$  are two static groups. At the time  $t = 0$ , all nodes in the group  $S_{+1}$  hold the positive opinion. However, the number of its neighboring nodes holding a negative opinion is larger than that holding a positive opinion plus one, so they have a tendency to turn into negative opinions; the group  $S_{-1}$  is the opposite. At time slot  $t = 1$ , group  $S_{+1}$  and  $S_{-1}$  both changed their opinions, but this has also led to a change in the local majority opinion. Then at the next time slot, they will change back to the same state as at  $t = 0$ .

### CYCLIC STEADY STATES OF THE BYZANTINE NCO MODEL

Unlike normal nodes, Byzantine nodes always declare a local minority opinion. Normal nodes drive the network to a consensus state, while Byzantine nodes drive the network to a balanced opinion state. When the network converges to the positive opinion consensus state, the sudden change of Byzantine nodes' opinions will drive the network to converge to the negative opinion consensus state, making it easier for the network to oscillate between two opinions. This property of Byzantine nodes makes the Byzantine NCO model more prone to cyclic behavior and exhibits longer cycle lengths.

Fig 3.3 shows a schematic plot of the dynamics of a cyclic case with a cycle period of 8. I find that at  $t = 3$ , as Byzantine nodes 1, 2 change their own opinions, the local majority opinion of node 5 changes and nodes 5, 4, 3 start to converge to negative opinions. However, once the network reaches a local consensus, the Byzantine node will change its opinion, and the network will turn to a positive consensus.

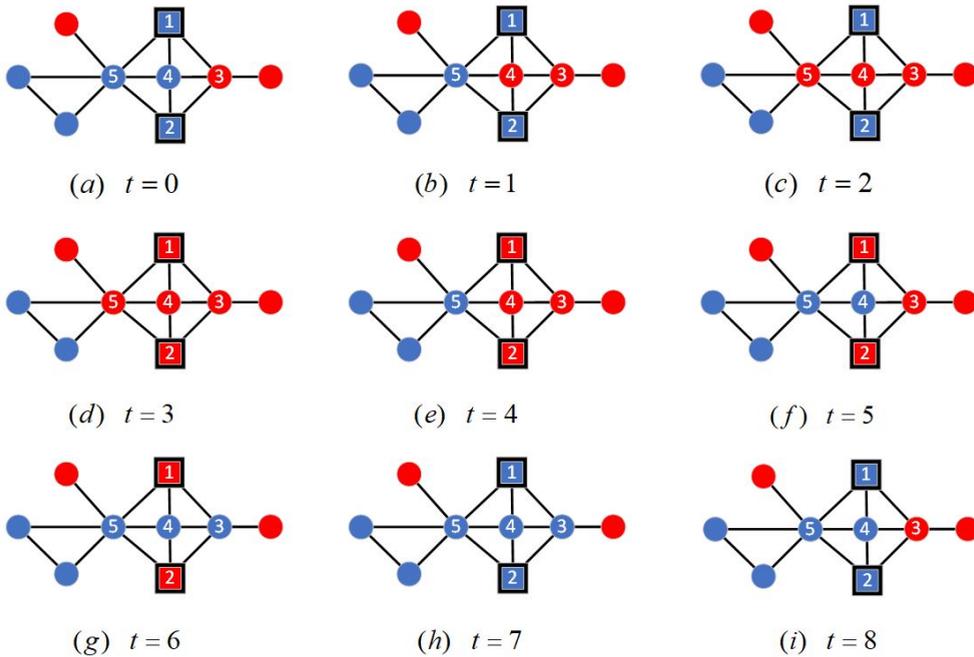


Figure 3.3: A demonstration of the cyclic steady states in the Byzantine NCO model for a network with 9 nodes. The node's color denotes the opinion of the node, red means the node holds a positive opinion  $\sigma^+$ , blue means the node has a negative opinion  $\sigma^-$ . At  $t = 0$  to  $t = 6$ , the normal nodes 3,4,5 change their opinion according to their local majority opinion. At  $t = 7$ , the Byzantine nodes 1, 2 change their opinion, then node 3's local majority opinion changes from negative to positive due to the lying nature of Byzantine nodes. At  $t = 8$ , the graph show the same state as initial state. The graph will repeat the previous dynamics.

### 3.2. CYCLIC CASES WITH LONG CYCLE PERIOD

Cyclic behaviour in the Byzantine NCO model is significantly richer than the NCO model, and to my best knowledge, there is no general rule to determine the cycle length for a given initial condition. I have found some fascinating cases in studying cyclic steady states, and select a few to show in this section.

#### SUN GRAPHS

We consider so called sun graphs, defined on  $2N$  nodes. To construct the sun graph, I start with a ring graph, consisting of  $N$  Byzantine nodes. Then  $N$  normal nodes are added, such that each normal node connects to two adjacent Byzantine nodes, see Fig. 3.4. As initial condition, one Byzantine node and a normal node it is connected to, has the negative opinion. All other nodes start with the positive opinion, see Fig. 3.4.

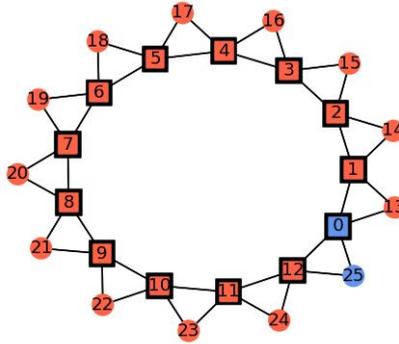


Figure 3.4: A demonstration of the sun graph with  $N = 13$

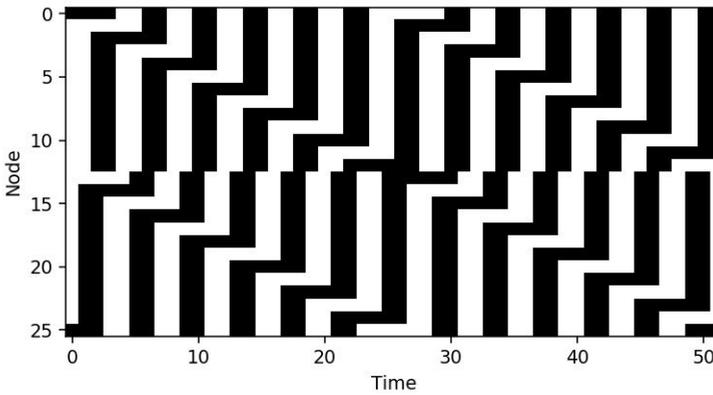


Figure 3.5: The output time series of sun graph cases in 3.4. The black color indicates a negative opinion and white color indicates the positive opinion

Fig. 3.4 shows an example of a sun graph with a cycle length of 52, and Fig. 3.5 shows the dynamics of this example. It can be shown that the cycle length for sun graphs with  $2N$  nodes satisfies

$$CycleLength = \begin{cases} 2N & \text{if } N \text{ is an even number} \\ 4N & \text{if } N \text{ is an odd number} \end{cases} \quad (3.5)$$

The advantage of the sun graph is we can generate arbitrary networks with a cycle period of  $4n$  ( $n > 2, n \in \mathbb{Z}^+$ ) in this way. However, using this method to generate long-period cyclic graphs is very inefficient because the number of nodes required to generate a long cycle is very large.

This construction is very inefficient because the period of the sun graph increases linearly with the number of nodes, which means if we want to generate a network with a

period of  $4n$ , at least  $4n$  or  $2n$  nodes is needed, half of which are Byzantine nodes.

**SMALL PATTERNS**

To find cyclic behavior with long periods, I first thought of performing exhaustive operations on some small-scale networks. The number of operations in the exhaustive enumeration operation is vast. I take small-scale graphs with  $N = 9$  nodes and  $L = 18$  links to simulate and find periodic networks with long periods. There are 33366 isomorphic different graphs with  $N = 9$  nodes and  $L = 18$ . These graphs are generated from the programs called Nauty and Traces, see [26]. At the beginning I select  $i$  nodes to hold a  $\sigma_-$  opinion and  $9 - i$  nodes to hold a  $\sigma_+$  opinion. There are

$$\binom{9}{i} = \frac{9!}{i! \cdot (9 - i)!} \tag{3.6}$$

different combinations for each number  $i$ . Thus, for each graph we have

$$\sum_{i=0}^9 \binom{9}{i} = 512 \tag{3.7}$$

different initial states. So does the Byzantine nodes settings, which means we need to do 8746696704 ( $512 \times 512 \times 33366$ ) simulations. It is very resource-intensive to exhaust all initial settings directly. Fortunately, there is a lot of redundancy, and we can save computational resources by reducing the redundancy in the algorithm. The redundancy comes from two main issues:

- 1) Repetitive operations caused by the symmetry of positive and negative opinions are an important source of operational redundancy. The symmetry of positive and negative opinions means that each initial state has an initial state symmetrical to it, and nodes in these two initial states have exactly opposite opinions. The dynamics of these two initial states with opinion symmetry also shows opinion symmetry. For the example in Fig. 3.6

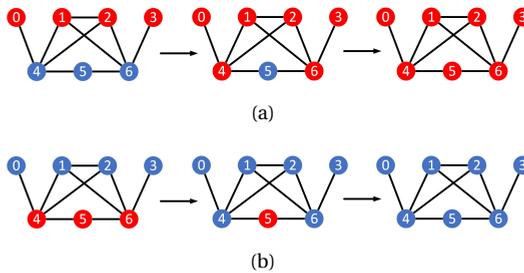


Figure 3.6: A demonstration of two symmetric dynamics

If I have the dynamics of Fig. 3.6(a) as follows

$$\begin{aligned} \vec{v}_0 &: [1 \ 1 \ 1 \ 1 \ -1 \ -1 \ -1] \\ \Rightarrow \vec{v}_1 &: [1 \ 1 \ 1 \ 1 \ 1 \ -1 \ 1] \\ \Rightarrow \vec{v}_2 &: [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1] \end{aligned}$$

there must be a symmetric version as well

$$\begin{aligned}\vec{v}_0 &: [-1 \quad -1 \quad -1 \quad -1 \quad 1 \quad 1 \quad 1] \\ \Rightarrow \vec{v}_1 &: [-1 \quad -1 \quad -1 \quad -1 \quad -1 \quad 1 \quad -1] \\ \Rightarrow \vec{v}_2 &: [-1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1]\end{aligned}$$

- 2) Repeated operations of the known convergence process also contribute to operational redundancy. A state  $s_1$  may already be present in the dynamics with a certain initial state  $s_0$ , so the dynamics simulation with this state as the initial state will result in unnecessary duplicate operations. For example

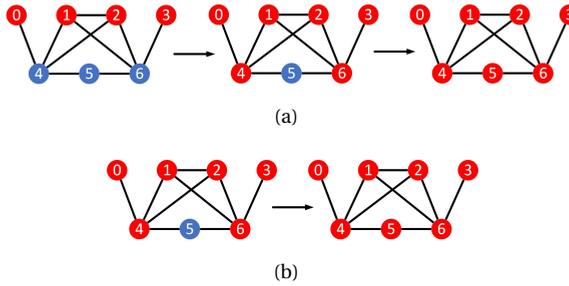


Figure 3.7: The probability of occurrence of cyclic behavior for Graphs with  $N = 7$ ,  $L = 10$

if we already have the dynamics in Fig. 3.7(a)

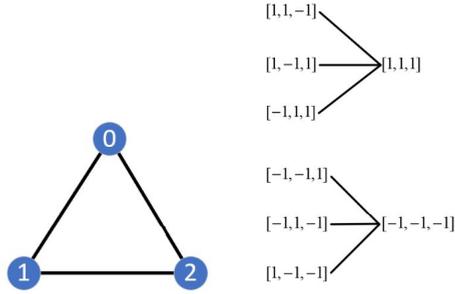
$$\begin{aligned}\vec{v}_0 &: [1 \quad 1 \quad -1 \quad -1 \quad 1 \quad -1 \quad 1] \\ \Rightarrow \vec{v}_1 &: [1 \quad 1 \quad -1 \quad 1 \quad 1 \quad 1 \quad 1] \\ \Rightarrow \vec{v}_2 &: [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]\end{aligned}$$

there is no need to operate dynamics with the initial state  $s_1$  in Fig. 3.7(b)

$$\begin{aligned}\Rightarrow \vec{v}_1 &: [1 \quad 1 \quad -1 \quad 1 \quad 1 \quad 1 \quad 1] \\ \Rightarrow \vec{v}_2 &: [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]\end{aligned}$$

I adopt the following strategy to reduce the number of operations:

- 1) Operate only on half of the initial state space (for the case of  $N=9$ ,  $L=18$  diagram, that is, only the initial states with 0 to 4 negative nodes is calculated)
- 2) If a state already occurs in the dynamics of a previous state, the state is no longer simulated.
- 3) Use trees to store all dynamics of a graph to save memory resources. For example, for the graph in Fig. 3.8(a), we have trees like Fig. 3.8(b)



(a) A demonstration graph (b) Trees of the graph

Figure 3.8: A demonstration graph and the trees of this graph.

This strategy cuts out enough redundant operations. In this way we can save about 80% of computing time.

Graph Pattern	(a)	(b)	(c)
Cycle Length	5	7	11
Graph Pattern	(d)	(e)	(f)
Cycle Length	13	17	19
Graph Pattern	(g)	(h)	(i)
Cycle Length	23	44	50

Figure 3.9: The long-cycle pattern table for graphs with  $N = 9, L = 18$

By exhaustive operations I obtain many patterns with long periods. I show several examples of long cycle periods in Fig. 3.9. The longest cycle period is 50.

### SPECIAL GRAPHS

In the course of studying the Byzantine NCO model, I found a diamond graph  $D_7$  (shown in Fig. 3.10).

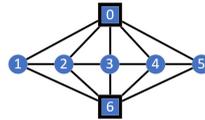


Figure 3.10: The diamond graph  $D_7$

The unique feature of this diamond graph is that all normal nodes in this graph are at the threshold of changing opinions, when the diamond graph is at a fixed steady state. As the dynamics in Fig. 3.11 shows, a slight perturbation can cause nodes 1 or 5 to change their opinions. Then a chain reaction will occur, with nodes 2, 3, and 4 changing their opinions one after another, eventually causing the opinions of the whole pattern to change.

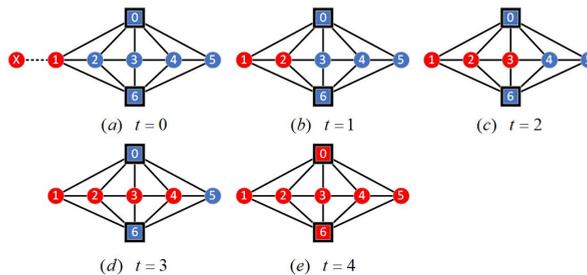


Figure 3.11: The dynamics of the cycle of the special diamond graph

Using this property of special patterns, we can generate networks with long cycles by chaining several  $D_7$ . More specifically, we chain the outer nodes in the middle (nodes 1 and 5 in Fig. 3.10) of a diamond graphs with the another diamond graph and chain several diamond graphs one by one, like Fig. 3.12.

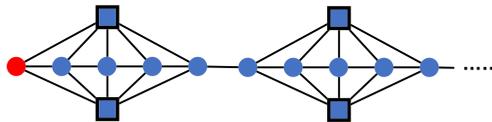


Figure 3.12: A demonstration of how the different diamond graphs chain together

Fig. 3.13 shows a network with 7  $D_7$ , and the cycle length of this network is 136. There is only one node in the graph holding  $\sigma_+$  opinion (red) at initial state, and the other nodes are all holding a  $\sigma_-$  opinion (blue).

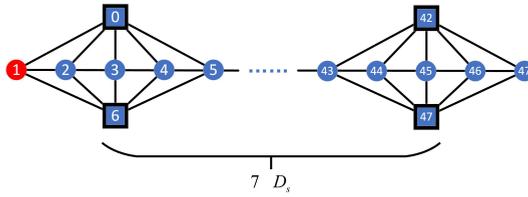


Figure 3.13: A demonstration graph with 7  $D_7$  whose cycle length is 136

After 195 iterations, the example graph will finally reach a cyclic steady state with a cycle length of 136. The dynamics of the cycle of this example network are depicted in Fig. 3.13 is

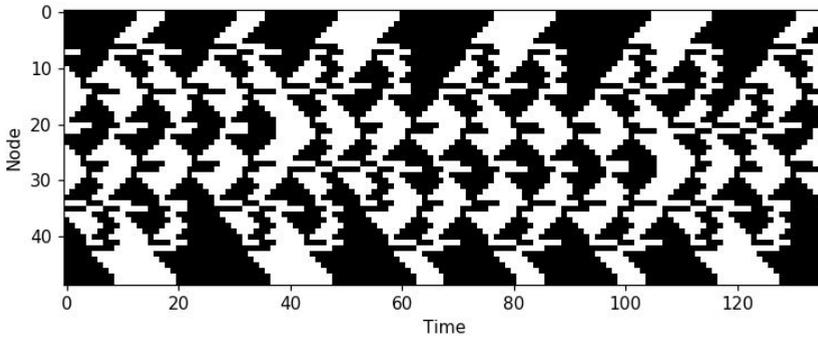


Figure 3.14: The dynamics of the cycle of the demonstration graph in Fig. 3.13

When studying the cycle period of such networks, a mistake in connecting two  $D_7$  graphs led us to another configuration with a long cycle period, which is shown in Fig 3.15. The cycle length of this type of network tends to become longer if we add more "mistakes"; for example, the network with 6  $D_7$  and two mistakes has a cycle length of 190, for which the dynamics are shown in Fig 3.16, but for the network without mistakes, the cycle length is 102.

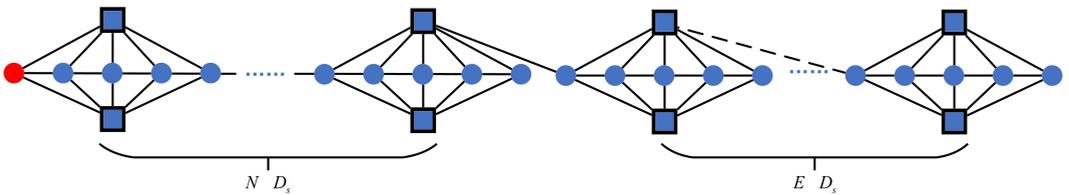


Figure 3.15: The special graph model with  $M + E D_7$  and  $E$  mistakes

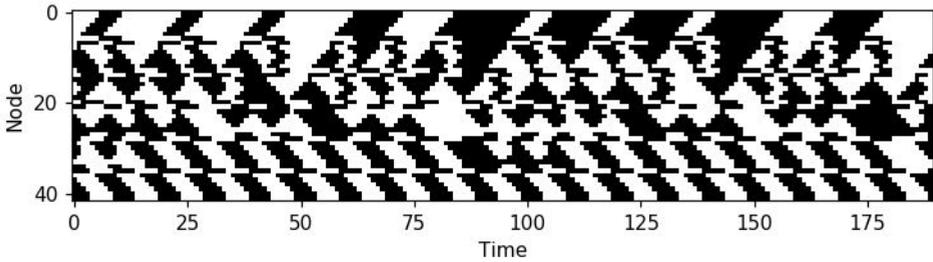


Figure 3.16: A demonstration of a graph consists of 6  $D_7$  and 2 mistakes

The cycle periods of special cases are so complex that we cannot obtain a general formula for their cycle period. I have simulated the dynamics of this type of network and got the following cycle periods depicted in Table 3.1. As is shown, the largest cycle period in this table is 22131760, which occurs in the network with 21  $D_7$  and one mistake. Also, I find that the cycle length of this type of network does not increase linearly with the number of  $D_7$ . I hope that future researchers will solve the mystery of the cycle lengths of this type of network.

Number of $D_7(M+E)$	2	3	4	5	6	7	8
Cycle length	10	1	20	26	102	136	28
Convergence time	5	16	6	88	17	195	159
Cycle length( $E = 1$ )	1	1	1	38	1178	520	174
Cycle length( $E = 2$ )		1	10	1	190	664	366
Cycle length( $E = 3$ )			1	10	1	190	170
Number of $D_7(M+E)$	9	10	11	12	13	14	15
Cycle length	3372	28	2615	4644	179	1113	28
Convergence time	577	313	173	7890	10228	443	6706
Cycle length( $E = 1$ )	5388	864	11832	612	45918	110490	56022
Cycle length( $E = 2$ )	402	4796	9514	24620	38360	69602	61308
Cycle length( $E = 3$ )	1	7152	1490	6320	89974	26449	93278
Number of $D_7(M+E)$	16	17	18	19	20	21	22
Cycle length	18	179	5682	28	126	28	3769654
Convergence time	1325	17710	343,632	56256	160442	3198374	631058
Cycle length( $E = 1$ )	756	179	347177	179	6620424	722301	22131760
Cycle length( $E = 2$ )	566850	390278	1952400	314951	76835	5000520	8229700
Cycle length( $E = 3$ )	128010	89352	255249	918374	1138078	1207604	11880976

Table 3.1: Cycle length for graphs consisting of concatenated diamond graphs  $D_7$

Due to the special network's long period and convergence time, I often meet memory errors during simulations. For example, for the network with 34 $D_7$ , the convergence time is about 280000000. I had to optimize the data type of the network state to reduce

the space complexity of the algorithm. The detail of the optimized algorithm is given in the Appendix A.

### 3.2.1. COMBINED NETWORK

In addition to the long-period patterns found through simulation, I also developed a network combination method that can generate long-period networks. In network combination we combine several different networks with different cycle periods into one connected network. Network combination is based on the idea that longer-cycle networks can be generated by combining different component networks with different cycles without breaking the cyclic behavior of each component network.

If we want to combine  $m$  graphs  $G_1, G_2, \dots, G_m$  with cycle periods  $C_1, C_2, \dots, C_m$ , the cycle period  $C_{G_c}$  of the combined network is the least common multiple  $lcm(C_1, C_2, \dots, C_m)$ . The proof is as follows:

$$\begin{aligned}
 s_{G_1}(t + x_1 C_1) &= s_{G_1}(t) \\
 s_{G_2}(t + x_2 C_2) &= s_{G_2}(t) \\
 &\dots \\
 s_{G_m}(t + x_m C_m) &= s_{G_m}(t)
 \end{aligned} \tag{3.8}$$

where  $C_1, C_2, \dots, C_m$  are the cycle period of graphs  $G_1, G_2, \dots, G_m$  and  $x_1, x_2, \dots, x_m \in \mathbb{N}^+$ .

The combined network follows:

$$s_{G_c}(t + C_c) = s_{G_c}(t) \tag{3.9}$$

where  $C_c$  is the cycle period of the combined network  $G_c$ . Since the combination does not break the cyclic behavior of each component graphs, there exist  $x_1, x_2, \dots, x_m \in \mathbb{N}^+$  such that

$$\begin{aligned}
 x_1 C_1 &= C_c \\
 x_2 C_2 &= C_c \\
 &\dots \\
 x_m C_m &= C_c
 \end{aligned} \tag{3.10}$$

which means  $C_c$  is the common multiple of  $C_1, C_2, \dots, C_m$ . Because  $C_c$  is the smallest positive integer that satisfies the above conditions, the cycle period of the combined network  $C_c$  is the least common multiple  $lcm(C_1, C_2, \dots, C_m)$ .

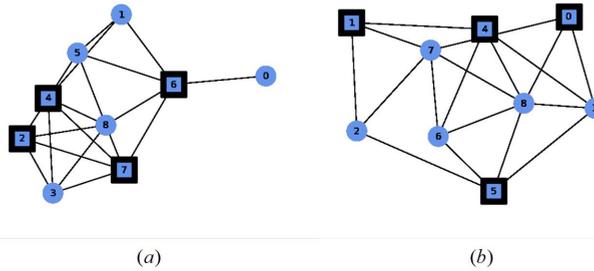


Figure 3.17: Two separate networks with cycle length 23 (left) and cycle length 44 (right).

If I simply consider as the combination the union of  $G_1, G_2, \dots, G_m$ , without any connections between the different graphs  $G_i$ , then  $lcm(C_1, C_2, \dots, C_m)$  is the cycle period of the resulting, disconnected, network. However, adding links between the component graphs might break the cyclic behavior.

Fig 3.17 gives two networks with cycle periods of 23 and 44, respectively. By connecting these two patterns, we can obtain a network with a longer cycle period, which equals the least common multiple of the cycle periods of the individual networks [27]. In most cases, the cyclic conditions of the component graphs with long cycle period are very demanding. Adding nodes or connections can easily break the cyclic behavior of the component graphs. Thus, simple adding a link between any two nodes in the networks, such as done in Fig. 3.18, can break the cyclic behaviour of the second component graph, and the state of the nodes in the second component graph become fixed. The cycle period of Fig. 3.18 is 23.

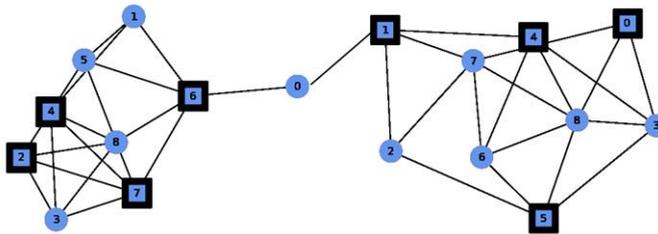


Figure 3.18: A demonstration of direct combined network

We came up with a way to solve this problem by adding mirrors. More specifically, by adding a mirror of each component network, where the mirror holds an opposite state vector to neutralize the effect of each component network on the cyclic behaviour of the other component network, like Fig. 3.19 shows.

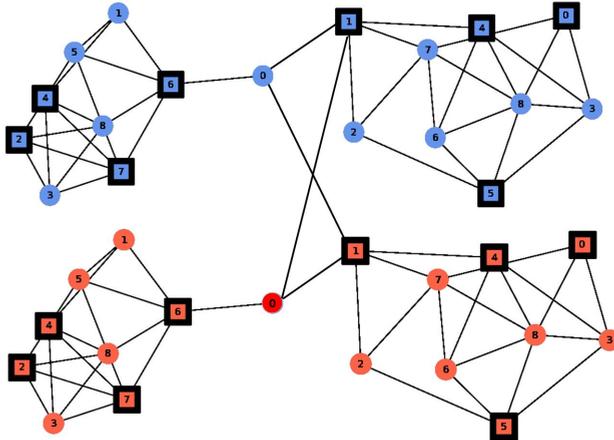


Figure 3.19: A demonstration of the combination of two graphs in Fig. 3.17. The cycle period is  $23 \times 44 = 1012$ .

Using this approach to combining networks, we do not need to consider the impact of adding connections on the cyclic behavior of the component graphs. We can indiscriminately add connections between the component graphs and their mirrors. We just need to make sure that the connections are added symmetrically to the component graphs and their mirrors. Any component graph group can be connected in this way.

Using the network combination, we can generate networks with ultra-long cycle periods. For example, as is shown in Fig 3.20, a combined network is established by combining five patterns with cycle periods 17, 19, 21, 23, 50 and their mirrors. The cycle period of the combined network is  $lcm(17, 19, 21, 23, 50) = 7800450$ .

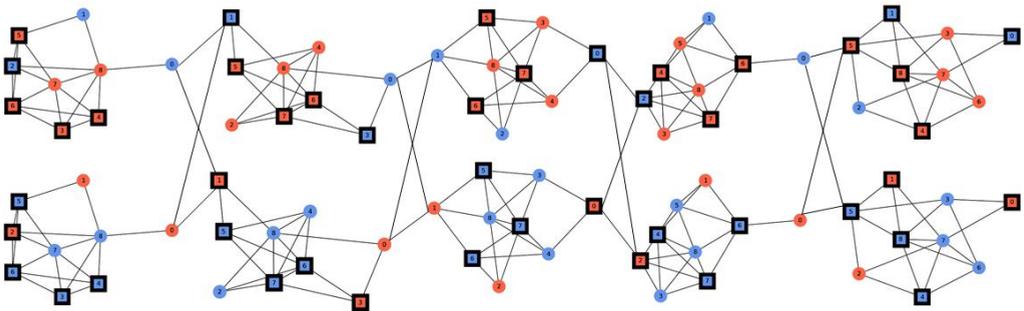


Figure 3.20: The combination network of the 5 graph patterns with a cycle period of 7800450.

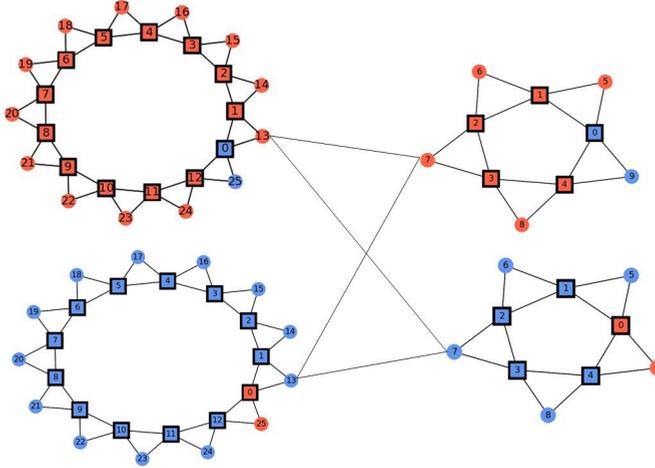
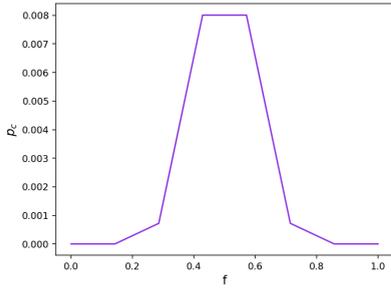


Figure 3.21: The combination network of two sun graphs with cycle periods 20 and 52. The cycle period of this network is 260.

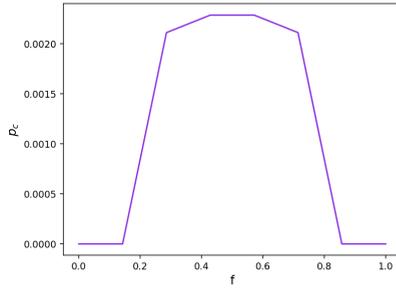
The cycle period of the combined network is the least common multiple of the cycle period of each component graphs. The sun graphs provides us with networks with a cycle period of  $4n$  ( $n \geq 2, n \in \mathbb{Z}$ ), where  $n$  can be any integer number. The least common multiple of prime numbers is the product of them. If we combining  $m$  sun graphs with cycle periods  $4n_1, 4n_2, \dots, 4n_m$ , where  $n_1, n_2, \dots, n_m$  are all prime numbers, we obtain a combination network with a cycle period of  $lcm(4n_1, 4n_2, \dots, 4n_m) = 4 \times \prod_{i=1}^m n_i$ . The example in Fig. 3.21, gives a combination network of two sun graphs with cycle periods 20 ( $4 \times 5$ ) and 52 ( $4 \times 13$ ). The cycle period of this combination network is 260 ( $4 \times 5 \times 13$ ). The sun graph provides us with an infinite number of primes. Therefore, using network combinations we can efficiently generate networks with infinitely long cycle period.

### 3.3. OCCURRENCE OF CYCLIC BEHAVIOR

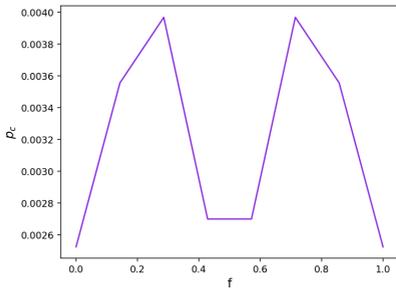
The introduction of Byzantine nodes not only produces long cycles but also changes the conditions for the occurrence of cyclic behavior. In Fig. 3.22, I plot the occurrence probability of occurrence of cyclic behaviour  $p_c(f)$  versus the initial  $\sigma_-$  opinion fraction  $f$  for a different number of Byzantine nodes  $N_B$ . I find that for the traditional NCO model, cyclic behavior is more likely to occur when the number of positive and negative opinions is balanced. As the number of Byzantine nodes increases, the peak of the curve begins to move to the left and right sides. For  $N_B \geq 3$ , the peak of the curve moves to the leftmost and rightmost sides, which means cyclic behavior is more likely to occur when one opinion prevails in the initial state.



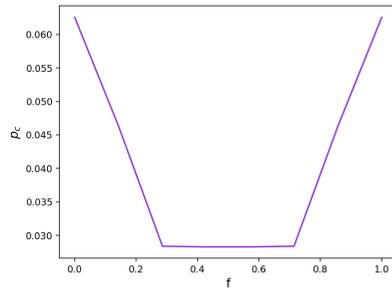
(a) No-Byzantine nodes



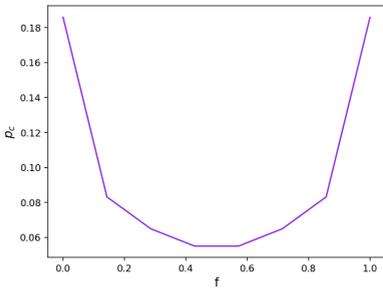
(b) 1 Byzantine node



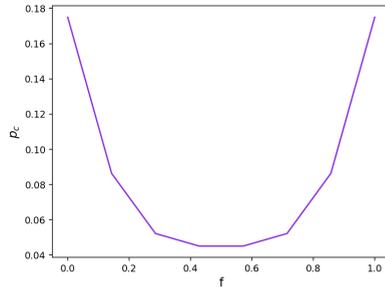
(c) 2 Byzantine nodes



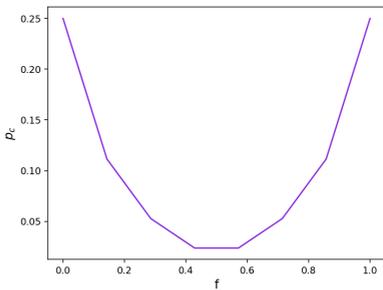
(d) 3 Byzantine nodes



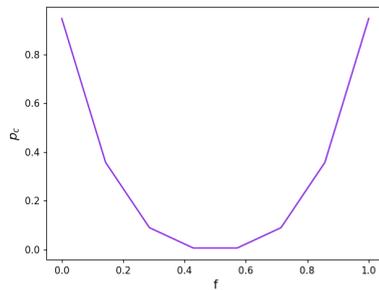
(e) 4 Byzantine nodes



(f) 5 Byzantine nodes



(g) 6 Byzantine nodes



(h) 7 Byzantine nodes

Figure 3.22: The probability of occurrence of cyclic behavior for all graphs with  $N = 7$  nodes and  $L = 10$  links. (There are 132 isomorphic different graphs obtained by using nauty and Traces [26].)

In order to test our findings, I further extend the study to some larger-scale networks. I perform simulations on ER networks with  $N = 100$ ,  $p = 0.047$  (the threshold of connection  $p_c$  is 0.04605. I set  $p$  a value larger than  $p_c$  to guarantee that the network is connected.) and SF networks (with  $N = 100$ ,  $\lambda = 3$  and  $k_{\min} = 2$ ).  $N_B$  nodes are randomly selected to be Byzantine nodes, and initial opinion  $\sigma_-$  and  $\sigma_+$  are randomly assigned to all the  $N$  nodes with a fraction of  $f$  and  $1 - f$  respectively ( $f = n/N$ , where  $n$  is the number of  $\sigma_-$  nodes) at time  $t = 0$ . Then I perform the simulation until the steady state is reached. For every initial opinion fraction value  $f$ , 20000 simulations are performed and the number of times cyclic behavior occurs is recorded.

Byzantine node settings include the number of Byzantine nodes and the selection strategy. I design the Byzantine nodes selection strategies according to the degree of the nodes. Byzantine nodes can cause their neighbouring nodes to misjudge their local majority opinion. The nodes with higher degree have more neighbors, which means they can influence more nodes. To find out the magnitude of the effect of the degree of Byzantine nodes on the behavior of the Byzantine NCO model, I take three different strategies to lay out the Byzantine nodes:

- 1) Strategy I: Randomly select  $N_B$  nodes to be Byzantine nodes.
- 2) Strategy II: Select  $N_B$  nodes with highest degree to be Byzantine nodes.
- 3) Strategy III: Select  $N_B$  nodes with lowest degree to be Byzantine nodes.

From Fig. 3.23(a), I find that the  $p_c(f)$  curve has a peak at  $f = 0.5$ , and the network show a cyclic behavior in the interval (0.36,0.66), which roughly overlap with the coexistence interval (which is introduced in Chapter 4). Outside this interval the network tends to reach a consensus steady state, where only the majority opinion exist. When only one opinion exists at the steady state, no cycles can occur.

For  $N_B \leq 70$ , the probability of occurrence of cyclic behavior is less than 2% for any initial opinion fraction  $f$ . As the number of Byzantine nodes increases, the interval of occurrence of cyclic behaviour expands to both sides, and the peak of the curve gradually moves to  $f = 0$  and  $f = 1$ . When all nodes in the network are Byzantine nodes, the network reaches a cyclic steady state in all cases in the interval  $f \in [0, 0.2)$  and the interval  $f \in (0.8, 1]$ . In this interval, most of the nodes in the network start with the same opinion. Due to the lying nature of Byzantine nodes, most of the nodes in the network will misjudge the local majority opinion and thus change their opinion, at which the changed opinion becomes the majority opinion again. The node's opinion will continue to change, oscillating between the two opinions and reaching a cyclic steady state.

In the convex part of the curve, the initial state of the ratio of these two opinions is balanced, and there is no absolute majority of opinions. In this state, the network is more stable and has a larger possibility to not oscillate.

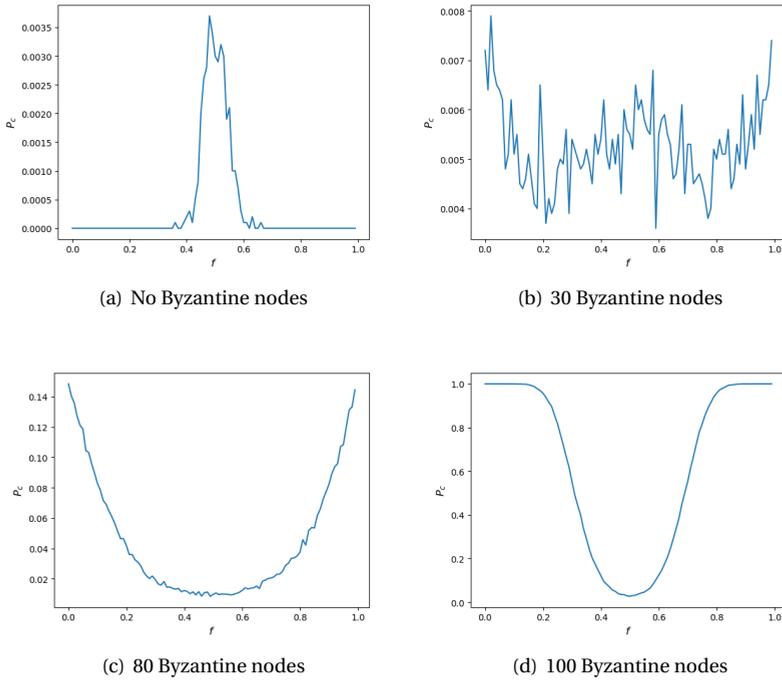


Figure 3.23: The probability of occurrence of cyclic behavior for ER networks with  $N = 100$ ,  $p = 0.047$ , and the placement of Byzantine nodes following Strategy I.

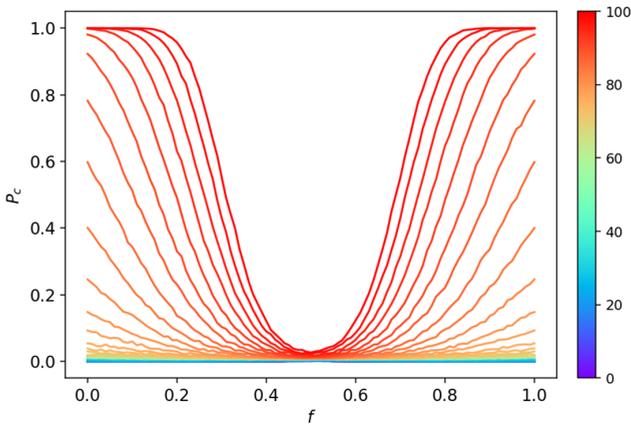


Figure 3.24: The probability of occurrence of cyclic behavior with a different number of Byzantine nodes (different line colours) for ER networks with  $N = 100$ ,  $p = 0.047$ , and the placement of Byzantine nodes following Strategy I.

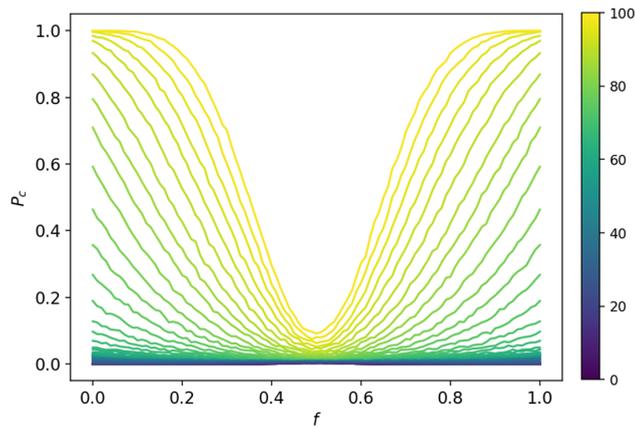


Figure 3.25: The probability of occurrence of cyclic behavior with a different number of Byzantine nodes for SF networks with  $N = 100$ ,  $k_{\min} = 2$ ,  $\lambda = 3$ , and the placement of Byzantine nodes following Strategy I.

Fig. 3.24 and Fig. 3.25 show that when the number of Byzantine nodes is larger than 70, the probability of occurrence of cyclic behavior increases significantly. For  $N_B \leq 70$ , the probability of occurrence of cyclic behavior is less than 5% for any  $f$ .

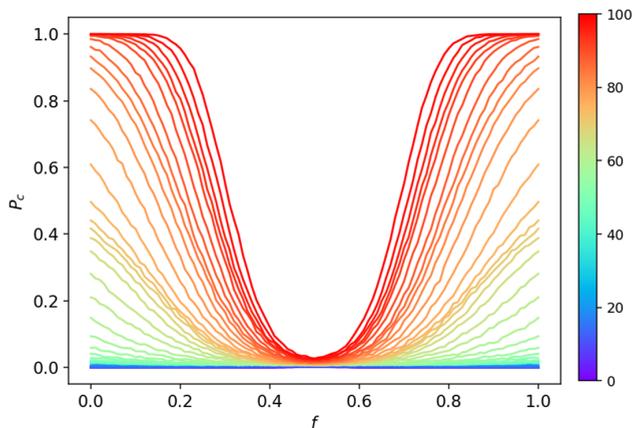


Figure 3.26: The probability of occurrence of cyclic behavior with a different number of Byzantine nodes (different line colours) for ER networks with  $N = 100$ ,  $p = 0.047$ , and the placement of Byzantine nodes following Strategy II.

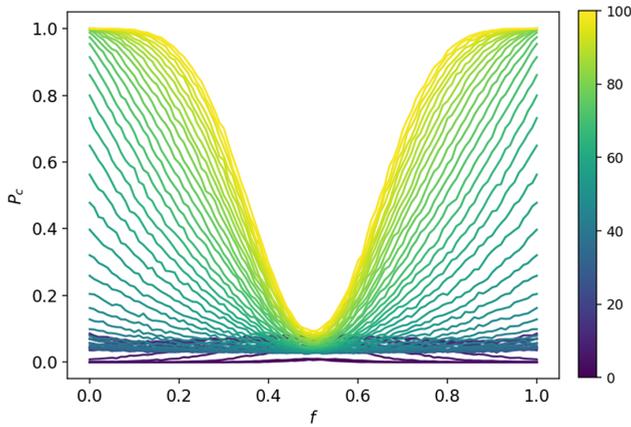


Figure 3.27: The probability of occurrence of cyclic behavior with a different number of Byzantine nodes for SF networks with  $N = 100$ ,  $k_{\min} = 2, \lambda = 3$ , and the placement of Byzantine nodes following Strategy II.

Fig 3.26 and Fig 3.27, indicate that selecting nodes with high degree as Byzantine nodes can better induce cyclic behavior than randomly selecting nodes as Byzantine nodes.

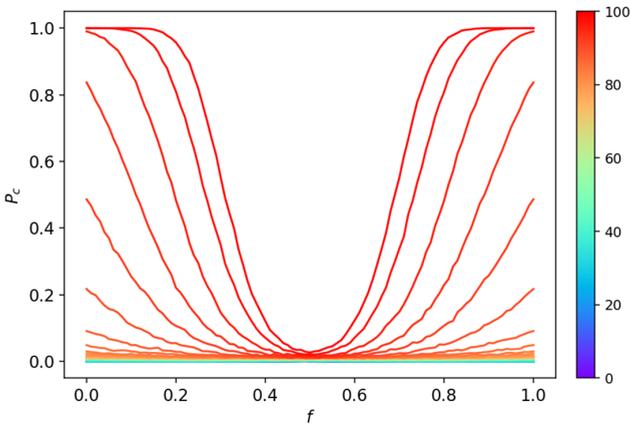


Figure 3.28: The probability of occurrence of cyclic behavior with a different number of Byzantine nodes (different line colours) for ER networks with  $N = 100$ ,  $p = 0.047$ , and the placement of Byzantine nodes following Strategy III.

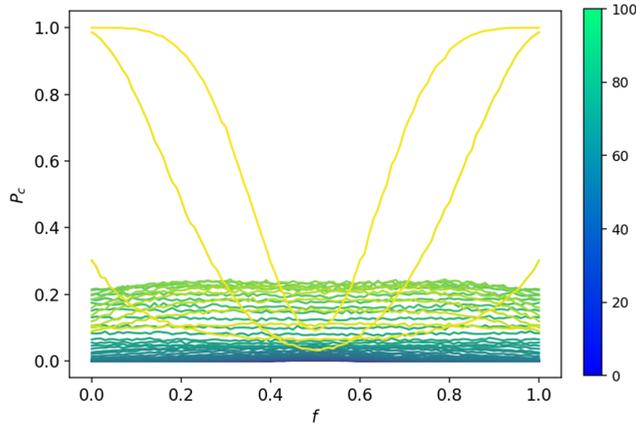


Figure 3.29: The probability of occurrence of cyclic behavior with a different number of Byzantine nodes for SF networks with  $N = 100$ ,  $k_{\min} = 2, \lambda = 3$ , and the placement of Byzantine nodes following Strategy III.

Finally, Fig 3.28 and Fig 3.29, show that Strategy III is less effective than Strategy I in promoting the cyclic behavior. Fig 3.29 shows that unless the two nodes with the largest degree in the network become Byzantine nodes, the  $p_c \leq 0.2$  for  $f \in [0, 1]$ , and the system can still reach a fixed steady state in most cases.

The above experiments show that nodes with a high degree have a greater impact on the probability of occurrence of cyclic behavior. For the same number of Byzantine nodes, taking strategy II has the greatest impact on the probability of occurrence of cyclic behavior. For ER networks, the degree distribution follows a binomial distribution and the degree of most nodes is concentrated around the mean degree value. However, for the SF network, the degrees of nodes in the network follow a power-law distribution, with most nodes having low degrees and a small number of nodes having very high degrees. Comparing ER and SF networks, the difference in degree between different nodes in SF networks is larger, and the difference in the experimental results when different strategies are taken is more significant.

Take the nuclear war story. For country B, they should bribe the more influential leaders in the decision-making hierarchy of country A who have more connections with other leaders and the spies should declare an opinion opposite to the local majority opinion. For country A, they need to ensure that key leaders, which are the leaders who can influence most of the other leaders, are not bought off, or build a more balanced leadership structure, where there are not a few leaders who can decide the big picture alone.

# 4

## OPINION DISTRIBUTION AT STEADY STATE

It does not always seem feasible to keep the decision-makers in Country A in a constant tug-of-war. Interfering with the outcome of a vote so that the decision-makers cannot make a decision in the interest of country A is also a very effective way to attack. The introduction of Byzantine nodes impacts the final opinion fraction, which can be used to develop attack and defense strategies.

For the NCO model, the final opinion fraction is an essential element of steady-state behavior because the final opinion fraction reflects which opinion is the majority opinion in the steady state. As mentioned in the previous sections, the NCO model has two opinion kinds of distributions when reaching steady state: consensus and coexistence. In the consensus state, all nodes in the network will hold the same opinion, while in the coexistence state, both opinions can coexist.

Shao *et al.* finds that there is a critical threshold of consensus and coexistence [9]. When the ratio of minority opinions in the initial state is less than this critical threshold, the network tends to reach a consensus. In contrast, the minority and majority opinions tend to coexist when it is greater.

This chapter explores the effect of the introduction of Byzantine nodes on the final opinion fraction and studies the critical threshold in the Byzantine NCO model.

### 4.1. FINAL OPINION FRACTION

The NCO model is a model used to study opinion formation. One of the essential elements I am concerned about for opinion formation is the number of people holding each of the two different opinions at steady state.

The final opinion fraction is defined as the percentage of the number of nodes holding a specific opinion among all nodes when the NCO model reaches a steady state, denoted as

$$F = n_{\sigma} / N \quad (4.1)$$

where  $n_{\sigma}$  is the number of nodes holding  $\sigma$  opinion ( $n_{\sigma_-}$  and  $n_{\sigma_+}$  are the number of nodes holding negative and positive opinions, respectively). Here I set  $\sigma$  to be the negative opinion  $\sigma_-$ .

For the cyclic steady state, the final opinion fraction is the mean value of the opinion fraction of each state in the cyclic steady state, denoted as

$$F = \frac{\sum_{i=1}^C n_{\sigma_i} / N}{C} \quad (4.2)$$

where  $C$  is the length of the cycle and  $n_{\sigma_i}$  is the number of nodes holding  $\sigma$  opinion at the  $i$ th state in the cycle.

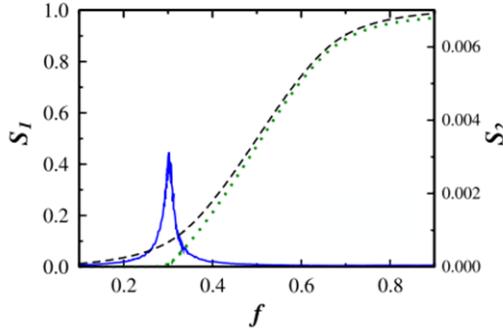


Figure 4.1: Plot of the fraction of  $\sigma_-$  nodes  $F$  (dashed line) in steady state as a function of  $f$  for an ER network with  $N = 10^5$  and  $p = 4 \times 10^{-6}$ . This figure is a copy of the plot in Shao's *et al.* paper. [9]

Shao *et al.* found that the fraction of one opinion at steady state  $F$  is positively correlated with the fraction of that opinion  $f$  at the initial state, which is not difficult to understand. What is surprising in Shao *et al.*'s study is that  $F$  does not increase smoothly with increasing  $f$  but increases smoothly in small increments before increasing sharply at some point with increasing  $f$  as Fig 4.1 shows.

In order to study the opinion distribution at steady state, I first set a certain number of nodes in the network to be Byzantine nodes. At the beginning, two opinions (denoted as  $\sigma_-$  and  $\sigma_+$ ) are randomly assigned to all nodes: nodes are assigned a negative opinion  $\sigma_-$  with an initial opinion fraction  $f$  and positive opinion  $\sigma_+$  with an initial opinion fraction  $1 - f$  [28], [29]. Then, the number of  $\sigma_-$  nodes at steady state is measured.

### 4.1.1.1. EXHAUSTIVE RESEARCH ON GRAPHS WITH $N=7, L=10$

I performed an exhaustive research on 132 small size graphs( $N=7, L=10$ ) with different initial state and Byzantine nodes settings [26].

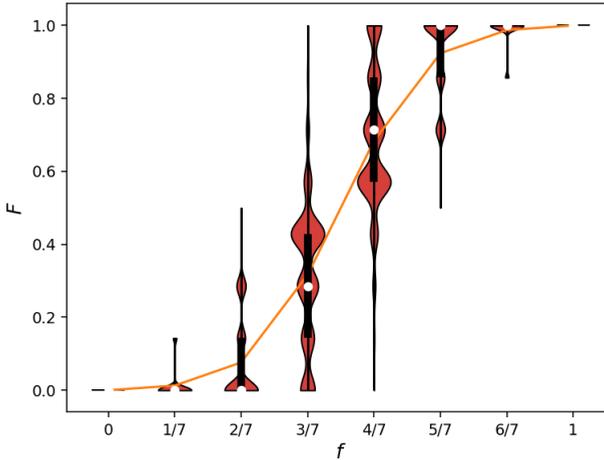
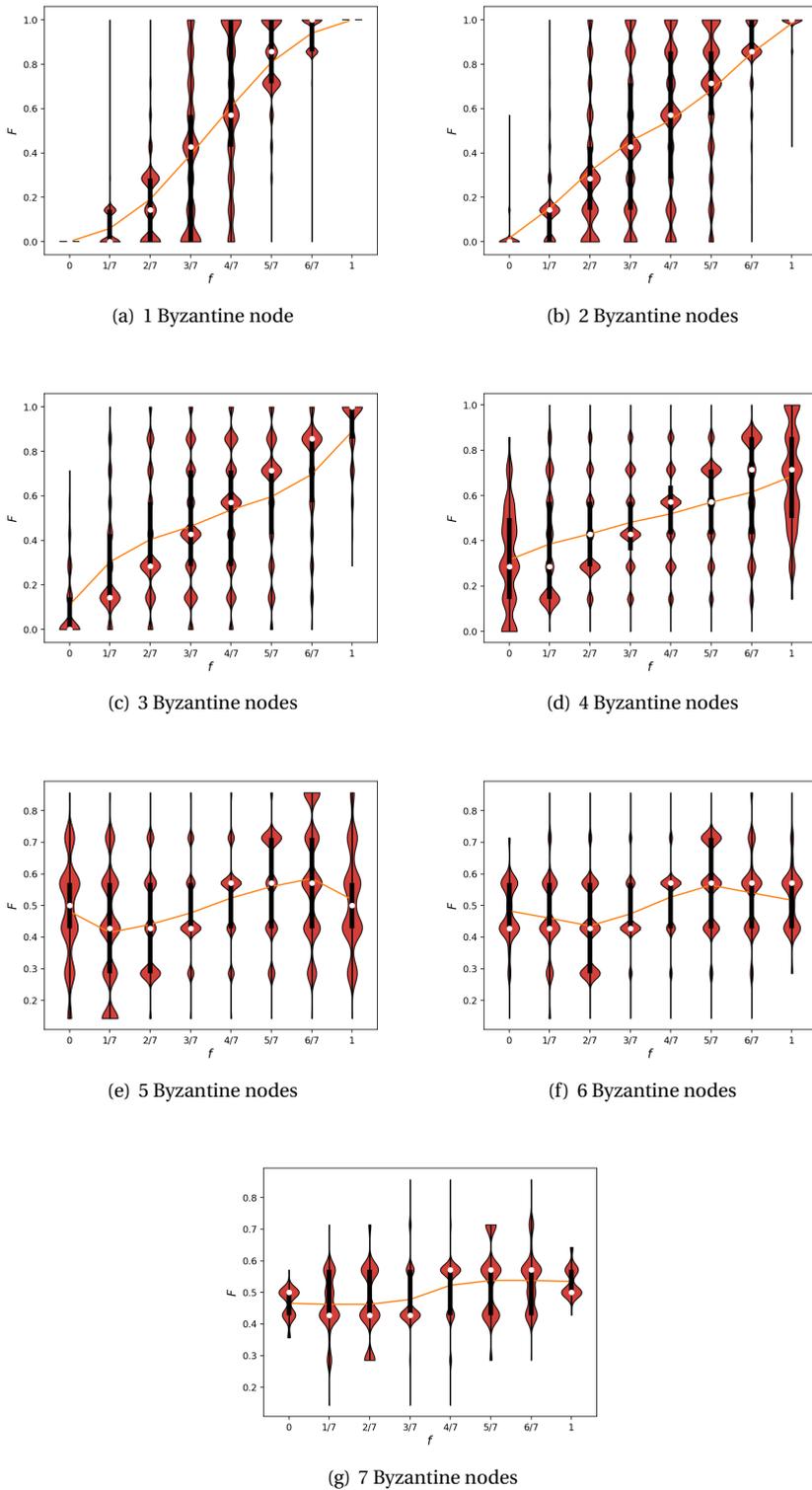


Figure 4.2: Violin plot of graphs with  $N = 7, L = 10$  and  $N_B = 0$ , the orange line shows the average value of  $F$  for different  $f$ , the red area shows the distribution of  $F$  for different  $f$ . The white point is the median of  $F$  for different  $f$ .

Fig. 4.2 shows the distribution of the number of nodes with the negative opinion  $\sigma_-$  at a steady state for the Non-Byzantine-node network. The orange curve and the red violin plot show the mean value and distribution of  $F$  at different  $f$ . I find that  $F$  is a monotonically increasing function of  $f$  with a symmetry around  $(f, F) = (0.5, 0.5)$ . For  $f \in \{0, 1/7, 2/7\}$ , the  $\sigma_-$  is the minority opinion, graphs tend to reach a positive consensus, and only a few cases reach a consensus. For  $f \in \{3/7, 4/7\}$ , the majority opinion and minority opinion are close in number, and the coexistence is reached in the majority of cases. There is a sharp increase for  $F$  at  $f \in \{2/7, 3/7\}$ ; the tendencies have also changed from consensus to coexistence.

From Fig. 4.3, I find that as more and more Byzantine nodes are added to the graph, the mean curve of  $F$  becomes flatter and flatter, which means the graphs are more likely to reach a coexistence stable or cyclic steady state. I also find that the correlation between  $f$  and  $F$  becomes weaker with increasing Byzantine nodes. As is shown in Fig. 4.3(e), even if there are no  $\sigma_-$  nodes initially( $f = 0$ ), there are still many cases that reach a steady state where the majority opinion is  $\sigma_-$ .

Figure 4.3: Violin plot of graphs with  $N = 7$ ,  $L = 10$  and different number of Byzantine nodes

From Fig. 4.3(d) I find that when there is a significant number of Byzantine nodes in the network, due to the lying nature of Byzantine nodes, the majority opinion presented by the network is the opposite of the real majority opinion, causing the opinion to flip and the initial majority opinion to become the minority opinion.

Fig. 4.4 gives an example of the introduction of Byzantine nodes reversing the majority opinion in a graph. The graph reaches a steady state and the majority opinion changes from  $\sigma_+$  to  $\sigma_-$ .

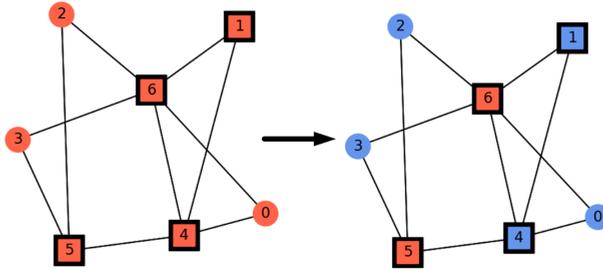


Figure 4.4: A demonstration case, where the introduction of Byzantine nodes reverses the majority opinion.

Fig. 4.3(a) to Fig. 4.3(d) show that for  $f = 0$  the main lobe of the violin plot is around  $F = 0$ , which means that for graphs with a few Byzantine nodes, the graph still tends to reach a consensus state when one opinion holds an absolute majority position. However, as the number of Byzantine nodes increases, it becomes increasingly difficult to reach a consensus state, which means that coexistence is more stable than consensus for the Byzantine NCO model.

Even when all nodes in the network are Byzantine nodes, as shown in Fig. 4.3(g), almost all cases reach a coexistence fixed steady state or cyclic steady state, and very few reach a consensus steady state. In the Byzantine NCO model, the nodes in the graph cannot correctly judge their local majority opinion since Byzantine nodes will confuse other nodes. Therefore, for the Byzantine NCO model, coexistence is a more stable state than consensus when a significant number of the nodes are Byzantine.

As the number of Byzantine nodes increases, the mean and median of  $F$  corresponding to different  $f$  get closer to 0.5, suggesting that the introduction of Byzantine nodes plays a role in balancing the number of nodes holding different opinions in the graph.

#### 4.1.2. SIMULATIONS ON ER AND SF NETWORKS

I also perform simulations on some bigger network models, such as ER networks and SF networks with  $N = 100$  to verify whether the above findings are still valid.

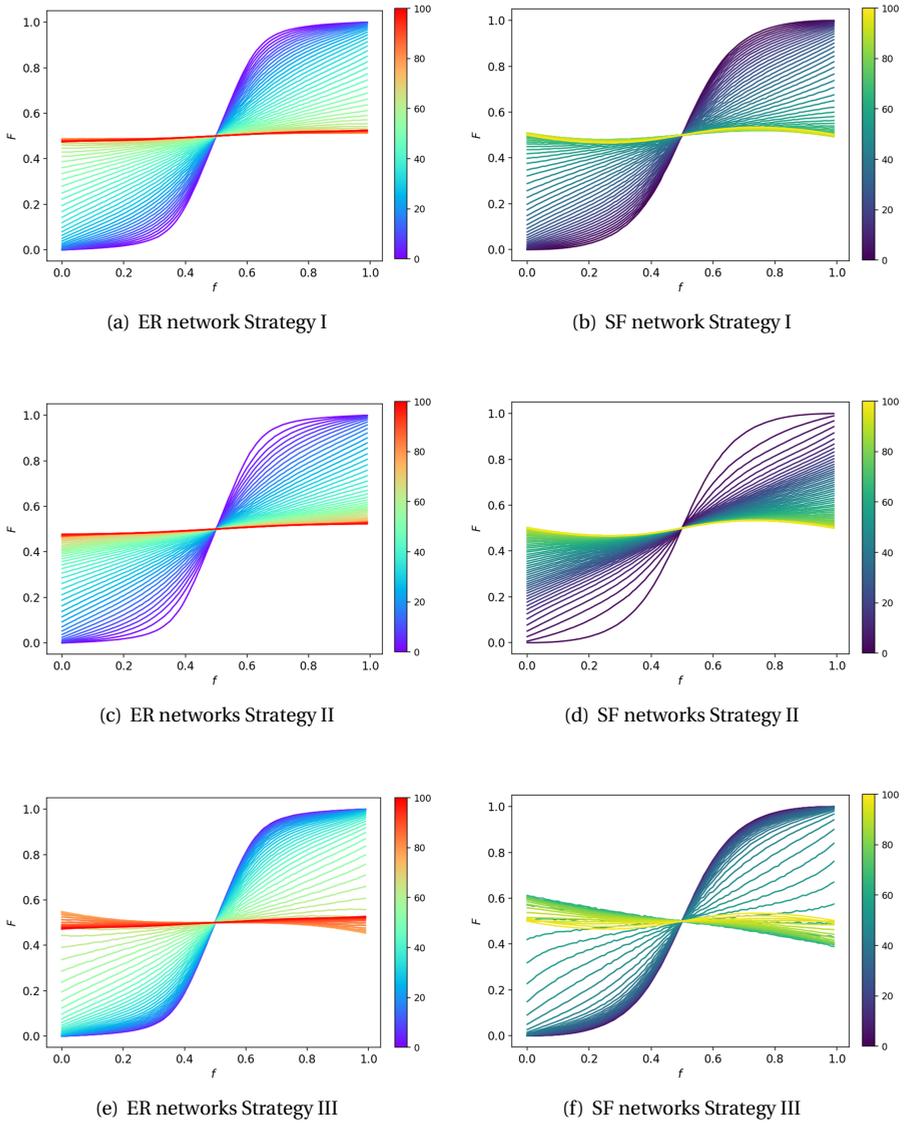


Figure 4.5: The final opinion fraction  $F$  for different initial opinion fraction  $f$  with  $N_B$  from 0 to 100, Strategy I, II, III

Fig. 4.5(a) and Fig. 4.5(b) show that, like small-scale graphs, as the number of Byzantine nodes increases, the  $F(f)$  curve of ER and SF networks becomes flattened, which indicates that the introduction of Byzantine nodes could balance the ratio of the two opinions in the network. When the number of Byzantine nodes is greater than 60, the  $F(f)$  curves for ER network becomes almost a horizontal curve, indicating the final

opinion fraction  $F$  has no relation with the initial opinion fraction  $f$ .

When a certain opinion prevails in the network, Byzantine nodes holding the majority opinion will declare the minority opinion due to the lying nature of Byzantine nodes. This affects the opinion of its neighboring nodes, shifting their opinion to the minority opinion, thus making the majority opinion less dominant. Based on the above analysis, I conjecture that for a network with a significant number of Byzantine nodes, the consensus state is unstable. Thus for the Byzantine NCO model, the coexistence steady state or the cyclic steady state is the more likely steady state. In the coexistence steady state, the number of the two opinions is closer. In the cyclic steady state, the final opinion fraction of the two opinions will also be closer since the nodes in oscillation always change to the local majority nodes. Therefore, the final opinion fraction of the Byzantine NCO model will be closer to 0.5, and the  $F(f)$  curve will be more horizontal than that of the NCO model.

Nodes with a larger degree have more neighboring nodes, which means they have a stronger influence, and it is reasonable to assume that selecting nodes with a higher degree as Byzantine nodes have a stronger effect on the final opinion fraction. In order to verify this argument, I compute the  $F-f$  curve for strategy II and strategy III and draw Fig. 4.5(c) to Fig. 4.5(f). I also find that for SF networks, adopting different strategies has a larger impact on the final opinion fraction  $F$ . The degree of the ER network follows a binomial distribution, and the differences in its degree between different nodes are small. In contrast, the degree of the SF network follows a power law distribution, so a small number of nodes in the network having a very high degree and most nodes have a low degree, which leads to a more pronounced difference in the change of the curve when different strategies are taken.

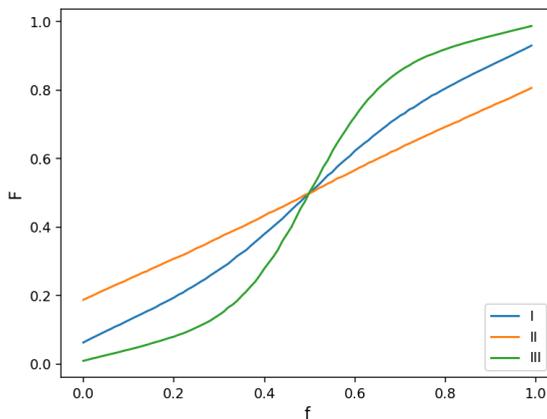


Figure 4.6: The final opinion fraction  $F$  as a function of initial opinion fractions  $f$  for ER network with  $N = 100$ ,  $p = 0.047$  and  $N_B = 30$  for three different Strategies (Strategy I: blue; Strategy II: orange; Strategy III: green)

I find that both for the ER and SF networks, the curve for strategy II is denser at the  $F(f)$  curve with  $N_B = 100$  while the curve for strategy III is denser at the  $F(f)$  curve with  $N_B = 0$ , which indicates that when strategy II is taken, the impact of Byzantine nodes is more obvious. In Fig. 4.6, I have selected the  $F(f)$  curves for  $N_B = 30$  under the three different strategies, the curve for strategy II is closer to the horizontal line  $F = 0.5$ , indicating that the balancing effect of strategy II is more significant than the balancing effect of strategies I and III.

## 4.2. CRITICAL THRESHOLD

In the NCO model, different opinions are existing in the form of clusters, where the nodes in a cluster holds the same opinion. In the consensus steady state, there is only a majority opinion cluster. In the coexistence steady state, there will be several  $\sigma_-$  and  $\sigma_+$  clusters in the network. Shao *et al.* suggest that, when the number of one opinion is above a certain critical threshold  $f_c$ , even when the opinion is still the minority, a large spanning cluster forms [9]. Once the cluster is formed, it becomes stable and cannot be penetrated by the other opinion. Below the critical threshold  $f_c$ , only the majority opinion could exist in a stable way, and the network reaches a consensus state.

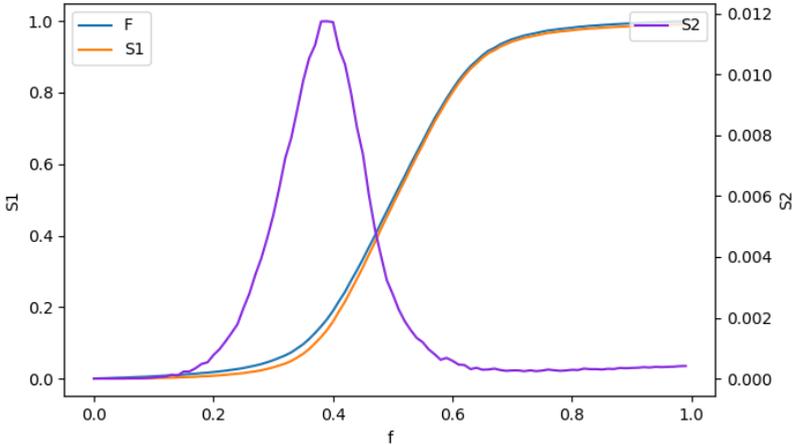


Figure 4.7: Plot of the normalized size of the largest cluster  $s_1$ , the second largest cluster  $s_2$  and the final fraction of  $\sigma_-$  nodes  $F$  for a ER network with  $N = 100$  and  $p = 0.047$

Fig. 4.7 shows  $s_1$ ,  $s_2$  and  $F$ , where  $s_1 = S_1/N$  and  $s_2 = S_2/N$  are the normalized largest and second largest  $\sigma_-$  cluster. The critical threshold  $f_c$  can be characterized by the sharp peak of  $s_2$ . For  $f < f_c$ , the size of the  $\sigma_-$  cluster follows a power law distribution:  $n \sim s^{-\tau}$ , with  $\tau = 2.5$  where  $s$  is the cluster size and  $\tau$  is the scaling exponent. Once  $f$  exceeds  $f_c$ , the size of the largest and second largest cluster diverges, where the size of the largest cluster keeps increasing but the second largest cluster  $S_2$  follows  $S_2 \sim |f - f_c|^{-\gamma}$ , where

$f_c$  is the critical threshold and  $\gamma$  is a scaling exponent [15].

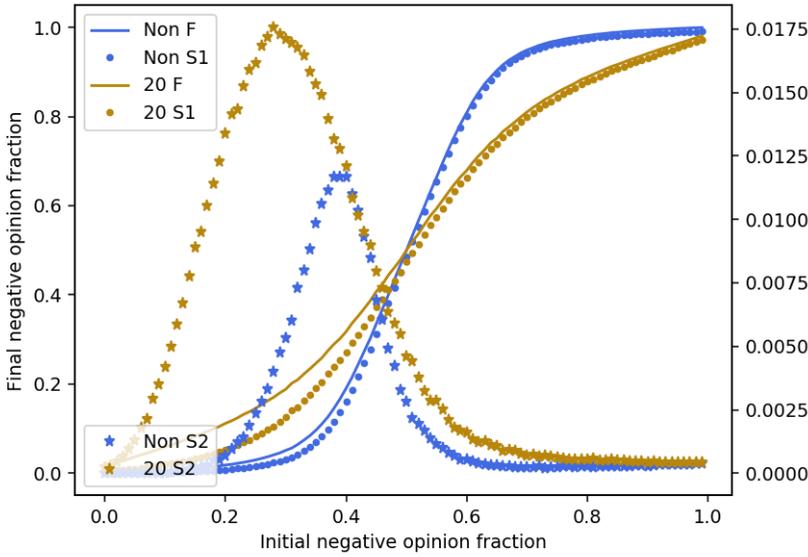


Figure 4.8: Plot of the normalized size of the largest cluster  $s_1$ , the second-largest cluster  $s_2$  and the final fraction of  $\sigma_-$  nodes  $F$  for a ER network ( $N = 100, p = 0.047$ ) with  $N_B = 0$  and  $N_B = 20$

The introduction of Byzantine nodes will impact the critical threshold. Byzantine nodes could help balance the number of nodes holding different opinions at the steady state by increasing the proportion of the minority opinion. In the Byzantine NCO model, minority opinions are more likely to form stable clusters. Fig. 4.8 shows the plots of  $F$ ,  $s_1$  and  $s_2$  for the NCO model with 20 Byzantine nodes and without Byzantine nodes. I find that the critical threshold  $f_c$  for the NCO model with Byzantine nodes is smaller than for the NCO model with no Byzantine nodes.

As the number of Byzantine nodes increases, the critical threshold  $f_c$  will keep moving to the left until it reaches 0. We denote the value of the number of Byzantine nodes  $N_B$  when the critical threshold value reaches zero as  $N_{B,c}$ . For  $N_B < N_{B,c}$ , the Byzantine NCO model still tend to reach a consensus steady state in  $f \in (0, f_c)$  and  $f \in (1 - f_c, 1)$ , where only the majority opinion could exist in a stable way. For  $N_B > N_{B,c}$ , the Byzantine NCO model tends to reach a coexistence steady state for all  $f$ . When  $N_B = N_{B,c}$ , the derivative of the  $s_2$  curve at  $f = 0$  is 0. For strategy I,  $N_{B,c}$  is 42, which means that the network will hardly reach a consensus steady state when 42 nodes are randomly selected to be Byzantine nodes.

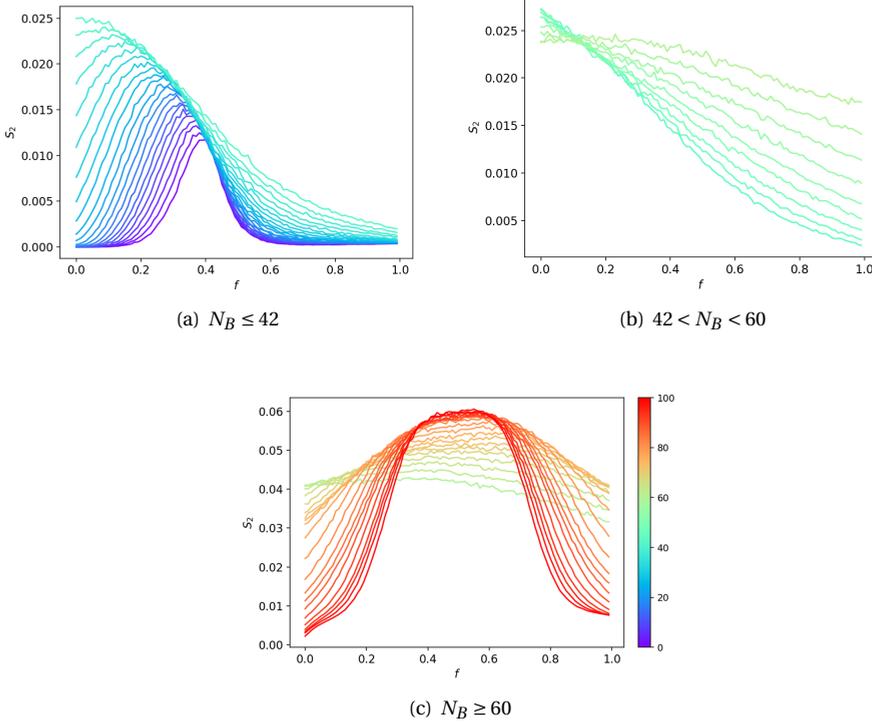


Figure 4.9: Plot of the normalized size of the largest cluster  $s_2$  for a ER network with different number of Byzantine nodes

Continuing to add Byzantine nodes, I find that the  $s_2$  curve no longer has a sharp peak in  $f \in [0, 1]$ , the largest minority cluster  $s_1$  could stably exist for all  $f$ . The network tends to reach a coexistence steady state for any initial opinion fraction. When the number of Byzantine nodes is larger than 60, the peak of  $S_2$  curve doesn't lie at  $f = 0$  again. When the number of Byzantine nodes is greater than 60, the probability of cyclic behaviour occurring around  $f = 0$  and  $f = 1$  will be significantly increased, the scattered clusters are not yet formed when the cycle occurs, and nodes holding the same opinion connect to form a large but unstable cluster. Therefore when  $N_B > 60$ ,  $s_2(f)$  is a convex function. I also find that the maximum value of  $s_1$  becomes smaller and smaller, indicating that the maximum size of the largest clusters in the network becomes smaller. The introduction of Byzantine node nodes prevents the formation of large majority opinion clusters. Fig. 4.10 shows that the maximum value of  $s_1$  becomes smaller as the number of Byzantine nodes increases, which means that large majority opinion clusters are not allowed to exist when there are a significant number of Byzantine nodes. When  $N_B > 60$ ,  $s_1(f)$  shows a peak that doesn't lie on  $f = 0$  because of the increased number of times the cyclic behavior occurs.

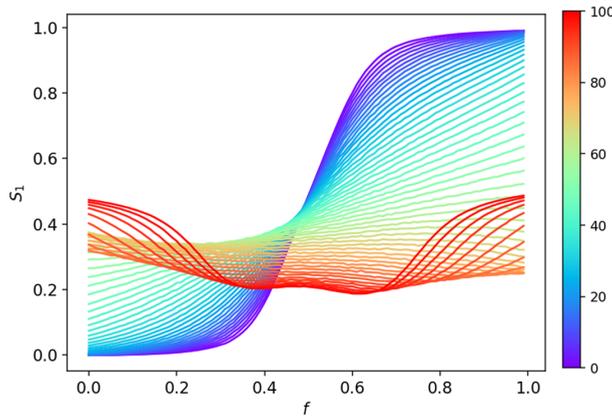


Figure 4.10: Plot of the normalized size of the largest cluster  $s_1$  for an ER network with different number of Byzantine nodes

Different Byzantine node selection strategies have different degrees of influence on the  $S_2$  curve. For strategy II and strategy III,  $N_{B,c}$  is 34 and 54, respectively, which means that the network will have difficulty reaching a consensus steady state when the 34 nodes with the highest degree in the network become Byzantine nodes, but for the nodes with the lowest degree, this value is 54. Comparing the figures in Fig. 4.11 and Fig. 4.12, I find that strategy II has a more pronounced effect on  $S_2$  curves. The  $S_2$  curves of strategy II are denser at  $N_B = 100$ , and the  $S_2$  curves of strategy II are denser at  $N_B = 0$ , which means strategy II shows a more significant influence on the  $S_2$  curve.

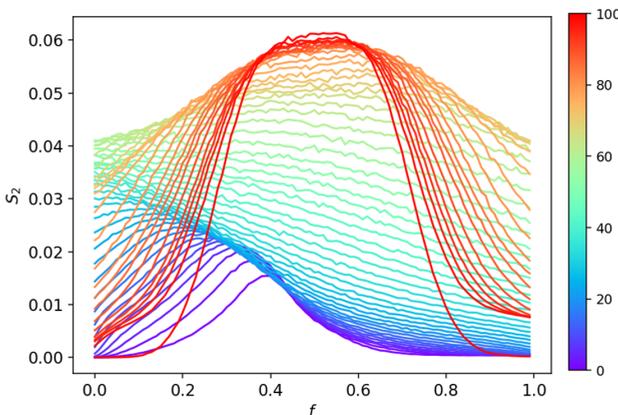


Figure 4.11: Plot of the normalized size of the second largest cluster  $s_2$  for an ER network with different number of Byzantine nodes and the placement of Byzantine nodes following Strategy II.

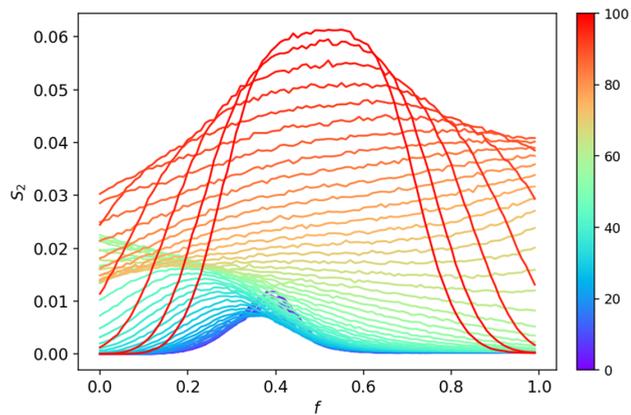


Figure 4.12: Plot of the normalized size of the second largest cluster  $s_2$  for a ER network with different number of Byzantine nodes and the placement of Byzantine nodes following Strategy III.

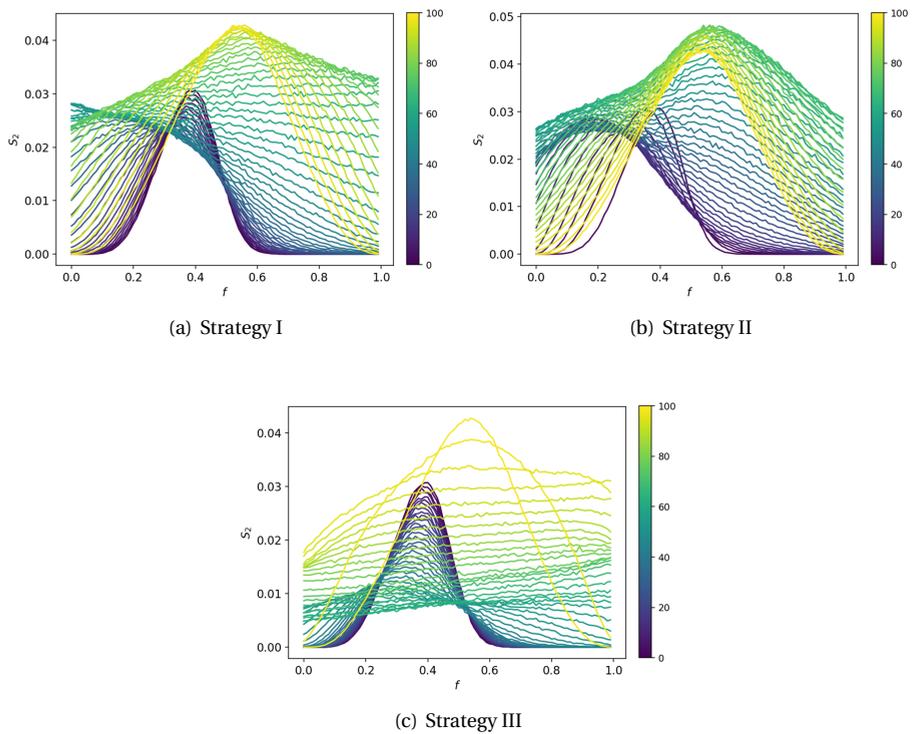


Figure 4.13: Plot of the normalized size of the largest cluster  $s_2$  for a SF network and the placement of Byzantine nodes following 3 different Strategies.

From Fig. 4.13, I find that the variation of the  $s_2$  curve is very different when different Byzantine selection strategies are adopted. For the SF network, when Strategy III is taken, the critical threshold changes very slowly. There are many nodes with a very low degree in the SF network, which only have very little impact on the system. The degree of different nodes in the SF network varies very much [30]. The large differences in the degrees of different nodes in the SF network lead to large differences in the simulation results when different strategies are selected. This further illustrates that nodes with high degree are more important than nodes with a low degree in the Byzantine NCO model.

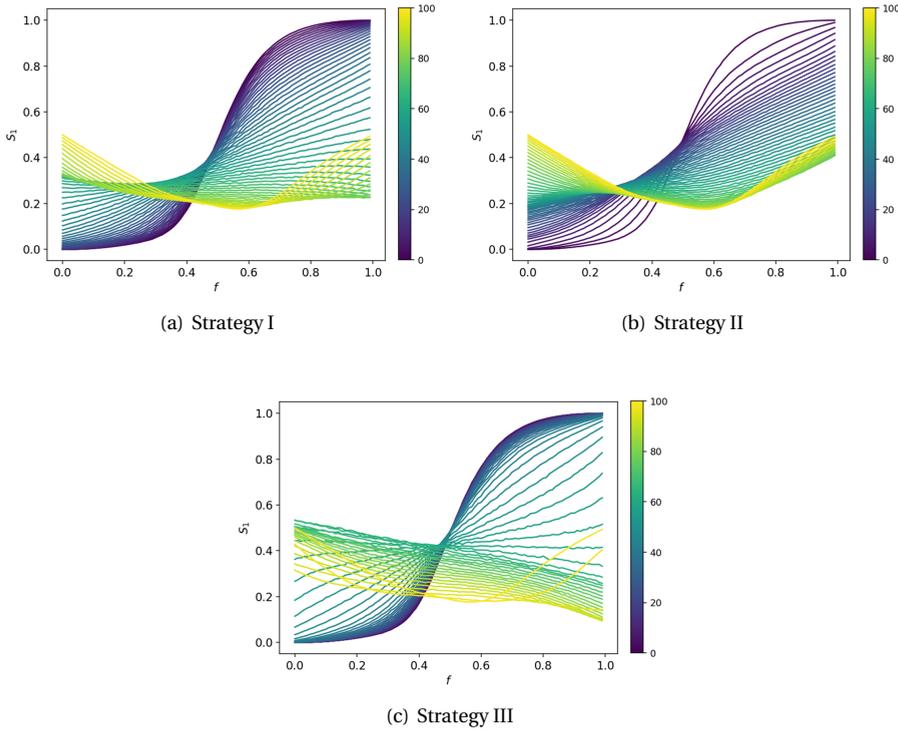


Figure 4.14: Plot of the normalized size of the largest cluster  $s_1$  for a SF network and the placement of Byzantine nodes following 3 different Strategies.

Take the nuclear war story. When there are several influential leaders (high degree nodes) in the decision making level of country A (like the SF network model), as long as these influential leaders are not bought by the enemy, the decision making level can still make the decision in favor of the country A. But if country B bribes these influential leaders, it will be a disaster for country A. Therefore it is safer for country A to build a power structure like the ER network.



# 5

## CONVERGENCE TIME OF THE BYZANTINE NCO MODEL

When the number of spies infiltrated into country A by country B is small, it will be difficult for the spies to influence decisions in country A. At this point, the spies can influence the timeliness of the decision by extending the decision time. If country A fails to make a nuclear counterattack within one hour, country B's nuclear bomb will destroy country A's nuclear counterattack capability. This relates to the convergence time problem of the Byzantine NCO model we introduce in this chapter, and spies can adopt strategies to extend the convergence time to achieve their aims.

### 5.1. CONVERGENCE TIME

Convergence time is another property of the Byzantine NCO model that we are interested in. We find that the introduction of Byzantine nodes can increase the convergence time of the network to some extent, which attackers can exploit to extend the decision time of the crowd.

#### CONVERGENCE TIME $T_c$

Convergence time is the number of iterations required for the NCO model to converge from the initial state to a steady state. The convergence time for the example graph given in Fig. 5.1 is 2, because it takes two iterations to reach a steady state from the initial state.

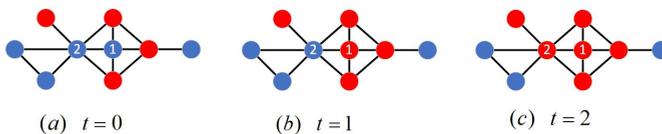


Figure 5.1: Dynamics of the NCO model on a network with  $N=9$  nodes.

A cyclic steady state is different from a fixed steady state in that cyclic steady states contains multiple states. Therefore, for the cyclic steady state, the convergence time is defined as the number of iterations used for the NCO model to converge from the initial state to the first appearance of a state in the cyclic steady state. The convergence time for the example graph in Fig. 5.2 is 0, because the initial state is part of the cycle.

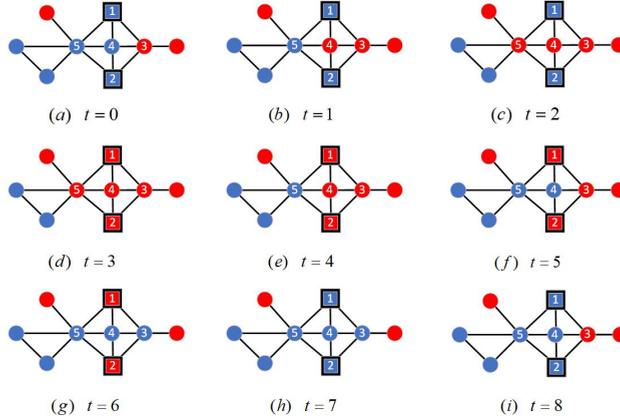


Figure 5.2: Dynamics of the Byzantine NCO model on a network with  $N = 9$  nodes.

### HAMMING DISTANCE $D$

The Hamming distance between two vectors is the number of positions at which corresponding symbols are different, denoted as

$$D(\vec{v}_t, \vec{v}_{t'}) = \sum_{i=0}^N d(i) \quad (5.1)$$

where

$$d(i) = \begin{cases} 0 & \text{if } \vec{v}_t[i] = \vec{v}_{t'}[i] \\ 1 & \text{if } \vec{v}_t[i] \neq \vec{v}_{t'}[i] \end{cases} \quad (5.2)$$

Compared with the initial opinion, Node 1, 3, 4 change their opinions at steady state. Thus, the Hamming distance between initial and final state for the example in Fig. 5.3 is

$$D(\vec{v}_{t_0}, \vec{v}_{t_3}) = 3 \quad (5.3)$$

We conjecture that the convergence time is positively related with the Hamming distance between the initial and converged states. Fig. 5.3 and Fig. 5.4 give two cases with different convergence time. The case in Fig. 5.3 has a longer  $D$ , which means much nodes in the graph change their opinions at the steady state. At each iteration, one node in the system changes its opinion. Thus it takes a longer time to make this convergence process. For the case in Fig. 5.3,  $D$  is small, and so is  $T_c$ . If the difference between the two states is large, the initial state requires multiple transformations to reach the steady state. To verify this conjecture, we measure the convergence time and compute the Hamming distance of different networks under different strategies.

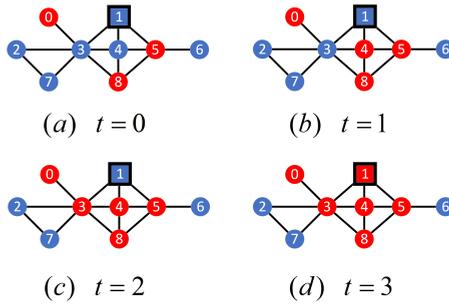


Figure 5.3: Dynamics of the Byzantine NCO model on a graph with  $N = 9$  nodes,  $T_c = 3, D = 3$ .

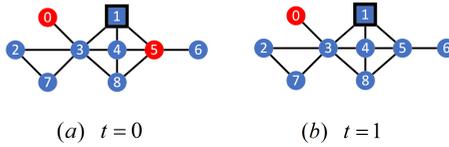


Figure 5.4: Dynamics of the Byzantine NCO model on a graph with  $N = 9$  nodes,  $T_c = 1, D = 1$ .

## 5.2. EXHAUSTIVE RESEARCH ON GRAPHS WITH $N=7, L=10$

### 5.2.1. THE CONVERGENCE TIME FOR GRAPHS WITH DIFFERENT NUMBER OF BYZANTINE NODES

I perform exhaustive research on all the 132 graphs with  $N = 7, L = 10$ , where all the possible initial states and Byzantine nodes settings are taken into consideration. Then I record the convergence time  $T_c$  and draw the plot of  $T_c$  as a function of Byzantine nodes number  $N_B$  in Fig. 5.5 to study the impact of the introduction of Byzantine nodes on the convergence time  $T_c$ .

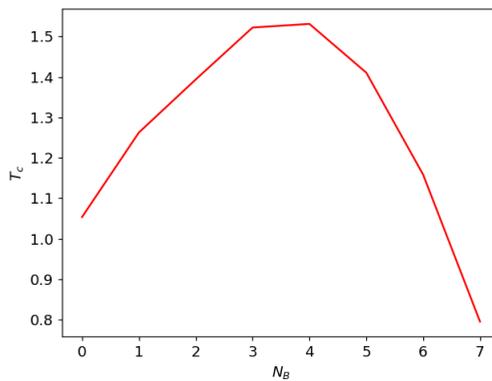


Figure 5.5: Plot of convergence time  $T_c$  as a function of Byzantine nodes number  $N_B$  for graphs with  $N = 7, L = 10$

From Fig. 5.5, we find that as the number of Byzantine nodes increases, the average convergence time first increases, reaches a peak at  $N_B = 4$ , and then decreases. To further study the convergence time of the Byzantine NCO model, I draw the plot of convergence time  $T_c$  and Hamming distance  $D$  as a function of initial opinion fraction  $f$ . To investigate the relationship between  $T_c$  and  $D$ , I compute the Pearson correlation coefficient between  $T_c$  and  $D$ , which can be calculated as follows:

$$\rho_{T_c, D} = \frac{E(T_c \cdot D) - E(T_c) \cdot E(D)}{\sqrt{E(D^2) - (E(D))^2} \cdot \sqrt{E(T_c^2) - (E(T_c))^2}} \quad (5.4)$$

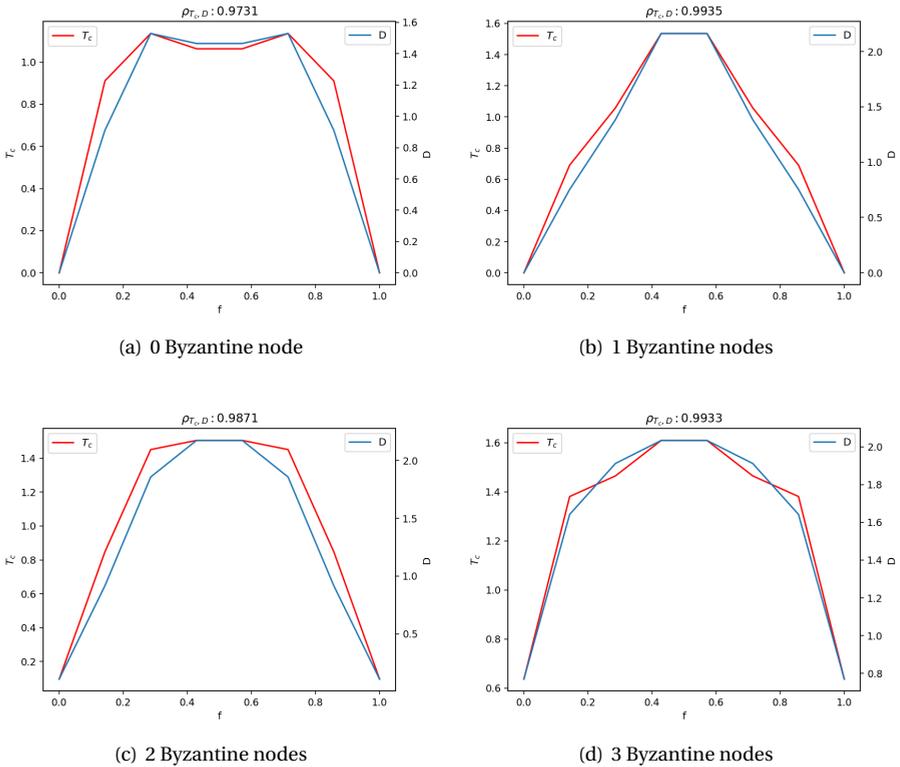


Figure 5.6: Plot of the convergence time  $T_c$  and Hamming distance  $D$  between initial and final state as a function of initial  $\sigma_-$  opinion fraction

Fig. 5.6 and Fig. 5.7 give the  $T_c(f)$  and  $D(f)$  curves when different number of Byzantine nodes are added. We find that these two curves are similar in shape. The convergence time  $T_c$  shows a high correlation with the Hamming distance  $D$  between the initial state and steady state.

In Fig. 5.6, we see that for  $N_B \in \{0,1,2,3\}$ , the  $D(f)$  curve has a shape of high in the middle and low on the sides. In Chapter 4, I showed that the network tends to reach a consensus steady state when the number of Byzantine nodes is small. Take  $0 \leq f \leq 0.5$ , when  $f$  is close to 0, the network tends to reach a consensus state, and the Hamming distance between the initial state and the steady state increases with  $f$ . When  $f$  is around 0.5, the network still tends to reach a consensus state or a coexistence state where one opinion is in the absolute majority, and the Hamming distance is large, so the convergence time is long.

As the number of Byzantine nodes increases, the tendency changes from reaching a consensus state to a coexistence state. When  $N_B = 4$ , the difference between  $D$  corresponding to different  $f$  is smaller, so the correlation coefficient between  $T_C$  and  $D$  is small. When  $N_B > 4$ , the network tends to reach a steady state with equal positive and negative opinions. Thus the middle part of the curve is low, and the sides are high.

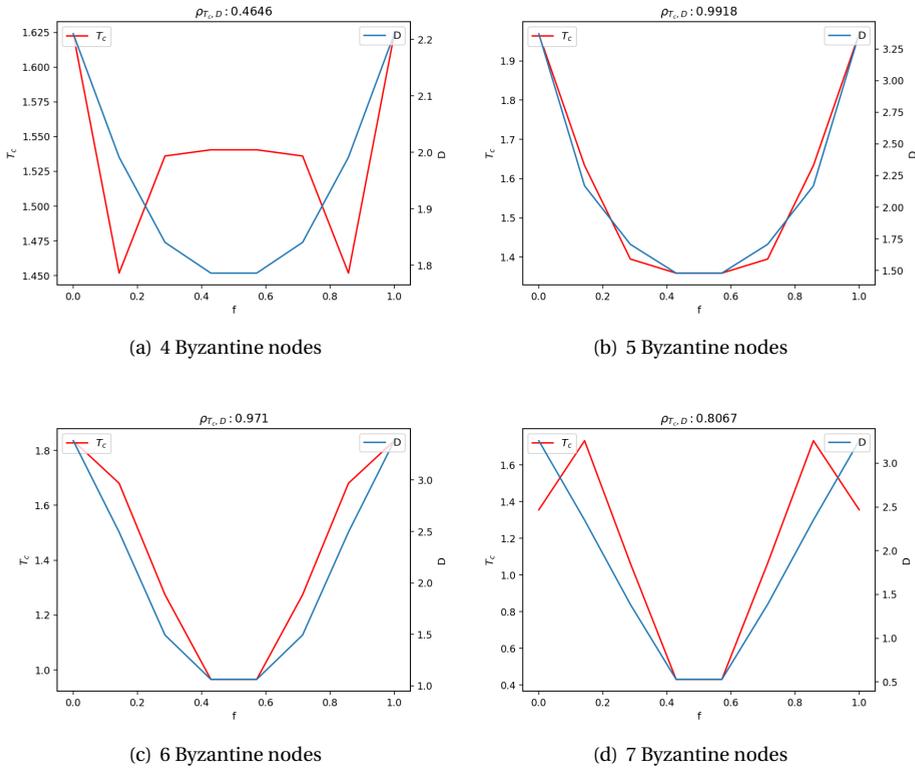


Figure 5.7: Plot of the convergence time  $T_C$  and Hamming distance  $D$  between initial and final state as a function of initial  $\sigma_-$  opinion fraction

After studying graphs with  $N = 7, L = 10$ , we found that the convergence time is highly correlated with the Hamming distance between the initial state and the steady

state. The setting of Byzantine nodes influences the steady state, which determines the Hamming distance between the initial state and the stable state. This is the mechanism by which the introduction of Byzantine nodes affects the convergence time.

### 5.3. RESEARCH ON ER AND SF NETWORKS

Extending the study to larger-scale networks, we use ER and SF networks as experimental subjects and adopted the three different Byzantine nodes selection strategies mentioned in the previous chapter for the study, and obtained the following experimental results.

From Fig. 5.8, like the small-size graphs, we find that the convergence time increases and then decreases with the increase of Byzantine nodes, and the curve peaks at  $N_B = 56$ .

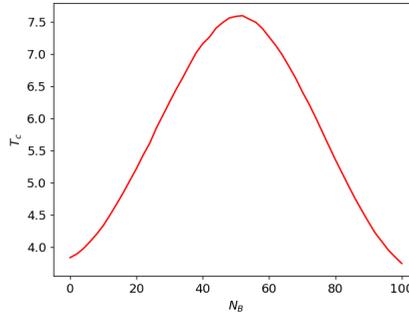


Figure 5.8: Plot of convergence time  $T_c$  as a function of Byzantine nodes number  $N_B$  for ER network and the placement of Byzantine nodes following Strategy I.

Fig. 5.9 depicts the convergence time  $T_c$  and the Hamming distance  $D$  for different numbers of Byzantine nodes. When  $N_B = 0$ , the  $T_c(f)$  and  $D(f)$  curves have an "M" shape, and the peak of the curve is near the critical threshold. This further validates our conjecture that the convergence time  $T_c$  is influenced by the Hamming distance  $D$  between initial and steady state. When  $f \in (0, f_c) \cup (1 - f_c, 1)$ , the network tends to reach a consensus steady state, thus the closer the value of  $f$  is to  $f_c$ , the greater the Hamming distance  $D$  between the initial state and the convergence state, and the longer the convergence time  $T_c$ . When  $f \in (f_c, 1 - f_c)$ , the network tends to reach a coexistence steady state. In  $f \in (f_c, 0.5]$ , the probability of the network reaching a consensus steady state decrease with the increase of  $f$ . The network is least likely to reach a consensus steady state at  $f = 0.5$ , and the average Hamming distance between the initial state and the convergence state is relatively small, so the convergence time in  $f = 0.5$  is shorter than in  $f \in (f_c, 0.5) \cup (0.5, 1 - f_c)$ .

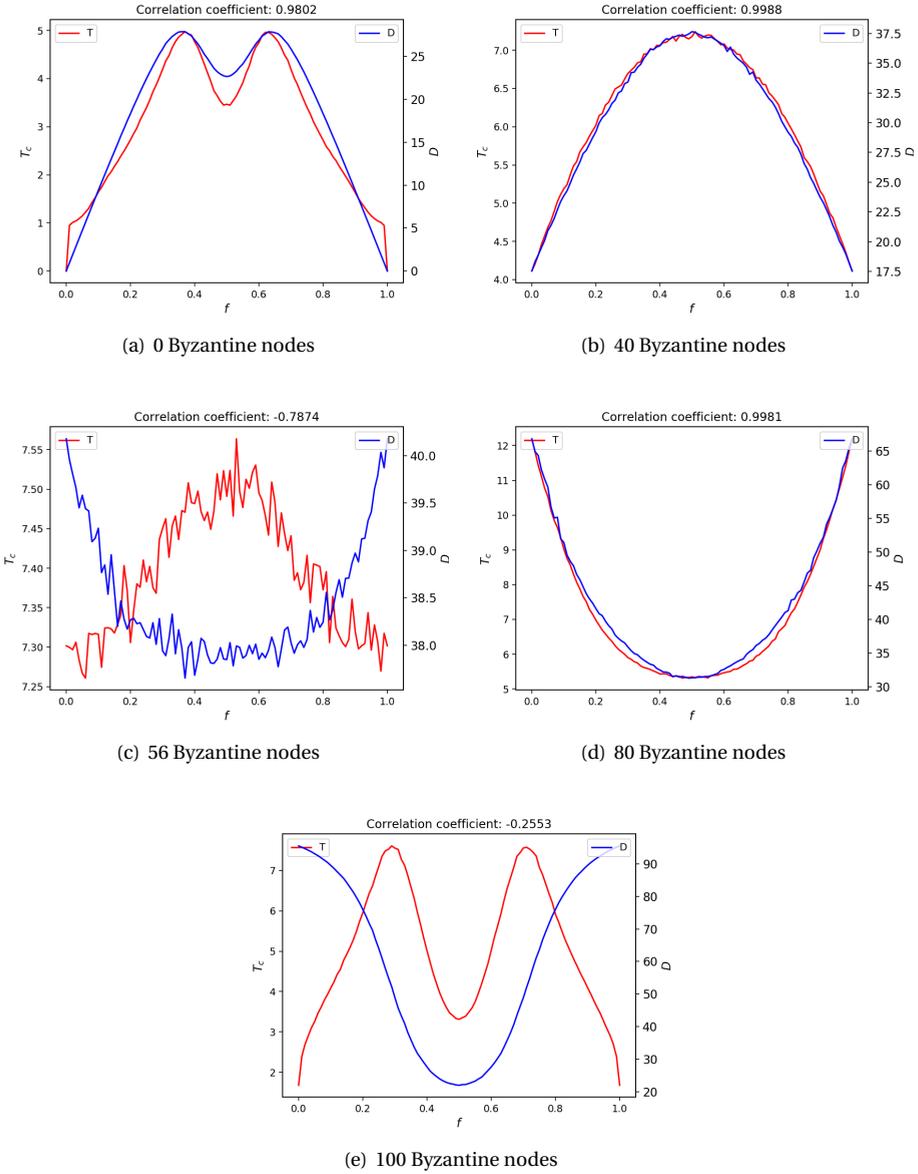


Figure 5.9: Plot of convergence time  $T_c$  as a function of initial  $\sigma_-$  opinion fraction  $f$  with different number of Byzantine nodes and the placement of Byzantine nodes following Strategy I

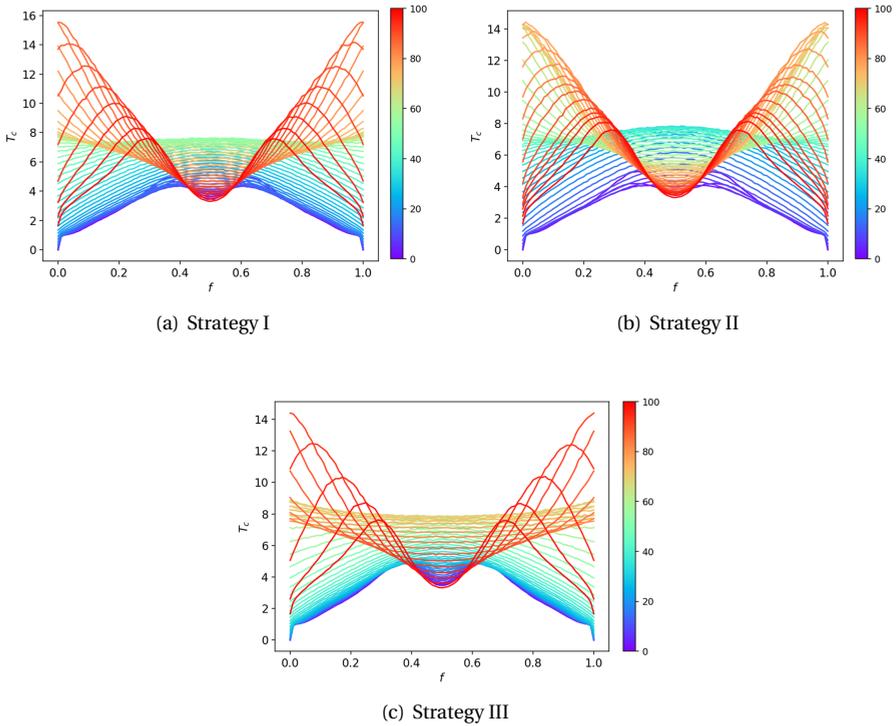


Figure 5.10: Plot of convergence time  $T_c$  as a function of initial  $\sigma_-$  opinion fraction  $f$  on ER network and the placement of Byzantine nodes following three different strategies.

With the increase of the number of Byzantine nodes, the Hamming distance between the initial state and the convergence state increases continuously, and the convergence time also increases, and the convergence time reaches the maximum when  $N_B = 56$ . At this point, the range of  $D$  and  $T_c$  is smaller, so the  $D$  curve and  $T_c$  curve are rough, and  $D$  shows a negatively correlation with  $T_c$ . From then on, like the graphs with  $N = 7$ ,  $L = 10$ , the network gradually takes the shape of high in the middle and low on the sides. When there are a large number of Byzantine nodes in the network, the  $T_c(f)$  curve has an "M" shape, which is different from the shape of the  $D(f)$  curve. This is because the probability of cyclic behavior is greatly increased, and when calculating the Hamming distance, we consider that the state of non-stationary nodes is different from the initial state, so the Hamming distance is larger.

From Fig. 5.11, when we adopt different Byzantine nodes selection strategies, the impact of Byzantine nodes on the convergence time is also different. The peak of the  $T_c(f)$  curve of strategy II corresponds to a smaller  $f$  than that of strategy I, indicating that with the same number of Byzantine nodes, the Byzantine nodes have a greater impact on the convergence time when strategy II is adopted. Byzantine nodes have less impact on convergence time when strategy III is adopted.

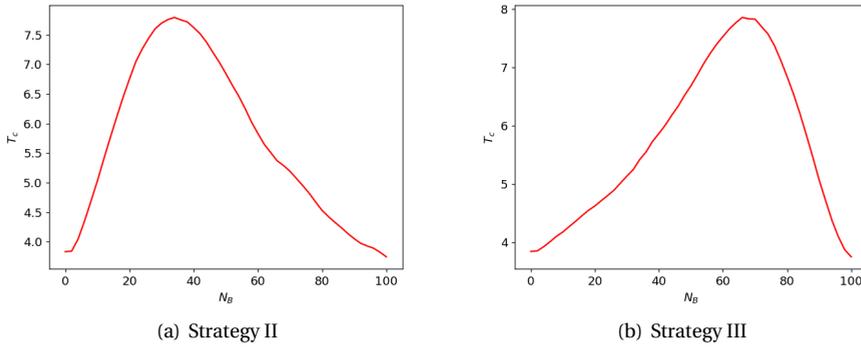


Figure 5.11: Plot of convergence time  $T_c$  as a function of Byzantine nodes number  $N_B$  for ER network and the placement of Byzantine nodes following Strategy II and III.

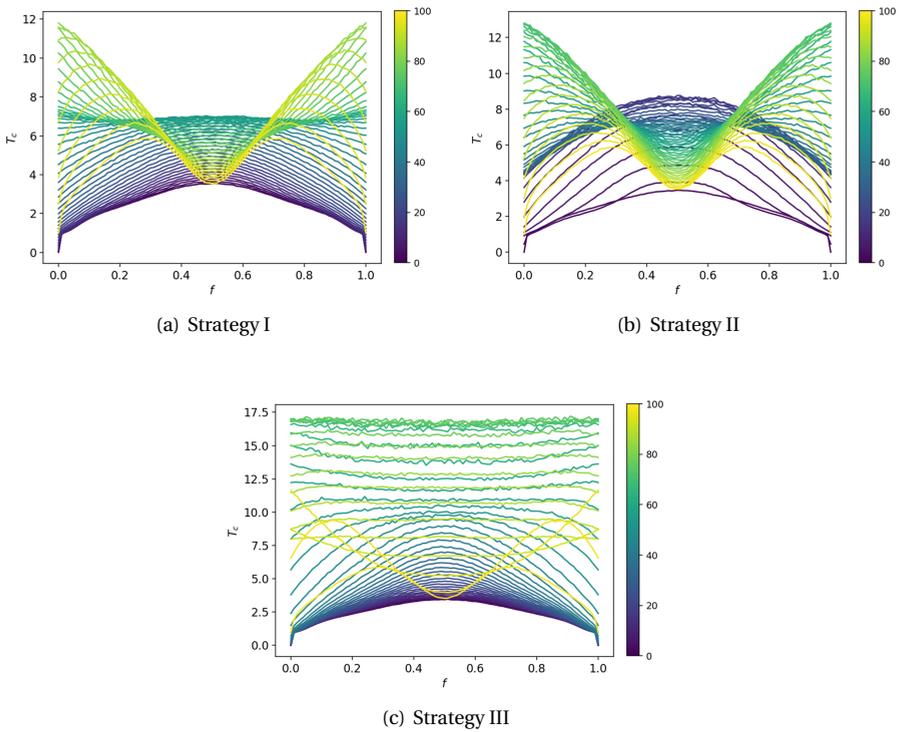


Figure 5.12: Plot of convergence time  $T_c$  as a function of initial  $\sigma_-$  opinion fraction  $f$  on SF network and the placement of Byzantine nodes following three different strategies.

Fig. 5.12 gives the  $T_c(f)$  curves of SF network with a different number of Byzantine nodes for three different strategies. It is worth noting that the convergence time of the SF network is longer compared to the ER network, especially when strategy III is adopted.

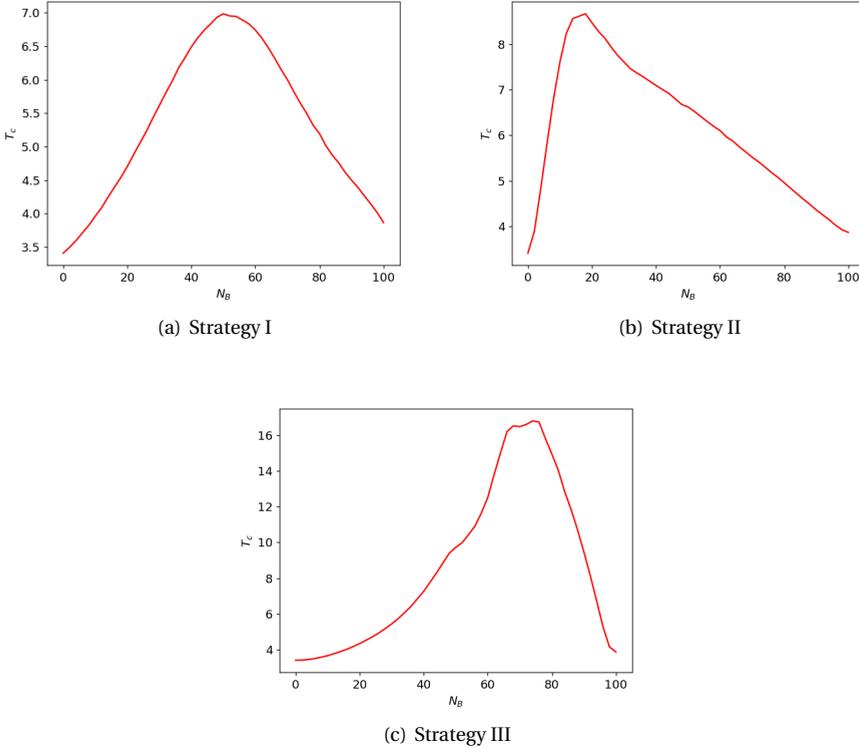


Figure 5.13: Plot of convergence time  $T_c$  as a function of Byzantine nodes number  $N_B$  for SF network and the placement of Byzantine nodes following Strategy I, II and III.

For the convergence time of SF networks, the difference caused by adopting different strategies is obvious. Introducing a small number of Byzantine nodes can significantly affect the convergence time when Strategy II is adopted. However, the maximum value of the convergence time is larger when Strategy III is adopted.

When there are only a small number of spies from country B in country A, if the spies in country B want to interfere with the decision-making time in country A, they can always declare a local minority opinion as the Byzantine nodes do.

# 6

## CONCLUSIONS AND FUTURE WORK

This chapter summarizes my findings and proposes some suggestions for future research.

### 6.1. CONCLUSIONS

This research extends the NCO model with Byzantine nodes and introduced the Byzantine NCO model. After that, the typical steady state, steady-state behavior, and Byzantine-NCO model dynamics are studied. In order to achieve this, I first designed a simulation method to simulate the dynamics of the Byzantine-NCO model. After verifying the validity of this simulation method, simulations were performed on some experimental networks using this method. Some small-scale networks and classical network models were used to study the dynamics under different initial parameter settings. The initial parameter settings include the initial opinion assignment and the Byzantine node setting, where some specific nodes are set up as the Byzantine nodes and two opinions denoted as  $\sigma^-$  and  $\sigma^+$  are randomly or intentionally assigned to all nodes in the network at a certain fraction.

In Chapter 3, I presented two typical steady states of the Byzantine NCO model: fixed steady state and cyclic steady state. In the fixed steady state, where all nodes in the network hold a local majority opinion, the state of the network stops changing. In the cyclic steady state, the network will cycle through several different states. Each state in the cyclic steady state is unfixed, where not all the nodes hold a local majority opinion. Some network patterns with long periods were found by an exhaustive study of small-scale networks. Some types of networks with long periods were given and a method was proposed to generate networks with ultra-long periods by combining different networks. I found that the introduction of Byzantine nodes affects the occurrence of cyclic behavior. When there are no or only a few Byzantine nodes in the network, the cyclic behavior occurs mainly when the initial opinion fraction  $f$  is close to 0.5. In contrast, when there are a significant number of Byzantine nodes in the network,

the cyclic behavior occurs mainly when the initial opinion fraction  $f$  is close to 0 or 1.

In Chapter 4, simulations were performed on different types of experimental networks with different initial settings and the fraction of different opinions in the steady state were measured to analyze the steady-state behavior of the Byzantine NCO model. I found that as the number of Byzantine nodes increases, the number of the two opinions in the steady state becomes closer and the correlation between the final opinion score and the initial opinion fraction becomes weaker. Shao *et al.* finds there is a critical threshold  $f_c$  in the NCO model, which is the threshold of system consensus and coexistence steady state. I found that the introduction of Byzantine nodes could change the critical threshold and make it easier for the system to reach a consensus steady state.

In Chapter 5, I measured the convergence time of the Byzantine NCO model and found a high correlation between the convergence time and the Hamming distance between the initial state and the steady state. The convergence time of different network models when different Byzantine node selection strategies are discussed. A mechanism that the Byzantine nodes influence the convergence time by influencing the steady state is revealed.

## 6

### 6.2. FUTURE RESEARCH

The Byzantine NCO model is a very worthwhile model with a lot of valuable but unexplored research in this thesis. The following are the research problems that I did not fully investigate during our research due to time constraints.

#### 1. Long-cycle Graphs

In Chapter 3 I introduced some graphs with long cycle periods and gave a method to generate networks with an extremely long cyclic period. However, this method can only generate networks based on the combination of the patterns I have found, and therefore cannot generate networks of arbitrary length. It would be very interesting research to construct a general method that can efficiently generate networks of an arbitrary cycle period.

#### 2. Taking more Byzantine nodes selecting strategies

In this thesis, I only tried three strategies according to the degree of the node. There are many other measures of the importance of a node in the network, such as betweenness, closeness and eigenvector centrality. Selecting Byzantine nodes based on these properties can help us better understand the importance of different nodes in the Byzantine NCO model.

#### 3. Taking more network models

In this thesis, only three network models were discussed. Further research can be performed on more network models, such as spatial modular networks and small-world networks.

#### 4. Extending the study of Byzantine NCO models to application areas

In this thesis, we discussed the dynamics of the NCO model on some toy networks but did not put it into practice. Researching the dynamics of the NCO model on the real-world network would be very exciting.



# BIBLIOGRAPHY

- [1] S. N. Durlauf and H. P. Young, *Social dynamics*. MIT Press, 2004, vol. 4.
- [2] G. Aletti, A. K. Naimzada, and G. Naldi, “Mathematics and physics applications in sociodynamics simulation: The case of opinion formation and diffusion,” in *Mathematical modeling of collective behavior in socio-economic and life sciences*, Springer, 2010, pp. 203–221.
- [3] A. Sîrbu, V. Loreto, V. D. P. Servedio, and F. Tria, “Opinion dynamics: Models, extensions and external effects,” *Participatory Sensing, Opinions and Collective Awareness*, pp. 363–401, May 2016, ISSN: 1860-0840. DOI: [10.1007/978-3-319-25658-0\\_17](https://doi.org/10.1007/978-3-319-25658-0_17).
- [4] R. Pastor-Satorras and A. Vespignani, *Evolution and Structure of the Internet: A Statistical Physics Approach*. Cambridge University Press, 2004. DOI: [10.1017/CB09780511610905](https://doi.org/10.1017/CB09780511610905).
- [5] M. E. J. Newman, *Networks: an introduction*. Oxford; New York: Oxford University Press, 2010, ISBN: 9780199206650 0199206651. [Online]. Available: [http://www.amazon.com/Networks-An-Introduction-Mark-Newman/dp/0199206651/ref=sr\\_1\\_5?ie=UTF8&qid=1352896678&sr=8-5&keywords=complex+networks](http://www.amazon.com/Networks-An-Introduction-Mark-Newman/dp/0199206651/ref=sr_1_5?ie=UTF8&qid=1352896678&sr=8-5&keywords=complex+networks).
- [6] R. Cohen and S. Havlin, *Complex networks: structure, robustness and function*. Cambridge University press, 2010.
- [7] D. J. Watts, “The “new” science of networks,” *Annual Review of Sociology*, vol. 30, no. 1, pp. 243–270, 2004. DOI: [10.1146/annurev.soc.30.020404.104342](https://doi.org/10.1146/annurev.soc.30.020404.104342).
- [8] B. Bollobás and O. Riordan, “Random graphs and branching processes,” in *Handbook of large-scale random networks*, Springer, 2008, pp. 15–115.
- [9] J. Shao, S. Havlin, and H. E. Stanley, “Dynamic opinion model and invasion percolation,” *Physical Review Letters*, vol. 103, no. 1, p. 018 701, 2009.
- [10] K. Sznajd-Weron and J. Sznajd, “Opinion evolution in closed community,” *International Journal of Modern Physics C*, vol. 11, no. 06, pp. 1157–1165, 2000.
- [11] T. M. Liggett *et al.*, *Stochastic interacting systems: contact, voter and exclusion processes*. Springer Science & Business Media, 1999, vol. 324.
- [12] S. Galam, “Minority opinion spreading in random geometry,” *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 25, no. 4, pp. 403–406, 2002.
- [13] P. L. Krapivsky and S. Redner, “Dynamics of majority rule in two-state interacting spin systems,” *Physical Review Letters*, vol. 90, no. 23, p. 238 701, 2003.

- [14] Q. Li, L. A. Braunstein, H. Wang, J. Shao, H. E. Stanley, and S. Havlin, "Non-consensus opinion models on complex networks," *Journal of Statistical Physics*, vol. 151, no. 1, pp. 92–112, 2013.
- [15] D. Stauffer and A. Aharony, *Introduction to percolation theory*. CRC press, 2018.
- [16] Q. Li, L. A. Braunstein, S. Havlin, and H. E. Stanley, "Strategy of competition between two groups based on an inflexible contrarian opinion model," *Physical Review E*, vol. 84, no. 6, p. 066 101, 2011.
- [17] P. Van Mieghem, *Performance analysis of complex networks and systems*. Cambridge University Press, 2014.
- [18] D. B. West *et al.*, *Introduction to graph theory*. Prentice Hall Upper Saddle River, 2001, vol. 2.
- [19] D. A. Bader, S. Kintali, K. Madduri, and M. Mihail, "Approximating betweenness centrality," in *International Workshop on Algorithms and Models for the Web-Graph*, Springer, 2007, pp. 124–137.
- [20] P. Bonacich, "Some unique properties of eigenvector centrality," *Social Networks*, vol. 29, no. 4, pp. 555–564, 2007.
- [21] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.
- [22] A.-L. Barabási and E. Bonabeau, "Scale-free networks," *Scientific American*, vol. 288, no. 5, pp. 60–69, 2003.
- [23] P. Erdős, A. Rényi, *et al.*, "On the evolution of random graphs," *Publ. Math. Inst. Hung. Acad. Sci.*, vol. 5, no. 1, pp. 17–60, 1960.
- [24] A.-L. Barabási, "Scale-free networks: A decade and beyond," *Science*, vol. 325, no. 5939, pp. 412–413, 2009.
- [25] N. Schwartz, R. Cohen, D. Ben-Avraham, A.-L. Barabási, and S. Havlin, "Percolation in directed scale-free networks," *Physical Review E*, vol. 66, no. 1, p. 015 104, 2002.
- [26] B. D. McKay and A. Piperno, "Practical graph isomorphism, ii," *Journal of Symbolic Computation*, vol. 60, pp. 94–112, 2014, ISSN: 0747-7171. DOI: <https://doi.org/10.1016/j.jsc.2013.09.003>.
- [27] D. G. Wells, *Prime numbers*. Wiley Hoboken, 2005.
- [28] S. Galam, "Contrarian deterministic effects on opinion dynamics: "the hung elections scenario", " *Physica A: Statistical Mechanics and its Applications*, vol. 333, pp. 453–460, 2004.
- [29] C. Borghesi and S. Galam, "Chaotic, staggered, and polarized dynamics in opinion forming: The contrarian effect," *Physical Review E*, vol. 73, no. 6, p. 066 118, 2006.
- [30] E. López, S. V. Buldyrev, S. Havlin, and H. E. Stanley, "Anomalous transport in scale-free networks," *Physical Review Letters*, vol. 94, no. 24, p. 248 701, 2005.
- [31] B. Preneel, "Cryptographic hash functions," *European Transactions on Telecommunications*, vol. 5, no. 4, pp. 431–448, 1994.

# A

## APPENDIX A

This appendix shows the optimized algorithm mentioned in Chapter 3 in detail. For networks with a short convergence time, we can directly use the list type to store the state vectors of the network. When the convergence time is longer, we can reduce the memory consumption by converting the state vectors to binary numbers. For example, for a network with  $N=378$ , the memory required to store a state vector directly as a list is 3304 bytes, and the memory needed after conversion to binary numbers is 76 bytes. This method has been very good at saving memory, but it still does not satisfy the need for memory saving when the convergence time of the network is extremely long, so we need more efficient memory saving methods.

We can compress the memory by encoding the state vector. During the simulation of the Byzantine NCO model, the simulation stops as soon as we find two identical state vectors in the dynamics, so that not all possible states appear in the dynamics, which means that we only need to encode the states that occur in the dynamics. The code length is:

$$L(C(x)) = \lceil \log_2 N_s \rceil \quad (\text{A.1})$$

where  $N_s$  is the number of states appearing in the dynamics, equal to the cycle length plus the convergence time.

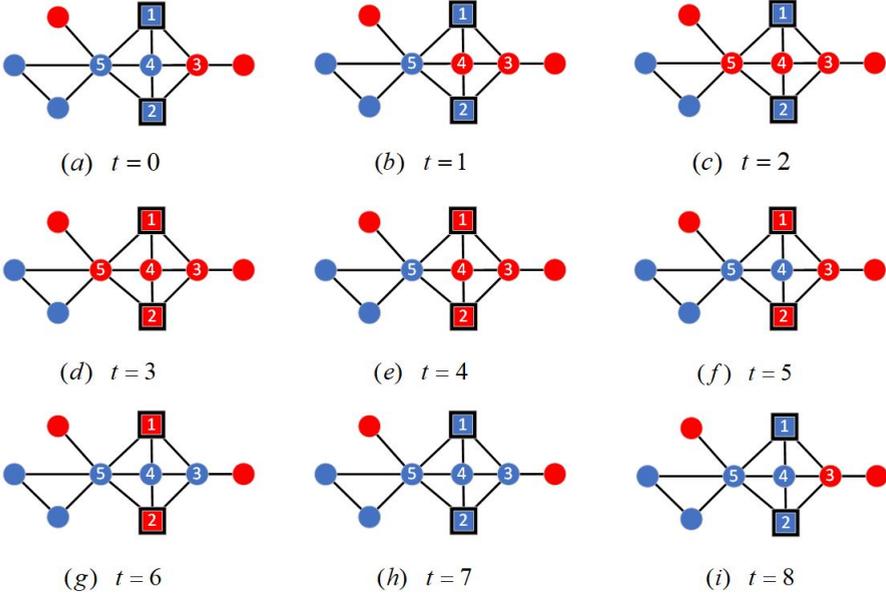


Figure A.1: A demonstration of the cyclic steady states in the Byzantine NCO model for a network with 9 nodes. The node's color denotes the opinion of the node, red means the node holds a positive opinion  $\sigma^+$ , blue means the node has a negative opinion  $\sigma^-$ . At  $t=0$  to  $t=6$ , the normal nodes 3,4,5 change their opinion according to their local majority opinion. At  $t=7$ , the Byzantine nodes 1, 2 change their opinion, then node 3's local majority opinion changes from negative to positive due to the lying nature of Byzantine nodes. At  $t=8$ , the graph shows the same state as initial state. The graph will repeat the previous dynamics.

For example, for the network in Fig. A.1, a total of eight states appear in its dynamics, so based on the previous analysis, we don't need to store a state vector as a list:

$$\vec{v}_0 : [-1 \quad -1 \quad 1 \quad 1 \quad -1 \quad -1 \quad 1 \quad -1 \quad -1] \quad (\text{A.2})$$

or a binary number of length 9:

$$v_0 : 001100100 \quad (\text{A.3})$$

but a binary number of length 3 ( $\text{Log}_2 8 = 3$ ) as:

$$v_0 : 000 \quad (\text{A.4})$$

This approach is more efficient but has the drawback: it requires us to predetermine the value of  $N_s$  and leave a certain amount of redundancy in the encoding space. However, this approach does not always work, especially when the value of  $N_s$  is large. To solve this problem, I came up with a way to compress data using hash functions.

A hash function is any function that is used to map data  $x$  of arbitrary size to fixed-size values  $y$ .

$$H(x) = y \quad (\text{A.5})$$

The hash function in Python could encrypt arbitrary data to a 36-byte string. For two different state vectors  $\vec{v}_1$  and  $\vec{v}_2$ , the hash value of these two data  $y_1$  and  $y_2$  should be different. For example, for a network with  $N = 378$ , there are

$$\begin{aligned}\vec{v}_1 : [-1 \quad -1 \quad -1 \quad \cdots \quad 1 \quad 1 \quad 1] &\rightarrow y_1 = H(\vec{v}_1) = -4898185504147061392 \\ \vec{v}_2 : [-1 \quad 1 \quad -1 \quad \cdots \quad 1 \quad -1 \quad 1] &\rightarrow y_2 = H(\vec{v}_2) = 6274575924978181886\end{aligned}\quad (\text{A.6})$$

One problem with using hash functions to compress data is that when the number of encrypted messages is close to or larger than the domain of the hash function, a collision occurs, where for two different data  $s_1$  and  $s_2$ , the corresponding hash values  $y_1$  and  $y_2$  are the same [31]. For instance, if we have  $N_s$  larger than  $2^{288}$ , the domain of hash values will be used up, there must be a collision. The solution to this problem is to check one more step when determining if the network has converged. We originally check whether

$$\vec{v}_{B,t_{i+L}} = \vec{v}_{B,t_i} \quad (\text{A.7})$$

holds to determine whether the network has converged, but now to check whether

$$\vec{v}_{B,t_{i+L}} = \vec{v}_{B,t_i} \text{ and } \vec{v}_{B,t_{i+L+1}} = \vec{v}_{B,t_{i+1}} \quad (\text{A.8})$$

which is equivalent to extending the domain of the hash without taking up additional memory.