

Saint Petersburg State University

Published in manuscript form

Chi Zhao

Modeling of binary opinion dynamics in social networks of complex configurations

Scientific specialty 2.3.1.

Systems analysis, management and processing of information, statistics

DISSERTATION

Thesis for a degree candidate of
technical sciences

Scientific advisor:
Doctor of physical and mathematical sciences,
Prof. E. M. Parilina

St. Petersburg
2024

Contents

Introduction	4
Chapter 1. Opinion Dynamics in Two-Layer Networks with Hypocrisy	21
1.1 Basic and Concealed Voter Models	21
1.1.1 Basic Voter Model (BVM)	22
1.1.2 Concealed Voter Model (CVM)	24
1.2 General Concealed Voter Model	28
1.2.1 Motivation	28
1.2.2 Symmetric case: Complete internal network	29
1.2.3 Asymmetric cases: Incomplete internal network	31
1.2.4 Asymmetric case: Incomplete external network	37
1.3 Experiments and Results	41
1.4 Conclusion to Chapter 1	44
Chapter 2. Analysis of consensus time and winning rate in two-layer networks with hypocrisy of different structures	46
2.1 Multi-layer network with replica nodes	46
2.2 Two-layer network with replica nodes	46
2.3 Model	47
2.3.1 The general concealed voter model (macro version)	47
2.3.2 The general concealed voter model (micro version)	48
2.4 Experiments and results	51
2.4.1 General description	51
2.4.2 Main results and observations	52
2.5 Conclusion to Chapter 2	71
Chapter 3. Centrality measures and opinion dynamics in two-layer networks with replica nodes	73
3.1 Two-layer network simplification	73
3.2 Zachary’s karate club network in two-layer setting	75
3.3 Centrality measures in one- and two-layer networks	76
3.3.1 Classical centrality measures	77
3.3.2 Random walk based centralities	79

3.3.3	Game-theoretic centrality measures	81
3.4	Experiments	87
3.4.1	Centralities based on the Shapley and Myerson values	87
3.4.2	Experiments to examine correlation of network properties and opinion dynamics	95
3.5	Conclusion to Chapter 3	100
Chapter 4. ShapG: new feature importance method based on the Shapley value		102
4.1	Explainable Artificial Intelligence (XAI) methods	102
4.2	ShapG: a novel XAI method	104
4.2.1	The Shapley value	104
4.2.2	ShapG (explanations based on the Shapley value for graphs) .	105
4.3	Experiments	111
4.3.1	Description of datasets	111
4.3.2	Preprocessing data for ShapG	112
4.3.3	AI prediction models	114
4.3.4	Evaluation of XAI methods	117
4.4	Results and analysis	117
4.4.1	Feature importance calculated by ShapG	117
4.4.2	Evaluation of XAI methods	120
4.4.3	Explanation of complex models	127
4.5	Conclusion to Chapter 4	129
Conclusions		131
List of acronyms and symbols		133
List of Figures		135
List of Tables		137
References		138
Appendix		149

Introduction

Relevance of thesis topic

Opinion dynamics is a critical area of study that focuses on forming beliefs, attitudes, and viewpoints, their spreading and evolving within social networks. Social networks, comprising individuals or entities linked through various relationships, can be considered as primary channels for information exchange and opinion sharing nowadays. Since our world becomes increasingly interconnected, understanding of forming opinions, their spreading, and evolving within these complex networks is crucial for addressing numerous real-world challenges, i.e. for combating misinformation, designing efficient public policies, etc. The study of opinion dynamics combines insights from sociology, psychology, physics, and computer science to model and analyze individual interactions leading to collective behavior and social phenomena.

Social networks are inherently complex systems, often consisting of multiple interconnected layers that represent different types of relationships or communication channels between individuals. In today's digital age, when online platforms and social media have dramatically altered the landscape of human interaction and information dissemination they also have introduced new complexities into social networks, such as the potential for rapid spread of information, the formation of echo chambers, and the diversity of network structure. Moreover, the existence of individuals who may publicly express opinions that are different from their private opinions adds another layer of complexity to these systems. This phenomenon is prevalent in various real-world scenarios, such as workplace environment where employees might conceal their true opinions to conform to organizational culture, or in political settings where individuals may alter their public position to align with common majority opinions. Understanding the impact of such behavior on overall opinion dynamics is crucial for developing more accurate models of social influence and for designing interventions to promote authentic discourse.

When people talk about spreading opinions, they often refer to the process of reaching a consensus or a common agreement on a particular issue. It is interesting to examine the dynamics of opinion formation and propagation in social networks with complex configurations, where individuals may exhibit stubborn be-

havior, hypocrisy, fickleness, etc., and the network structure may be multi-layered, cyclic, star, or of other types. Different network structures cause different network characteristics and heterogeneity between individuals. For instance, the network structure can significantly change the centrality of individuals, the average shortest path, and the path of information dissemination affecting the opinion propagation. And different individual's behavior can also affect the opinion propagation, such as the stubborn never changes his opinion, the hypocritical individuals who may hold different private and public opinions, the fickle individuals who may change their opinions frequently, etc. The existence of these individuals introduces new challenges to the consensus formation and opinion propagation. The analysis of consensus time and winning rates for different network structures and initial individual's configuration provides insights into the effect of the network topology and individual's behavior on speed and outcome of opinion propagation. This knowledge can be applied for designing more efficient communication strategies and understanding the resilience of different social structures to opinion change.

The thesis is devoted to studying opinion dynamics in social networks with complex configurations in terms of a network structure and individual's behavior. A key aspect of this research is the exploration of two-layer network models, which distinguish public (external) and private (internal) communication layers. This approach allows for a more nuanced and realistic representation of real-world social interactions. At the same time, we also aim to explore centrality measures in these complex network structures, and this analysis offers valuable insights into identifying influential individuals within a social system. This has practical applications in the areas such as targeted marketing, political campaigning, or public health interventions, where identifying key influencers is crucial for effective message dissemination. Furthermore, in this thesis, we broaden the application of the developed centrality measures to explainable artificial intelligence. Usage of the developed centrality measures to estimate feature importance in regression and classification tasks is helpful for artificial intelligence researchers to find the optimal model training strategy. From a social perspective, the optimal training strategy can reduce energy consumption, thereby decreasing carbon dioxide emissions and contributing to carbon neutrality.

Overview of the results in this area

Wiener's pioneering work in cybernetics [1] laid the foundation for socio-cybernetics [2]. Socio-cybernetics focuses on the inherent laws of self-organization and self-adaptation of social systems, exploring how social mechanisms and social structures of a system representing society can spontaneously accomplish specific coordination and control behaviors. The combination of sociology, systems and control theory has shifted the focus of social network research from social network analysis to the study of the evolution of perceptions, behaviors, and social relationships in social networks from dynamic systems perspective, giving a rise to a new research area — Opinion Dynamics.

In the 21st century, the development of multi-agent systems and complex networks has provided researchers with a wealth of mathematical models and tools for quantitative analysis or numerical simulation of large-scale social networks.

Opinion dynamics models can be divided into two main groups: macroscopic and microscopic. Macroscopic models examine social networks using statistical-physical methods and applying probability and statistics theories to analyze how the distribution of opinions evolves, e.g., the Ising model [3] and voter model [4]. The Ising model has a long history in statistical physics [5]. The Sznajd model [6] is one of the well-known modifications of the Ising model. In each round of the Sznajd model, a pair of agents a_i and a_{i+1} is selected to influence the nearest neighbors, i.e. agents a_{i-1} and a_{i+2} . In a voter model [4], a random agent a_i is chosen, then his random neighbor is chosen, and this neighbor adopts a_i 's opinion.

Microscopic models directly describe how individuals' opinions evolve from social individuals' perspectives, e.g., see the DeGroot model [7], the Friedkin-Johnsen (F-J) model [8], and bounded confidence models [9, 10].

In the DeGroot model, each individual updates his opinion based on his own and neighbors' opinions. The F-J model is one of the major extension of the DeGroot model, and in the F-J model, the presence of stubborn-agents extends the DeGroot model. In the F-J model, actors can also factor their initial prejudices into every iteration of opinion [11]. The possibility to control the agents' opinions by nonmembers of network is considered in [12, 13]. The upgraded F-J model with passive and active agents is introduced in [14]. A bounded confidence model (BCM) is a model, in which agents ignore the opinions that are very far from their own ones [5]. The BCM includes two essential models: the Deffuant-Weisbuch model (D-W) proposed in paper [10], and the Hegselman-Krause (H-K) model introduced

in the work [9]. In the D-W model, two individuals a_i, a_j are randomly chosen, and they determine whether to interact according to the bounded confidence [15]. The H-K model is also an extension of the DeGroot model, in which it is assumed that every individual in the network has a confidence bound, and the individual's opinion can be affected only by others within their own confidence bound. Therefore, the opinion updating rule of the H-K model is state-related. The Krasnoshchekov's model (K) of opinion dynamics in the society represented by one layer is introduced in [16] and then it is examined in [17]. This opinion dynamics can be reduced to the F-J dynamics. In fact, if the opinion updating rule presented in K model is applied, the corresponding dynamics gives the convergence of the agents' opinions to some terminal opinions. Therefore, a consensus is reached. The comprehensive survey [18] examines various models in the bounded confidence opinion dynamics domain, highlighting key mechanisms leading to consensus emergence, polarization, and fragmentation within groups.

The presence of a group of agents who can manipulate the opinions of the society is examined in [19]. The agents are assumed to be heterogeneous in this model, taking into account that the group of the so-called leaders knows the initial opinions of all the agents, while the rest of the agents do not know this information.

According to [20], opinion dynamics models are usually composed of a few essential elements: (i) opinion expression formats defining how to represent opinion mathematically, (ii) fusion rule determining how individuals interact with each other, and (iii) opinion dynamics environments, that is, the structure of such a social network.

In a social network, individuals neither fully accept nor completely ignore the opinions of other individuals. To a certain extent, they consider these opinions in forming their new opinions in a process defined by a fusion rule. Through a group interaction, individuals continuously update and integrate their opinions on the same issue. Eventually, there are three varieties of stabilized fusion results: consensus, polarization, fragmentation, and one unstable fusion result, that is oscillation [9].

The basic voter model proposed by Richard and Thomas is called BVM [4] and the concealed voter model proposed by Gastner et al. as CVM is designed [21, 22]. Both BVM and CVM belong to the macroscopic model. The structure of BVM is based on a complete network, and the general assumption of this model is that individuals always express their opinions publicly. Therefore, the fusion rule of BVM is quite intuitive, that is, selecting an individual and his neighbor randomly, the individual adopts his neighbor's opinion. CVM assumes that the social network

is divided into external and internal layers, and the individuals feel free to conceal or publicly express their opinions. The external layer of CVM is a complete network, and each node in the external layer is linked with a node in the internal layer. Moreover, there are no connections between nodes in the internal layer. Therefore, internal interaction is not allowed in CVM.

In this thesis, we assume that the individuals can interact in the internal layer, which is an assumption making our model different from CVM. We propose this idea motivated by the fact that the individuals always share their real opinions with their close friends. The multilevel approach to model the structure of the society including team networks in companies is widely spread in the models of industrial organization [23]. Therefore, we also incorporate this multilevel communication approach in the model of opinion dynamics, and call CVM with possible interactions in the internal layer as GCVM [24–27]. In Chapter 1, we start with a network structure and use statistical-physical methods and probability theory to formulate and simulate the opinion dynamics process (i.e. in the simulations, we do not create a real network and simulate this model based on formulas), and we call the proposed model macroscopic-GCVM [24]. In Chapter 2, we create the corresponding networks for the given internal and external structures instead of only use statistical-physical methods, the corresponding model is referred to microscopic-GCVM [25–27].

Since network structure in GCVM is two-layer, it is interesting to examine how this structure in general, and also network characteristics, e.g. different centrality measures [28], affect opinion dynamics and resulting opinion in consensus if it is reached. We consider two key performance indicators of opinion dynamics, namely, winning rate and consensus time.

Social power (influence centrality) is a concept ranks the importance of nodes in a network. Centrality measures are used to identify the most powerful nodes in a network. Understanding the powerful nodes is very important for opinion dynamics, which can help us to know which nodes play a crucial part for spreading opinions. The most common centrality measures are betweenness centrality [29], closeness centrality [30–32] and degree centrality [33]. There are also some centrality measures based on random walks, such as a random walk occupation centrality [34], which is the frequency of a node in the network being accessed during a random walk, a random walk betweenness centrality [35], that is the proportion of the paths through a node to all paths during a random walk. A random walk betweenness centrality does not depend on the shortest path, therefore it is more general than betweenness centrality. We also mention a random walk closeness centrality, that is a variant of

closeness centrality [34], and the computation of random walk closeness centrality is based on the mean first-passage time (MFPT). The analytical expressions of random walk based on centrality measures can be found in [34]. Game-theoretic network centrality is a flexible and sophisticated approach to identify the most powerful nodes in a network, the idea of which is borrowed from cooperative game theory. Tarkowski provided a good review of game-theoretic network centralities in [36]. The Shapley value [37] and the Myerson value [38] are both concepts from cooperative theory which are used to fairly distribute the total payoff among players based on their marginal contribution. In the paper [39], the authors introduce how to use the Shapley value to determine the top- k nodes in the social network. Mazalov et al. propose a game-theoretic centrality measure for weighted graph based on the Myerson value in [40]. In [41], Mazalov and Khitraya propose a modified Myerson value for unweighted undirected graphs. The characteristic function used for this modification considers not only simple paths but also includes cycles. The next work of Mazalov and Khitraya is [42], where the authors introduce the concept of integral centrality for unweighted directed graphs and provide an accurate mathematical proof that this centrality measure satisfies the Boldi-Vigna axioms [43].

In Chapter 3, we examine the connection between centrality measures and opinion dynamics based on the Zachary’s karate club network, and proposed two fast and accurate algorithms to approximate the game-theoretic centrality measures and examine them in a randomly generated network and the Zachary’s karate club network.

If we consider features in the machine learning model as nodes in the network, and the importance of features as the power of nodes, then the centrality measures can be used to explain the machine learning models. The feature importance is a crucial concept in machine learning, which can help us to understand the model and make the right decision which features to choose for creating an accurate model. Arrieta et al. reviewed concepts related to explainable AI and analyzed the types of explanations provided by XAI, primarily categorized into two types: global and local explanations [44]. The methods of local explanations in XAI focus on providing explanations for individual samples or predictions [45], while the methods of global explanations provide explanations for the whole model [46]. These methods are highly beneficial to help users understand the decision-making process and feature importance of a model, enhancing trust in the model, providing explanation-based decision support. In local explanations, LIME (Local Interpretable Model-Agnostic Explanations) [47] and SHAP (Shapley Additive Explanations) [48] are two com-

monly used methods. In global explanations, Feature Importance and SHAP can also be used to provide explanations [49]. In Chapter 4, we proposed a new method of global explanations and compared it with the state-of-the-art methods, such as SHAP and Feature Importance.

Summarizing the above, it is essential and significant to investigate the opinion dynamics in social networks with complex configurations, in particular, different network structures and individual's behavior; the centrality measures and their connection with opinion dynamics, and the practical applications of the developed centrality measures. The results of this research can be applied to various fields, such as sociology, psychology, physics, computer science, and artificial intelligence.

Goals of the thesis

The goal of the thesis is to study the opinion dynamics in social networks with complex configurations by different approaches and from various perspectives, modeling the opinion dynamic in a statistical-physical (macroscopic) or real network (microscopic) way. In modeling, we also assume existence of hypocrisy individuals, from different network properties, in particular, from the point of view of centrality measures to study the propagation processes and the practical applications of the developed methods. To pursuit the central goal, several specific questions are addressed and answered throughout the four chapters in the thesis.

This thesis can be considered as a series of results including modeling, analyzing the opinion dynamics in social network with complex configurations, and the analysis of network properties with their applications.

Main tasks

To achieve the goals of the thesis, the following key tasks are identified:

1. Modeling opinion dynamics in two-layer networks with hypocrisy individuals by statistical-physical methods and probability theory assuming that interaction in the internal layer is allowed, which is more realistic than the concealed voter model. Compare opinion propagation processes in specific internal network structures (i.e. empty, complete, cycle, star, two-star and two-clique) and initial individual's configurations. The key indicators are the time to reach consensus (consensus time) and the winning rate of the specific opinion (e.g., the red opinion).

2. Using microscopic approach to model the opinion dynamics in the same assumption as in Task 1. Provide the model for a two-layer network. The model should be general enough for any given two-layer network structure satisfying the definition. Simulate the opinion dynamics process in various network structures and individuals' behaviors. Analyze how network properties and individual's behavior affect the key indicators (consensus time and winning rate).
3. Generating the more realistic two-layer network according to the definition of a two-layer network and a real dataset (i.e. Zachary's karate club network). Examining the microscopic model on the generated network. Centrality measures are used to identify the most powerful nodes in the network, but not much is known about centrality measures in a two-layer network. Therefore, we should figure out how to simplify the two-layer network to one-layer network and apply the centrality measures to the reduced network. Moreover, the novel centrality measures should be developed and verified. The connection between centrality measures and key indicators in opinion dynamics should be examined.
4. Finding the practical applications of the developed centrality measures. The feature importance is a crucial concept in machine learning, which can help us to understand the model and make the right decision. The developed centrality measures can be used as the feature importance in the machine learning models if we consider the features as nodes. Develop a new feature importance algorithm based on the developed centrality measures and compare its work with the state-of-the-art methods (SHAP, LIME, and Feature Importance).

Scientific novelty

In this thesis, we proposed a general concealed voter model (GCVM), in which individuals interact in two layers and can exchange their opinions in the internal layer. This interaction is not allowed in a concealed voter model (CVM). By exchanging opinions in the internal layer we mean that individuals share their real or internal opinions with their close friends. There are two version of GCVM, the macroscopic-GCVM and microscopic-GCVM. The macroscopic-GCVM is based on the statistical-physical methods and probability theory, while the microscopic-GCVM is based on a real network structure modeling.

For the macroscopic-GCVM, the process of opinion formation in GCVM with different internal structures is presented in this thesis. We make the series of numerical simulations of macroscopic-GCVM with different network structures (both external and internal) and get some counterintuitive conclusions. For instance, we find out that sometimes with a relatively simple network structure of an external layer the consensus within the individuals' opinions cannot be reached, and if individuals in the network are not good at expressing their opinions publicly (in an external layer), exchanging opinions with their close friends (in an internal layer) is almost useless.

For the microscopic-GCVM, we provide the definition of a two-layer network for opinion dynamics. Macroscopic-GCVM uses only statistical-physical methods, while microscopic-GCVM starts with a real network. We conduct a series of simulations with different network structures and individuals' behaviors. Moreover, we propose and validate the hypothesis that there exists a strong linear relationship between a consensus time and pairwise average shortest paths d in the network structure. We performed a controlled variable approach to validate the impact of each individual parameter on key performance indicators (KPIs) including a consensus time and winning rate. Furthermore, we assess the influence of parameter combinations (some specific behaviors of individuals) on KPIs by analyzing the results using the K-means algorithm. We conclude that certain parameter combinations can have a significant impact on the consensus time.

We examine centrality measures defined on two-layer networks. The approach to simplify the two-layer network related to opinion dynamics to one layer weighted network is proposed. The two fast and accurate algorithms for one-layer weighted network to approximate the game-theoretic centrality measures are proposed and connection between centrality measures and characteristics of opinion dynamic processes in such networks is examined. As an example, the Zachary's karate club social network is considered and extended by adding the second (internal) layer of communication. The structures of the external and internal layers may be different. By key characteristics of the opinion dynamic process we mean a consensus time and winning rate of a particular opinion. Significantly strong positive correlation between internal graph density and consensus time, and significantly strong negative correlation between centrality of authoritative nodes and consensus time are found.

A new Explainable Artificial Intelligence (XAI) method called ShapG (Explanations based on the Shapley value for Graphs) for measuring feature importance is developed. ShapG is a model-agnostic global explanation method. At the first

stage, it defines an undirected graph based on the dataset, where nodes represent features and edges are added based on calculation of correlation coefficients between features. At the second stage, it calculates an approximated Shapley value by sampling the data taking into account this graph structure. The sampling approach of ShapG allows to calculate the importance of features efficiently, i.e. to reduce computational complexity. Comparison of ShapG with other existing XAI methods shows that it provides more accurate explanations which is shown on two examined datasets. We also compared other developed XAI methods based on cooperative game theory with ShapG in running time, and the results show that ShapG exhibits obvious advantages in its running time, which further proves efficiency of ShapG. In addition, extensive experiments demonstrate a wide range of applicability of the ShapG method for explaining complex models. ShapG is found as an important tool in improving explainability and transparency of AI systems and we believe it can be widely used in various fields.

Research methods

This thesis uses the methods of modeling in statistical physics (Monte Carlo method, molecular dynamics simulation), statistics (correlation coefficient, hypothesis testing), probability theory (distributions of random variables and stochastic processes), game theory (elements of cooperative game theory including the Shapley value and the Myerson value), machine learning (SHAP, LIME, Feature Importance, LightGBM, and multilayer perceptron), graph theory (centrality measures and shortest path), and optimization theory (K-means algorithm).

Theoretical and practical significance

The results presented in this thesis are focused in opinion dynamics modeling, network analysis and their applications. Their theoretical significance can be summarized as follows: (i) the mathematical formulations of statistical-physics model and microscopic model of opinion dynamics in social networks with complex configurations are proposed. This provides a new perspective in opinion dynamics modeling in social networks with complex configurations, and can be used to study the impact of different network structures and individual's behavior on opinion propagation; (ii) a general definition of two-layer network with replica nodes is proposed, a new approach to simplify the two-layer network to one-layer weighted network based on

given opinion dynamics is developed; (iii) two fast and accurate algorithms to find centrality measures based on game theory are developed; (iv) the novel approach of defining the feature importance in machine learning models based on centrality measures is proposed, and the new XAI method called ShapG is developed.

The practical significance of the results is manifested in the research on design of effective communication strategies, targeted marketing, political campaigning, public health interventions, and artificial intelligence. The developed centrality measures can be used to identify the most powerful nodes in a network, which is crucial not only for opinion dynamics but also for transportation networks, communication networks, etc. The developed feature importance algorithm (ShapG) can be used to explain the machine learning models, design the optimal training strategy, save energy on machine learning model training and contribute to carbon neutrality.

In Chapter 1, a series of numerical simulations based on the mathematical form of macroscopic-GCVM is conducted. It is found that the external cyclic structure can significantly prolong the consensus time, and if individuals in the network are not good at expressing their opinions publicly, exchanging opinions with their close friends is almost useless. In Chapter 2, we propose and validate the hypothesis that there exists a strong linear relationship between a consensus time and pairwise average shortest paths d in the network structure, which is consistent with the results of examining the macroscopic-GCVM model. Chapter 3 of the thesis is devoted to the investigation of the relationship between centrality measures and opinion dynamics in the Zachary’s karate club network. A significantly strong negative correlation between centrality of authoritative nodes and consensus time is found. In Chapter 4, the global explanation method based on the Shapley value for graphs (ShapG) is developed, which is more accurate and efficient than the state-of-the-art methods (SHAP, LIME, and Feature Importance).

The research conducted in the thesis is supported by the Chinese Government Scholarship (CSC) No. 202109010084 (2021–2025); the Russian Science Foundation (RSF) grant No. 22-21-00346 “Game theoretic methods of opinion dynamics control in social networks” (2022–2023).

Brief description of the thesis structure

The thesis consists of an introduction, four chapters, conclusions, list of acronyms and symbols, list of figures, list of tables, bibliography and appendix.

The content of each chapter is composed by basic notations, model description and definitions, the detailed design of experiments, the analysis of the results, and a brief summary of the chapter. This thesis contains 149 pages (158 pages in a Russian version) including 38 figures and 34 tables. The bibliography cites 109 items.

The first chapter of the thesis is devoted to examining the opinion dynamics in two-layer networks with hypocritical individuals using a macroscopic approach based on statistical physics and probability theory methods. In Section 1.1, the basic voter model (BCM) and concealed voter model (CVM) are introduced. In Section 1.2, the motivation for general concealed voter model (GCVM) is given (i.e. introduction of the interaction in the internal layer), and the mathematical formulation of macroscopic-GCVM with different network structures is proposed. The design of experiments is introduced and the numerical simulation results are presented and discussed in Section 1.3. Section 1.4 includes a brief summary of the first chapter.

In contrast to the first chapter, the second chapter focuses on a microscopic approach to model opinion dynamics in two-layer networks with hypocritical individuals. In Section 2.1 and 2.2, the definition of multi-layer and two-layer network with replica nodes is given, respectively. The comparison of microscopic-GCVM with macroscopic-GCVM is presented in Section 2.3. Then in Section 2.4, the experiment design and the observations are presented and discussed. We briefly summarize the second chapter results in Section 2.5.

Instead of using the classical network structures, the third chapter adopts a real network structure, i.e. the Zachary's karate club network, and extends it by adding the second (internal) layer of communication. The relationship between centrality of authoritarian nodes and key performance indicators (consensus time and winning rate) is examined there. Specifically, Section 3.1 introduces the approach to simplify the two-layer network with the given opinion dynamics to one-layer weighted network. In Section 3.2, the Zachary's karate club network setting is given. In Section 3.3, some network properties (pairwise average shortest paths, and graph density), a series of centrality measures (classical, game-theoretic, and random walk-based centralities), and two algorithms to approximate the game-theoretic centrality measures are introduced. In Section 3.4, we verified the efficiency of the simplification approach, the accuracy and performance of the game-theoretic centrality measures, and the correlation between network properties and key performance indicators (consensus time and winning rate). Section 3.5 finally concludes the third chapter.

By adopting the idea of approximated game-theoretic centrality measures from the third chapter, the fourth chapter focuses on the application of the developed algorithm in explainable artificial intelligence. In Section 4.1, we briefly introduce the existing methods of explainable artificial intelligence. In Section 4.2, we propose a new method of global explanation. The detailed description of the experiments are presented in Section 4.3. In Section 4.4, we compare the results of the proposed method with the state-of-the-art methods, such as SHAP and Feature Importance. Section 4.5 finally concludes the fourth chapter.

The conclusion of the thesis contains a brief description of the results obtained in the work.

Results submitted for defense

1. The mathematical formulation of opinion dynamics called a general concealed voter model (GCVM) is proposed. The model generalizes a concealed voter model (CVM) by introducing internal interactions between agents.
2. The mathematical formulation of statistical-physics model (macroscopic-GCVM) of opinion dynamics in different structures of two-layer networks with hypocrisy and internal interaction is proposed. The relationship between individual's characteristics and opinion dynamics processes (consensus time and winning rate) is examined within this model.
3. The definition of two-layer network with replica nodes is proposed. The microscopic-GCVM model for any network structure satisfying this definition is introduced. Methodology for validating the impact of each individual parameter on consensus time and winning rate is proposed. The relationship between network properties and opinion dynamics processes is examined within this model of opinion dynamics.
4. The approach to simplify the two-layer network with the given opinion dynamics to one-layer weighted network is introduced. Novel approaches to approximate the game-theoretic centrality measures speeding up the original measures with a high accuracy are proposed. The connection between centrality measures and opinion dynamics processes is examined.
5. The novel feature importance method called ShapG used to explain the machine learning models is proposed. The methodology of ShapG is based on

considering the features in machine learning models as nodes in the network, then the feature importance can be considered as the centrality of nodes. The detailed algorithm of ShapG method is developed in the thesis. The tests of ShapG demonstrate that it is more accurate and efficient than the state-of-the-art methods (SHAP, LIME, and Feature Importance).

6. An open-source implementation of the global explanation algorithm ShapG [50] is developed.
7. Golang programs for simulating microscopic and macroscopic GCVM with different network structures and individuals' behaviors are developed.

Main scientific results

1. The General Concealed Voter Model including the mathematical formulation and corresponding interaction mechanisms of opinion dynamics processes is proposed, see papers [24, 25, 27] in the bibliography (personal contribution is at least 70%).
2. The relationships between individual's characteristics, network properties, and opinion dynamics processes (consensus time and winning rate) are examined through numerical simulations, see papers [24, 25, 27] in the bibliography (personal contribution is at least 70%).
3. The definition of two-layer network with replica nodes that suitable for microscopic-GCVM model with arbitrarily given network structure, see the paper [51] in the bibliography (personal contribution is at least 70%).
4. Experimental verification of the relationship between the centrality of authoritative nodes and the opinion dynamics processes is conducted, see papers [26, 51] in the bibliography (personal contribution is at least 70%).
5. The approach to simplify the two-layer network with given opinion dynamics to the one-layer weighted network, see papers [26, 51] in the bibliography (personal contribution is at least 70%).
6. Novel approaches to approximate the game-theoretic centrality measures with high performance and accuracy, see the paper [51] in the bibliography (personal contribution is at least 70%).

7. ShapG, a novel feature importance method to explain machine learning models, is proposed, see the paper [52] in the bibliography (personal contribution is at least 70%).
8. An open-source implementation of the global explanation algorithm ShapG is developed, see the entry [50] in the bibliography (personal contribution is 100%).
9. A Golang program aiming to simulate the macroscopic GCVN with different network structure and individuals' behavior is developed, see the program certificate [53] in the bibliography (personal contribution is 100%).

Verification of results

The main results of the thesis were presented at the International Conferences “Stability and control processes” (Saint Petersburg, 2022); International Conferences “Game Theory and Management” (Saint Petersburg, 2023, 2024); Scientific School “Dynamics of Complex Networks and their Applications” (2023); “Dynamic games and applications” seminar of GERAD (Online, 2024); International Conference “Mathematical Optimization Theory and Operations Research” (Omsk, 2024).

Publications

Based on the results of the thesis, the following works were published: [24–27], they all are indexed in Scopus. The following items [24, 25] are published in peer-reviewed journals from the list of the Higher Attestation Commission. These papers [51, 52] are submitted to journals and now under review. The certificate of registration of the computer program [53] with registration number No. 2023661532 has been obtained. The open-source program [50] is developed and released to the python package index.

- [24] Zhao C., Parilina E. M. Opinion Dynamics in Two-Layer Networks with Hypocrisy // Journal of the Operations Research Society of China. – 2024. – Mar. – Vol. 12, no. 1. – P. 109-132. – Access mode: <https://doi.org/10.1007/s40305-023-00503-2>.
- [25] Zhao C., Parilina E. M. Analysis of consensus time and winning rate in two-layer networks with hypocrisy of different structures // Vestnik of Saint Peters-

- burg University. Applied Mathematics. Computer Science. Control Processes. – 2024. – Vol. 20, no. 2. – P. 170-192.
- [26] Zhao C., Parilina E. M. Network Structure Properties and Opinion Dynamics in Two-Layer Networks with Hypocrisy // Mathematical Optimization Theory and Operations Research. – Cham : Springer Nature Switzerland. – 2024. – P. 300-314.
- [27] Zhao C., Parilina E. M. Consensus time and winning rate based on simulations in two-layer networks with hypocrisy // 2023 7th Scientific School Dynamics of Complex Networks and their Applications (DCNA). – 2023. – P. 68-71.
- [50] Zhao C. shapG. – PyPI: <https://pypi.org/project/shapG/>; GitHub: <https://github.com/vectorsss/shapG>. – 2024. – July. – (Accessed on 11/16/2024).
- [51] Zhao C., Parilina E. M. Centrality measures and opinion dynamics in two-layer networks with replica nodes // arXiv preprint arXiv:2406.18780v2. – 2024. – 2406.18780v2.
- [52] Zhao C., Liu J., Parilina E. M. ShapG: new feature importance method based on the Shapley value // arXiv preprint arXiv:2407.00506. – 2024. – 2407.00506.
- [53] Certificate of state registration of the computer program No. 2023661532 Russian Federation : Program for modeling the dynamics of binary opinion spread in two-layer networks Zhao C. (CN) ; Federal State Budgetary Educational Institution of Higher Education “Saint Petersburg State University” (SPbSU) ; Zhao Chi. – no. 2023660681 ; req. 24.05.2023 ; publ. 01.06.2023.

Acknowledgments

The author is profoundly grateful to many passionate and generous teachers who guided him, making this thesis possible. Heartfelt thanks to Doctor of physical and mathematical sciences, professor Parilina Elena Mikhailovna (Saint Petersburg State University), for her invaluable guidance, support, and enlightening discussions that contributed greatly to his growth as a researcher. The author also wishes to thank Gankevich Ivan Gennadevich, his first team leader at Huawei, from whom he learned a lot during their time working together. The template of this thesis is

mostly based on Gankevich's open-source work. The author would like to extend his gratitude for his girlfriend Chelnokova Anastasia Dmitrievna, for her unwavering love, understanding and constant support under any circumstances. A special acknowledgment goes to his grandfather, his first mathematics teacher, for inspiring his love of learning. The author expresses heartfelt appreciation to his family for their unwavering support and encouragement throughout this challenging but rewarding journey. Furthermore, he would like to acknowledge the support and encouragement received from the China Scholarship Council during his doctoral studies.

Chapter 1. Opinion Dynamics in Two-Layer Networks with Hypocrisy

In this chapter, we add the internal interaction mechanisms to the concealed voter model and formularize the macroscopic-GCVM for different network structures. A series of numerical simulations is conducted, and the results are analyzed. We also discuss the impact of the network structure on the consensus time and the winning rate of an opinion. Results presented in this chapter are published in the paper [24].

1.1 Basic and Concealed Voter Models

We assume that the society is represented by the agents or individuals who have opinions on the topic. The opinion is binary, i.e., each individual's opinion is red or blue. We introduce the models of opinion dynamics taking into account the network structure the individuals form within the social network which we call general concealed voter model (GCVM). Before introducing GCVM, we briefly introduce a basic voter model (BVM) and concealed voter models (CVM) (see [21, 22]).

We call a classical voter model as a basic voter model. BVM assumes that everyone in a network can express his opinion publicly, so there is only one layer of information exchange and a unique opinion of an individual about the topic in this model of opinion dynamics. In CVM described in Section 1.1.2, individuals can either express or conceal their real (hidden) opinion publicly. An individual has both publicly expressed and private opinions about the topic, and the private opinion is unknown to other individuals. If an individual conceals his real opinion, i.e. his publicly known opinion is different from the private one, we call him *hypocrisy*: an individual with cognitive dissonance [54]. There are two layers (external and internal) in the CVM but the information exchange in the internal layer is not allowed.

We focus on the time when groups under different network structures reach consensus, this time is called consensus time, and the winning rate of an opinion after a series of simulations.

1.1.1 Basic Voter Model (BVM)

Suppose we have a predefined network G representing the communication or network structure connecting players in the society. Denote the number of individuals in network G by N . We examine the evolution of network in continuous time. We use the following notations:

- $\omega_{ext}(\alpha, t) \in \{0, 1\}$ is the opinion of individual α at time t (0 is represented by blue and 1 is represented by red color);
- c is a copy rate, that is the probability of individuals adapt his neighbor's opinion;
- r is the number of individuals with red opinion;
- ρ is the proportion of individuals with red opinion in the population (the strength of red opinion), that is $\rho = r/N$.
- T_{cons} is the consensus time, in BVM it means the time required for all individuals to form the same opinion ($\rho = 1$ or $\rho = 0$).

We use the proportion of red opinion to represent the state of such a BVM system, and $\rho \in \{0, \frac{1}{N}, \dots, \frac{N-1}{N}, 1\}$. It is assumed that any individual can change his opinion as a result of a stochastic event realization. We suppose that the interval of such an event obeys the exponential distribution with arrival rate λ . Then we define the following stochastic matrix with elements $P(\rho, \cdot)$ for each $\rho \notin \{0, 1\}$, where $P(\rho, \rho')$ is the probability of transition of BVM system from state ρ to state ρ' after the event happens. If $\rho \in \{0, 1\}$, we have $P(0, 0) = 1$ and $P(1, 1) = 1$ meaning that if all individuals in the network have the same opinion, they will never change it. The proportion of red opinion ρ can be changed to $\rho \pm \frac{1}{N}$ or remains the

same with positive probabilities:

$$\begin{aligned}
P\left(\rho, \rho + \frac{1}{N}\right) &= \frac{c(N-r)r}{N(N-1)} = \frac{c(1-\rho)r}{N-1}, \\
P\left(\rho, \rho - \frac{1}{N}\right) &= \frac{cr(N-r)}{N(N-1)} = \frac{cr(1-\rho)}{N-1}, \\
P(\rho, \rho) &= 1 - 2\frac{cr(1-\rho)}{N-1}.
\end{aligned} \tag{1}$$

If we consider the change of the BVM system state as a random event, then the change rate per unit time in this system can be represented as the sum of positive direction change rate λ_+ and negative direction change rate λ_- , where

$$\begin{aligned}
\lambda_- &= cr\frac{N-r}{N} = \lambda_+ = c(N-r)\frac{r}{N}, \\
\lambda &= \lambda_+ + \lambda_- = 2c(N-r)\frac{r}{N}.
\end{aligned} \tag{2}$$

Therefore, we have a system with an arrival rate λ per unit time. The following steps show how individuals interact with each other in BVM:

1. **Initialization:** Given fraction ρ of the individuals with red opinion, other individuals hold blue opinion. Initialize the time $t \rightarrow 0$.
2. **Iteration:**
 - a. Choose a ‘‘focal’’ individual f uniformly at random from all of N individuals.
 - b. Pick a neighbor a of the focal individual uniformly at random from all of his neighbors.
 - c. Generate a temporary variable $\lambda_t = \lambda \cdot u$, where $u \sim U(0, 1)$, individual f adopts neighbor a 's opinion according to λ_t by the following way:
 - i. The state of this system will be changed from ρ to $\rho + \frac{1}{N}$, if $\lambda_t < \lambda_+$;
 - ii. The state of this system will be changed from ρ to $\rho - \frac{1}{N}$, if $\lambda_+ \leq \lambda_t \leq \lambda_+ + \lambda_- = \lambda$,
 - d. We increase the time by a random number Δt drawn from an exponential distribution with mean $\frac{1}{\lambda}$, $t \rightarrow t + \Delta t$.
 - e. If the group has reached a consensus, we set $t \rightarrow T_{\text{cons}}^{(\text{BVM})}$ and terminate. Otherwise we go back to step a.

Fig. 1 represents the network of BVM with 10 individuals on the complete network.

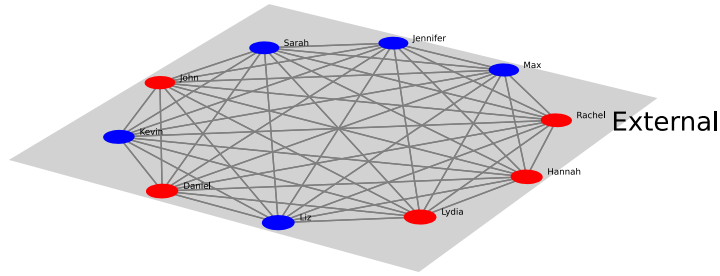


Figure 1: A complete network connecting 10 individuals in BVM

1.1.2 Concealed Voter Model (CVM)

In CVM the predefined networks G_1 and G_2 are given with the same set of N individuals. Individual i is represented by node E_i in network G_1 called an external layer, and by node I_i in network G_2 called an internal layer. Therefore, each individual is represented by a pair of nodes (E_i, I_i) , $i = 1, \dots, N$. We should notice that there is no restriction on the network structure G_1 but network G_2 is always an empty network with N individuals. The two-layer network structure satisfies the following properties:

1. There are no links between nodes I_i and I_j , $\forall i, j = 1, \dots, N$.
2. There must exist a link between E_i and I_i , $\forall i = 1, \dots, N$.

Fig. 2 represents such a two-layer network with 10 individuals when the external layer is a complete network.

In CVM, we use R, B (r, b) to represent individuals' external (internal) red and blue opinions respectively. Denote the state set of an individual's opinion by $S = \{Rr, Rb, Bb, Br\}$. We use the following notations for CVM systems:

- $\omega(\alpha, t) \in S$ is the state of individual α 's opinion at time t ;
- c is a copy rate (equivalent to the one in the BVM);

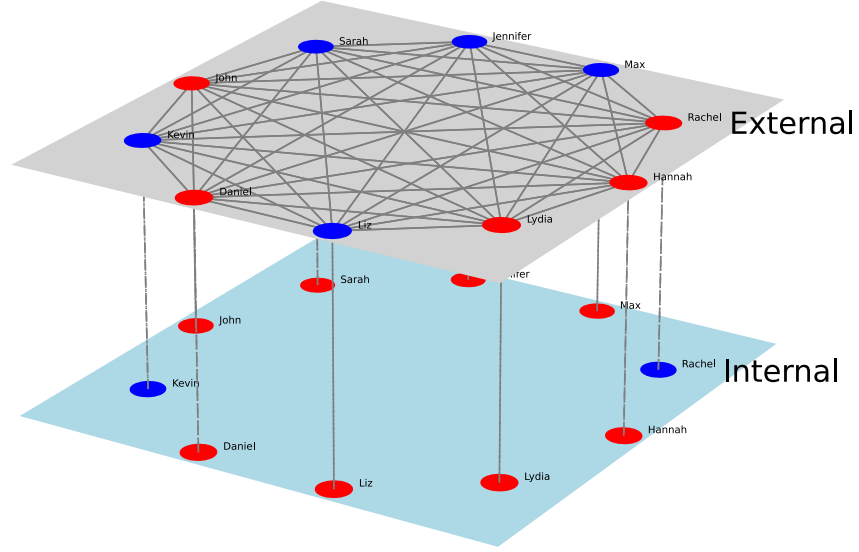


Figure 2: A two-layer network structure in CVM with 10 individuals

- e is the externalization rate, that is the probability of hypocrisy choosing to publicly express his internal opinion;
- i is the internalization rate, that is the probability of hypocrisy choosing to accept his external opinion;
- r_e is the number of individuals with external red opinion;
- r_i is the number of individuals with internal red opinion;
- r is the number of individuals with both external and internal red opinion;
- $\rho_{r_e}, \rho_{r_i}, \rho_r$ – the rate of external red, internal red and both external and internal red opinions.
- T_{cons} is the consensus time, in (G)CVM it means the time required for all individuals to form the same opinion in internal and external layers (i.e., $\rho_{r_e} = \rho_{r_i} = \rho_r = 0$ or $\rho_{r_e} = \rho_{r_i} = \rho_r = 1$).

In the network represented in Fig. 2 there are five hypocrisies.

There is a complete network in the external layer of CVM, and the two-layer network satisfies Properties 1-2 given above. Hypocrisy and cognitive dissonance can be reduced by externalization or internalization. Within externalization, the hypocrisy will express an opinion publicly, and within internalization, the hypocrisy accepts his previously expressed opinion.

If we are interested in the number of hypocrisies, we can focus on ρ_r describing the stochastic matrix Q_{CVM} in the CVM system, or we can describe the number of individuals having any of the state from set S (the latter is considered in [21]). Suppose the external layer in the CVM system is a complete graph, then the number of individuals of each state is as follows:

$$\begin{aligned}
N &= \#Rr + \#Rb + \#Br + \#Bb, \\
r_e &= \#Rr + \#Rb, \\
r_i &= \#Rr + \#Br, \\
r &= \#Rr, \\
\#Bb &= N - r_e - r_i + r, \\
\#Rb &= r_e - r, \\
\#Br &= r_i - r,
\end{aligned} \tag{3}$$

where $\#s$ is the number of individuals in state $s \in S$.

We can represent the state of the system with N individuals by triple $(\rho_{r_e}, \rho_{r_i}, \rho_r)$. The transitions with positive probabilities from one state to other states for the described CVM system are given in Table 1. We use Q to denote a transition matrix from here on.

The updating procedure of CVM is similar to BVM, we can summarize the differences as follows:

- Use a triple $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ instead of ρ to represent the state;
- As shown in Table 1, there are 6 possible state changes, we use the same scheme as in BVM, but with a different piecewise function to determine state changes;

Based on Table 1, we can describe the CVM system by expressing ρ_r , the proportion of hypocrisies in the network, given by equation (4), where $Q_{CVM}[x, y] = \{P(x, y)\}$ represents the probability of transition from state x (the previous value of ρ_r) to state y (the next value of ρ_r). For $\rho_r \notin \{0, 1\}$, we have equation (4) and

Table 1: Transitions from state $(\rho_r, \rho_{r_i}, \rho_r)$ in CVM

New state (x, y, z)	How can the new state be reached?	Probability	Transition rate matrix element Q_{CVM} per unit time
$(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r)$	$Bb \rightarrow Rb$, one individual with state Bb copies a neighbor with external opinion R	$\frac{c(N-r_e-r_i+r)r_e}{N(N-1)+r_e+r_i-2r}$	$\frac{c(N-r_e-r_i+r)r_e}{N}$
$(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r + \frac{1}{N})$	$Br \rightarrow Rr$, one individual with state Br copies a neighbor with external opinion R or expresses his internal opinion	$\frac{c(r_i-r)r_e+e(r_i-r)}{N(N-1)+r_e+r_i-2r}$	$(r_i - r)(c\frac{r_e}{N} + e)$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N})$	$Rb \rightarrow Rr$, one individual with state Rb accepts his external opinion	$\frac{i(r_e-r)}{N(N-1)+r_e+r_i-2r}$	$(r_e - r)i$
$(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r)$	$Rb \rightarrow Bb$, one individual with state Rb copies a neighbor with external opinion B or expresses his internal opinion	$\frac{c(r_e-r)(N-r_e)+e(r_e-r)}{N(N-1)+r_e+r_i-2r}$	$(r_e - r)(c\frac{N-r_e}{N} + e)$
$(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Br$, one individual with state Rr copies a neighbor with external opinion B	$\frac{cr(N-r_e)}{N(N-1)+r_e+r_i-2r}$	$\frac{cr(N-r_e)}{N}$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r)$	$Br \rightarrow Bb$, one individual with state Br accepts his external opinion	$\frac{i(r_i-r)}{N(N-1)+r_e+r_i-2r}$	$(r_i - r)i$
$(\rho_{r_e}, \rho_{r_i}, \rho_r)$	inverse to all above	one minus the sum above	negative sum of above elements

$$P(0, 0) = P(1, 1) = 1:$$

$$\begin{aligned}
P\left(\rho_r, \rho_r + \frac{1}{N}\right) &= Q_{CVM} \left[(\rho_{r_e}, \rho_{r_i}, \rho_r), \left(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r + \frac{1}{N}\right) \right] \\
&\quad + Q_{CVM} \left[(\rho_{r_e}, \rho_{r_i}, \rho_r), \left(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N}\right) \right], \\
P\left(\rho_r, \rho_r - \frac{1}{N}\right) &= Q_{CVM} \left[(\rho_{r_e}, \rho_{r_i}, \rho_r), \left(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r - \frac{1}{N}\right) \right] \\
P(\rho_r, \rho_r) &= 1 - P\left(\rho_r, \rho_r + \frac{1}{N}\right) - P\left(\rho_r, \rho_r - \frac{1}{N}\right).
\end{aligned} \tag{4}$$

1.2 General Concealed Voter Model

1.2.1 Motivation

The General Concealed Voter Model (GCVM) idea is straightforward. We assume that there exist connections in the internal layer. It is reasonable for an individual to have his own close friends and share his real (internal) opinion with them. Therefore, an individual and his friends may form either their own group or, a clique in the network. If the cliques are formed in the internal layer, and individuals never share true opinions with others outside the clique except they are connected in the internal layer. First, we define the transition probabilities for a special case of internal network when it is represented by a complete graph, then we describe the dynamics of the network in the case of incomplete internal layer.

1.2.2 Symmetric case: Complete internal network

First, we consider the simplest case of GCVM when both external and internal layers are complete networks. Figure 3 represents a symmetric case of GCVM with ten individuals.

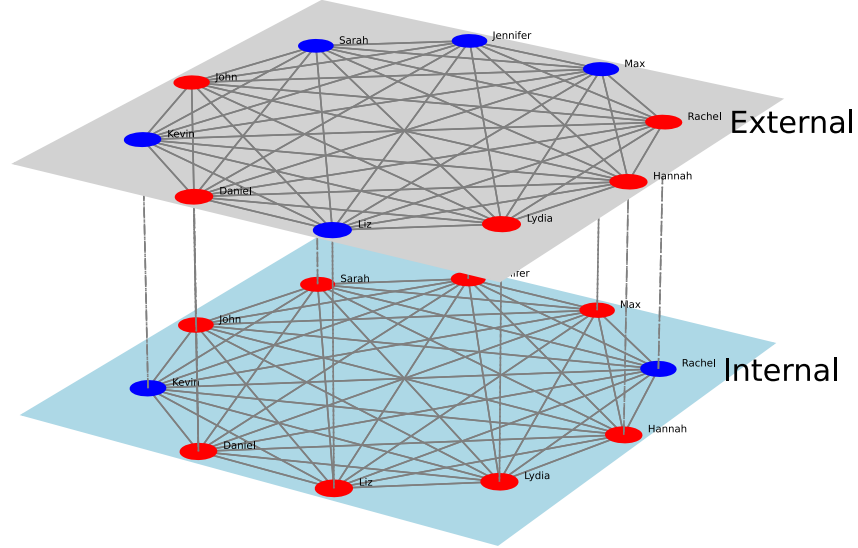


Figure 3: Representation of GCVM with 10 individuals: symmetric case

The transition matrix is represented in Table 2. There are eight cases of state transitions with nonzero probability while in the CVM system there are six cases, because in the GCVM system states Bb , Rr are allowed to be changed to Br and Rb respectively caused by internal interactions.

Compared to CVM, new GCVM assumes internal interactions which influence the internal opinion transition process. In CVM, the changes of internal opinion can only happen through internalization (an individual accepts his external opinion), while in GCVM, internal opinion can be changed through internalization or interaction between internal nodes (or individuals' interaction in the internal layer). For example, non-hypocrisy individuals (individuals with states Bb or Rr) can not change their states to Br , Rb respectively in CVM system, but it is possible in GCVM. Additionally, internal interaction will prolong the consensus time and change the probability of each opinion winning.

Table 2: Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVN system with symmetric case

New state (x, y, z)	How can the new state be reached?	Probability	Transition rate matrix element Q_{GCVN} per unit time
$(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r)$	$Bb \rightarrow Rb$, one individual with state Bb copies a neighbor with external opinion R	$\frac{c(N-r_e-r_i+r)r_e}{2N(N-1)+r_e+r_i-2r}$	$\frac{c(N-r_e-r_i+r)r_e}{N}$
$(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r + \frac{1}{N})$	$Br \rightarrow Rr$, one individual with state Br copies a neighbor with external opinion R or expresses his internal opinion	$\frac{c(r_i-r)r_e + e(r_i-r)}{2N(N-1)+r_e+r_i-2r}$	$(r_i - r)(c\frac{r_e}{N} + e)$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N})$	$Rb \rightarrow Rr$, one individual with state Rb accepts his external opinion or copies a neighbor with internal opinion r	$\frac{i(r_e-r) + c(r_e-r)r_i}{2N(N-1)+r_e+r_i-2r}$	$(r_e - r)(i + c\frac{r_i}{N})$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r)$	$Bb \rightarrow Br$, one individual with state Bb copies a neighbor with internal opinion r	$\frac{c(N-r_e-r_i+r)r_i}{2N(N-1)+r_e+r_i-2r}$	$\frac{c(N-r_e-r_i+r)r_i}{N}$
$(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r)$	$Rb \rightarrow Bb$, one individual with state Rb copies a neighbor with external opinion B or expresses his internal opinion	$\frac{c(r_e-r)(N-r_e) + e(r_e-r)}{2N(N-1)+r_e+r_i-2r}$	$(r_e - r)(c\frac{N-r_e}{N} + e)$
$(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Br$, one individual with state Rr copies a neighbor with external opinion B	$\frac{cr(N-r_e)}{2N(N-1)+r_e+r_i-2r}$	$\frac{cr(N-r_e)}{N}$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r)$	$Br \rightarrow Bb$, one individual with state Br accepts his external opinion or copies a neighbor with internal opinion b	$\frac{i(r_i-r) + c(r_i-r)(N-r_i)}{2N(N-1)+r_e+r_i-2r}$	$(r_i - r)(i + c\frac{N-r_i}{N})$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Rb$, one individual with state Rr copies a neighbor with internal opinion b	$\frac{cr(N-r_i)}{2N(N-1)+r_e+r_i-2r}$	$\frac{cr(N-r_i)}{N}$
$(\rho_{r_e}, \rho_{r_i}, \rho_r)$	inverse to all above	one minus the above sum	negative sum of above elements

Similarly, we can represent the stochastic matrix with respect to ρ_r . For $\rho_r \notin \{0, 1\}$, we have calculated transition probabilities by (5) with $P(0, 0) = P(1, 1) = 1$:

$$\begin{aligned}
P\left(\rho_r, \rho_r + \frac{1}{N}\right) &= Q_{GCV M} \left[(\rho_{r_e}, \rho_{r_i}, \rho_r), \left(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r + \frac{1}{N}\right) \right] \\
&\quad + Q_{GCV M} \left[(\rho_{r_e}, \rho_{r_i}, \rho_r), \left(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N}\right) \right], \\
P\left(\rho_r, \rho_r - \frac{1}{N}\right) &= Q_{GCV M} \left[(\rho_{r_e}, \rho_{r_i}, \rho_r), \left(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r - \frac{1}{N}\right) \right] \\
&\quad + Q_{GCV M} \left[(\rho_{r_e}, \rho_{r_i}, \rho_r), \left(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r - \frac{1}{N}\right) \right], \\
P(\rho_r, \rho_r) &= 1 - P\left(\rho_r, \rho_r + \frac{1}{N}\right) - P\left(\rho_r, \rho_r - \frac{1}{N}\right).
\end{aligned} \tag{5}$$

We use the same measure as in [21, 22] to represent the strength of the red opinion, that is

$$m(\rho_{r_e}, \rho_{r_i}) = \frac{i\rho_{r_e} + e\rho_{r_i}}{e + i}, \tag{6}$$

where $m \in [0, 1]$ is the weighted value that combines the proportion of red opinions in the internal and external layers, and m can be considered as an equivalent to ρ in BVM.

1.2.3 Asymmetric cases: Incomplete internal network

In this section, we consider several asymmetric cases including internal star-coupled network, internal two star-coupled networks, and internal two-clique networks while the external layer is always the complete network. We assume that there are two groups which are equal in size both in the internal two star-coupled and two-clique networks.

Internal layer: Star-coupled network

In this case, we have a central node in the internal layer (star graph) as represented in Fig. 4. In this case, the interaction in the internal layer exists only between the central node and any noncentral node. There is no interaction between noncentral nodes.

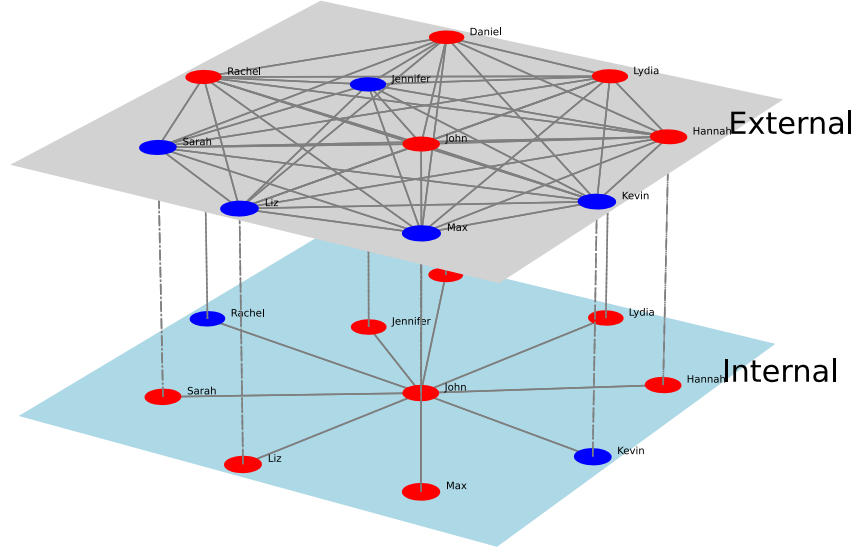


Figure 4: Representation of an internal star-coupled network

For the internal star-coupled network, there are 8 types of transitions with nonzero probabilities described in Table 3.

If we compare the transitions listed in Tables 2 and 3, we can conclude that the changes of the internal network structure influence only the transition rate of these transitions: $Rb \rightarrow Rr$, $Bb \rightarrow Br$, $Br \rightarrow Bb$ and $Rr \rightarrow Rb$. Therefore, in the following sections, where we consider other internal network structures, we will only show the transition rates for these four cases, all other transition rates ($Bb \rightarrow Rb$, $Br \rightarrow Rr$, $Rb \rightarrow Bb$ and $Rr \rightarrow Br$) remain the same as in Table 2.

Internal layer: Two star-coupled network

Assume that there is a two star-coupled network in the internal layer, and we consider two possible cases:

1. Two star-coupled networks share one common node (the total number of nodes is an odd number).
2. Two star-coupled networks share two common nodes (the total number of nodes is an even number).

Fig. 5a shows a simple example of the first case, each star subnetwork share a common node “Sarah”. If we define a number of hubs (central nodes) by a , then for this case $a = 2$. We also define ρ_g as a proportion of nodes in one-star subnetwork

Table 3: Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with internal star-coupled network

New state (x, y, z)	How can the new state be reached?	Transition rate matrix element Q_{GCVM} per unit time
$(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r)$	$Bb \rightarrow Rb$, one individual with state Bb copies a neighbor with external opinion R	$\frac{c(N-r_e-r_i+r)r_e}{N}$
$(\rho_{r_e} + \frac{1}{N}, \rho_{r_i}, \rho_r + \frac{1}{N})$	$Br \rightarrow Rr$, one individual with state Br copies a neighbor with external opinion R or expresses his internal opinion	$(r_i - r)(c\frac{r_e}{N} + e)$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N})$	$Rb \rightarrow Rr$, the authority in the internal layer has opinion Rb , he accepts his external opinion or copies a neighbor with internal opinion r ; other individuals have the opinion Bb , one of them accepts his external opinion, or copies the authority iff the authority has internal opinion r	$\frac{1}{N}(r_e - r)(i + c\frac{r_i}{N}) + \frac{N-1}{N}(r_e - r)(i + c\frac{r_i}{N^2})$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r)$	$Bb \rightarrow Br$, the authority in the internal layer has opinion Bb , he copies a neighbor with internal opinion r ; other individuals have the opinion Bb , one of them copies the authority iff the authority has internal opinion r	$\frac{1}{N}(N - r_e - r_i + r)c\frac{r_i}{N} + \frac{N-1}{N}(r_e - r)(i + c\frac{r_i}{N^2})$
$(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r)$	$Rb \rightarrow Bb$, one individual with state Rb copies a neighbor with external opinion B or express his internal opinion	$(r_e - r)(c\frac{N-r_e}{N} + e)$
$(\rho_{r_e} - \frac{1}{N}, \rho_{r_i}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Br$, one individual with state Rr copies a neighbor with external opinion B	$\frac{cr(N-r_e)}{N}$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r)$	$Br \rightarrow Bb$, the authority in the internal layer has opinion Br , he accepts his external opinion or copies a neighbor with internal opinion b ; other individuals have the opinion Br , one of them accepts his external opinion, or copies the authority iff the authority has internal opinion b	$\frac{1}{N}(r_i - r)(i + c\frac{N-r_i}{N}) + \frac{N-1}{N}(r_i - r)(i + c\frac{N-r_i}{N^2})$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Rb$, the authority in the internal layer has opinion Rr , he copies a neighbor with internal opinion b ; other individuals have the opinion Rr , one of them copies the authority iff the authority has internal opinion b	$\frac{r}{N}c\frac{N-r_i}{N} + \frac{r(N-1)}{N}c\frac{N-r_i}{N^2}$
$(\rho_{r_e}, \rho_{r_i}, \rho_r)$	inverse of all above	negative sum of above elements

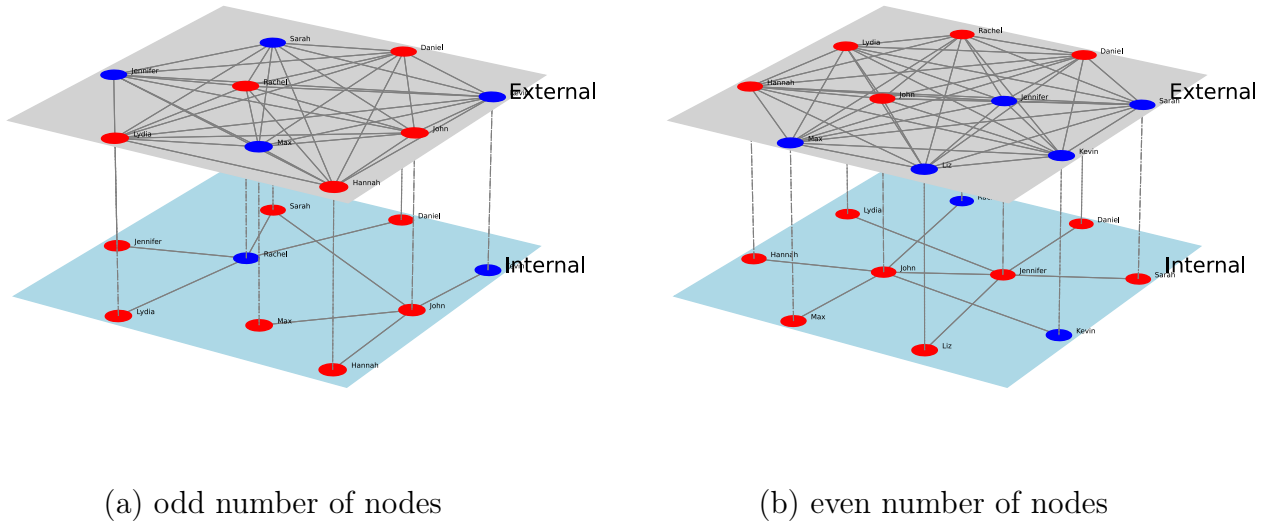


Figure 5: Representation of internal two star-coupled network

of the nodes in the whole network, and $\rho_g = \frac{N+1}{2N}$. We characterize the transitions for the first case of two star-coupled internal network in Table 4.

Fig. 5b shows an example when in the internal layer, two star-coupled networks are connected by one edge through their hubs (central nodes) [55]. There are two common nodes “John” and “Jennifer” in each star subnetwork, and each star subnetwork has five nodes. In this case, $\rho_g = \frac{N+2}{2N}$. We list the transitions in Table 5.

Internal layer: two-clique network

Assume that there are two cliques in the internal layer, and there is one inter-clique link in this network. Each clique has the same number of individuals. There are two possible cases:

1. Two cliques share a common node (this common node has degree $N - 1$);
2. Two cliques share two common nodes (these two common nodes have degree $\frac{2}{N}$).

Fig. 6a shows an example of a two-clique network when two cliques share a common node “Sarah”. The degree of node “Sarah” is $N - 1$. Other nodes have the same degree $\frac{N-1}{2}$, and $\rho_g = \frac{N+1}{2N}$ in this case. The transitions for this type of GCVM system are given in Table 6.

Table 4: Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with two star-coupled internal networks (odd case)

New state (x, y, z)	How can the new state be reached?	Transition rate matrix element Q_{GCVM} per unit time
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N})$	$Rb \rightarrow Rr$, the authority in the internal layer has opinion Rb , he accepts his external opinion or copies a neighbor with internal opinion r ; other individuals have the opinion Rb , one of them accepts his external opinion, or copies the authority iff the authority has internal opinion r	$\frac{a}{N}(r_e - r)(i + c\frac{r_i}{N}\rho_g) + \frac{N-a}{N}(r_e - r)i + \frac{N-a+1}{N}(r_e - r)c\frac{r_i}{N^2}$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r)$	$Bb \rightarrow Br$, the authority in the internal layer has opinion Bb , he copies a neighbor with internal opinion r ; other individuals have the opinion Bb , one of them copies the authority iff the authority has internal opinion r	$\frac{a}{N}(N - r_e - r_i + r)c\frac{r_i}{N}\rho_g + \frac{N-a+1}{N}(N - r_e - r_i + r)c\frac{r_i}{N^2}$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r)$	$Br \rightarrow Bb$, the authority in the internal layer has opinion Br , he accepts his external opinion or copies a neighbor with internal opinion b ; other individuals have the opinion Br , one of them accepts his external opinion, or copies the authority iff the authority has internal opinion b	$\frac{a}{N}(r_i - r)(i + c\frac{N-r_i}{N}\rho_g) + \frac{N-a}{N}(r_i - r)i + \frac{N-a+1}{N}(r_i - r)c\frac{N-r_i}{N^2}$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Rb$, the authority in the internal layer has opinion Rr , he copies a neighbor with internal opinion b ; other individuals have the opinion Rr , one of them copies the authority iff the authority has internal opinion b	$r\frac{a}{N}c\frac{N-r_i}{N}\rho_g + r\frac{(N-a+1)}{N}c\frac{N-r_i}{N^2}$

Table 5: Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with two star-coupled internal networks (even case)

New state (x, y, z)	How can the new state be reached?	Transition rate matrix element Q_{GCVM} per unit time
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N})$	$Rb \rightarrow Rr$, the authority in the internal layer has opinion Rb , he accepts his external opinion or copies a neighbor with internal opinion r ; other individuals have the opinion Rb , one of them accepts his external opinion, or copies the authority iff the authority has internal opinion r .	$\frac{a}{N}(r_e - r)(i + c\frac{r_i}{N}\rho_g) + \frac{N-a}{N}(r_e - r)(i + c\frac{r_i}{N^2})$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r)$	$Bb \rightarrow Br$, the authority in the internal layer has opinion Bb , he copies a neighbor with internal opinion r ; other individuals have the opinion Bb , one of them copies the authority iff the authority has internal opinion r .	$\frac{a}{N}(N - r_e - r_i + r)c\frac{r_i}{N}\rho_g + \frac{N-a}{N}(N - r_e - r_i + r)c\frac{r_i}{N^2}$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r)$	$Br \rightarrow Bb$, the authority in the internal layer has opinion Br , he accepts his external opinion or copies a neighbor with internal opinion b ; other individuals have the opinion Br , one of them accepts his external opinion, or copies the authority iff the authority has internal opinion b .	$\frac{a}{N}(r_i - r)(i + c\frac{N-r_i}{N}\rho_g) + \frac{N-a}{N}(r_i - r)(i + c\frac{N-r_i}{N^2})$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Rb$, the authority in the internal layer has opinion Rr , he copies a neighbor with internal opinion b ; other individuals have the opinion Rr , one of them copies the authority iff the authority has internal opinion b .	$r\frac{a}{N}c\frac{N-r_i}{N}\rho_g + r\frac{(N-a)}{N}c\frac{N-r_i}{N^2}$

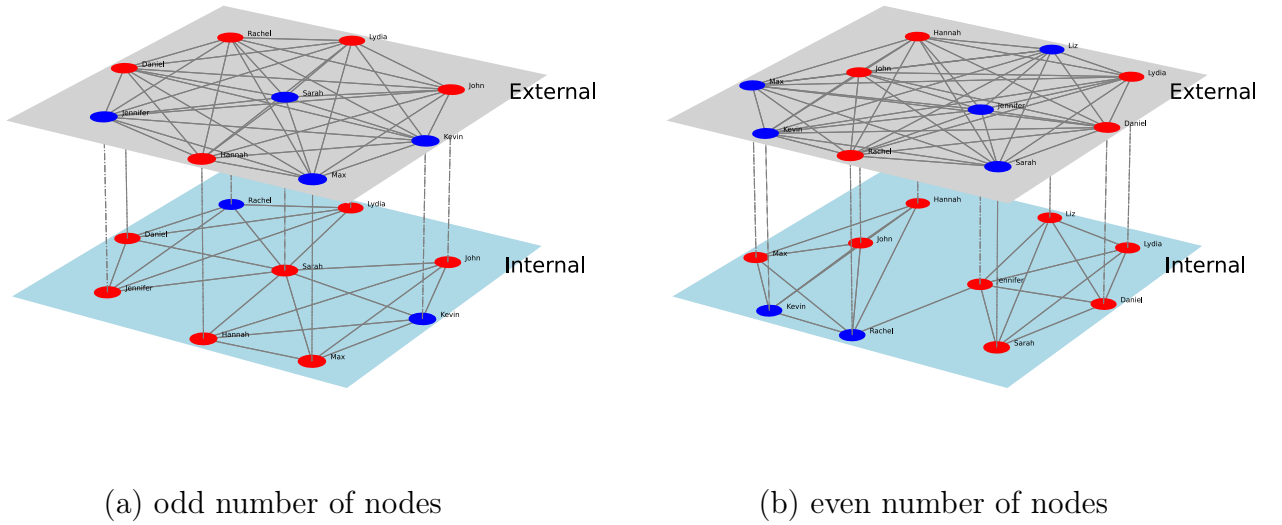


Figure 6: Representation of internal two-clique network

Fig. 6b shows a simple example of the second case of a two-clique internal network, when each clique has a special individual through whom two cliques are connected, i.e. there is a link connecting two special individuals, which is called an inter-clique link. In the second case, $\rho_g = \frac{N+2}{2N}$. The transition of this GCVN system are given in Table 7.

1.2.4 Asymmetric case: Incomplete external network

In this section, we assume that not only the internal layer but also the external layer can be incomplete graph. These changes in the external layer will affect four transitions: $Bb \rightarrow Rb$, $Br \rightarrow Rr$, $Rb \rightarrow Bb$, and $Rr \rightarrow Br$.

External layer: Cycle network

In this section, we assume that the external layer is given by a cycle. Then we represent how changes in the internal layer structure influence the transitions. We also consider the internal layers given by two-star and two-clique networks. The representations for such two-layer networks for GCVN are given in Figure 7, 8, 9.

Table 6: Transitions from the state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with internal two clique-coupled networks (odd case)

New state (x, y, z)	How can the new state be reached?	Transition rate matrix element Q_{GCVM} per unit time
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N})$	$Rb \rightarrow Rr$, the common node of the internal layer with opinion Rb accepts his external opinion or copies a neighbor with internal opinion r ; other individuals have the opinion Rb , one of them accepts his external opinion, or copies his neighbor's opinion.	$\frac{1}{N}(r_e - r)(i + c\frac{r_i}{N}) + \frac{N-1}{N}(r_e - r)(c\frac{r_i}{N}\rho_g + i)$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r)$	$Bb \rightarrow Br$, the common node of the internal layer has opinion Bb copies a neighbor with internal opinion r ; other individuals have the opinion Bb , one of them copies his neighbor with internal opinion r .	$\frac{1}{N}(N - r_e - r_i + r)c\frac{r_i}{N} + \frac{N-1}{N}(N - r_e - r_i + r)c\frac{r_i}{N}\rho_g$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r)$	$Br \rightarrow Bb$, the common node of the internal layer has opinion Br accepts his external opinion or copies a neighbor with internal opinion b ; other individuals have the opinion Br , one of them accepts his external opinion, or copies his neighbor with opinion b .	$\frac{1}{N}(r_i - r)(i + c\frac{N-r_i}{N}) + \frac{N-1}{N}(r_i - r)(i + c\frac{N-r_i}{N}\rho_g)$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Rb$, the common node of the internal layer has opinion Rr copies a neighbor with internal opinion b ; other individuals have the opinion Rr , one of them copies his neighbor with internal opinion b .	$r\frac{1}{N}c\frac{N-r_i}{N} + r\frac{(N-1)}{N}c\frac{N-r_i}{N}\rho_g$

Table 7: Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with internal two clique-coupled networks (even case)

New state (x, y, z)	How is the new state reached?	Transition rate matrix element Q_{GCVM} per unit time
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r + \frac{1}{N})$	$Rb \rightarrow Rr$, the common nodes of the internal layer have opinion Rb , one of them accepts its external opinion or copies a neighbor with internal opinion r ; other individuals have the opinion Rb , one of them accepts his external opinion, or copies his neighbor's opinion.	$\frac{a}{N}(r_e - r)(i + c\frac{r_i}{N}\rho_g) + \frac{N-a}{N}(r_e - r)(c\frac{r_i}{aN} + i)$
$(\rho_{r_e}, \rho_{r_i} + \frac{1}{N}, \rho_r)$	$Bb \rightarrow Br$, common nodes of the internal layer have opinion Bb , one of them copies a neighbor with internal opinion r ; other individuals have the opinion Bb , one of them copies his neighbor with internal opinion r ; other individuals have the opinion Bb , one of them copies his neighbor with internal opinion r .	$\frac{a}{N}(N - r_e - r_i + r)c\frac{r_i}{N}\rho_g + \frac{N-a}{N}(N - r_e - r_i + r)c\frac{r_i}{aN}$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r)$	$Br \rightarrow Bb$, common nodes of the internal layer have opinion Br , one of them accepts his external opinion or copies a neighbor with internal opinion b ; other individuals have the opinion Br , one of them accepts his external opinion, or copies his neighbor with opinion b .	$\frac{a}{N}(r_i - r)(i + c\frac{N-r_i}{N}\rho_g) + \frac{N-a}{N}(r_i - r)(i + c\frac{N-r_i}{aN})$
$(\rho_{r_e}, \rho_{r_i} - \frac{1}{N}, \rho_r - \frac{1}{N})$	$Rr \rightarrow Rb$, common nodes of the internal layer have opinion Rr , one of them copies a neighbor with internal opinion b ; other individuals have the opinion Rr , one of them copies his neighbor with internal opinion b .	$r\frac{a}{N}c\frac{N-r_i}{N}\rho_g + r\frac{(N-a)}{N}c\frac{N-r_i}{aN}$

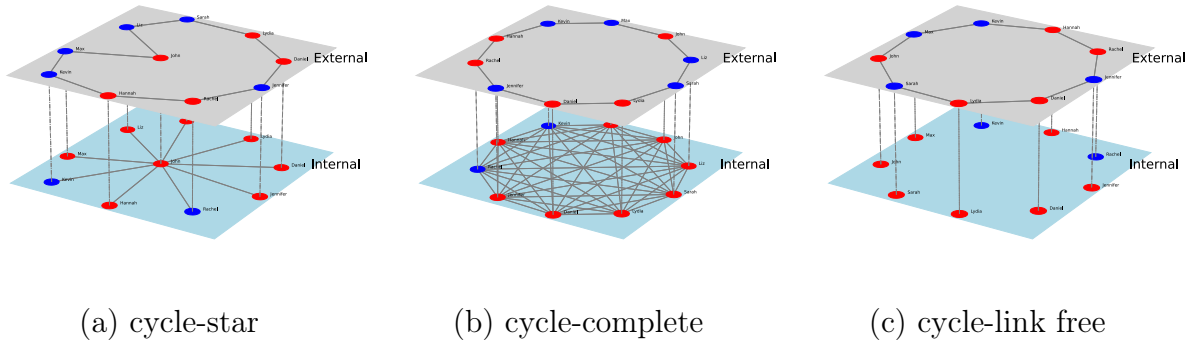


Figure 7: Representation of external cycle network

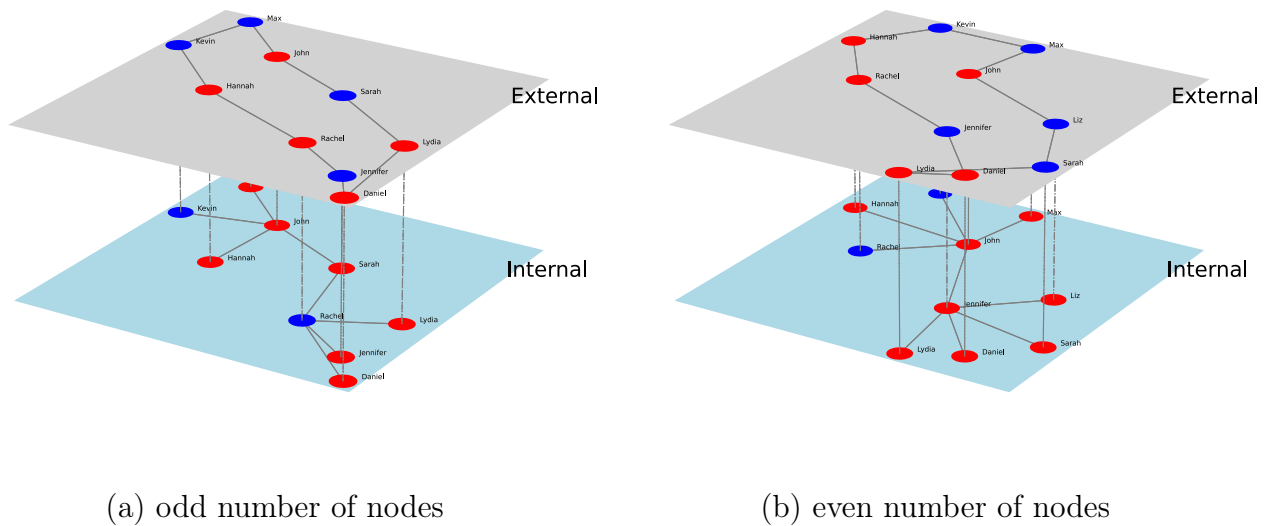
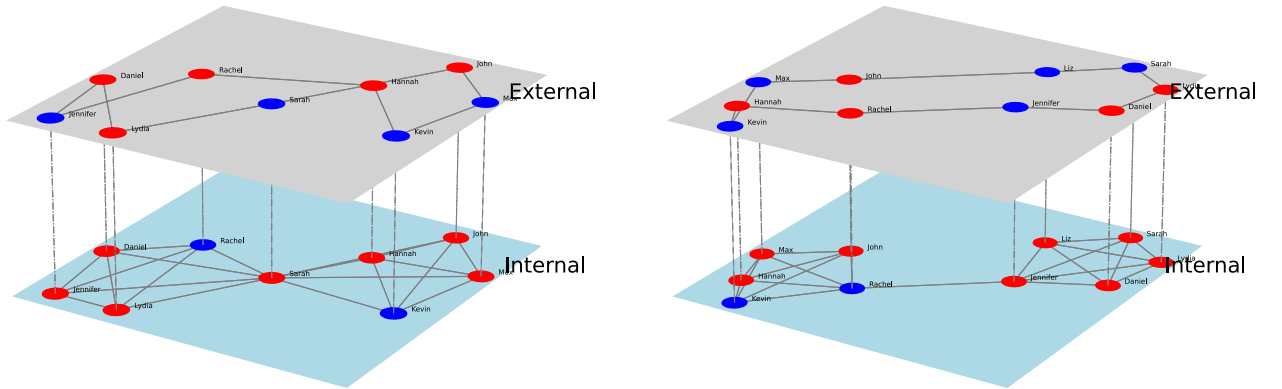


Figure 8: Representation of external cycle and internal two star-coupled network



(a) odd number of nodes

(b) even number of nodes

Figure 9: Representation of external cycle and internal two-clique network

1. $Bb \rightarrow Rb$: One individual with state Bb copies a neighbor with external opinion R . The changing rate is $\frac{c(N-r_e-r_i+r)r_e}{N} \frac{3}{N}$;
2. $Br \rightarrow Rr$: One individual with state Br copies a neighbor with external opinion R , or expresses his internal opinion. The changing rate is $(r_i - r)(c \frac{r_e}{N} \frac{3}{N} + e)$;
3. $Rb \rightarrow Bb$: One individual with state Rb copies a neighbor with external opinion B , or expresses his internal opinion. The changing rate is $(r_e - r)(c \frac{N-r_e}{N} \frac{3}{N} + e)$;
4. $Rr \rightarrow Br$: One individual with state Rr copies a neighbor with external opinion B . The changing rate is $\frac{cr(N-r_e)}{N} \frac{3}{N}$.

1.3 Experiments and Results

We use the Monte Carlo method to simulate the process of opinion transition in two-layer network. The simulations are organised as follows:

1. Set a random seed, the number of runs (denoted by n), and initialize the model by setting particular parameters;
2. Verify whether the consensus is reached for a given model (i.e. a unique opinion exists in the networks), and if so, stop the simulation, or else go to Step 3;
3. Calculate the transition rate of each possible state change under the current state;
4. Determine the next state under the assumption that the time between two transitions is exponentially distributed with a current change rate;
5. Go to Step 2.

For simplicity, we refer to a consensus time and a winning rate as KPIs and examine them. The simulation is organized as follows:

1. We run simulations varying parameters as shown below. Specifically, we focus on the strength of red opinion for each simulation, in this part we could see how the strength of red opinion influence the KPIs. Parameters used in the simulations are as follows:
 - **Set 1:** $\rho = 0.75, N = 400, r = 300, c = 1$ for BVM, and $m \approx 0.74, N = 400, r_e = 300, r_i = 100, r = 80, e = 0.01, i = 0.50, c = 1$ for GCVM;
 - **Set 2:** $\rho = 0.80, N = 500, r = 400, c = 1$ for BVM, and $m \approx 0.79, N = 500, r_e = 400, r_i = 100, r = 80, e = 0.01, i = 0.80, c = 1$ for GCVM;
2. We change the structure of internal layer and show how internal network structure influences the KPIs;
3. We modify the structure of external layer and observe the impact of different external network structures on KPIs.

The rest of this section demonstrates the most interesting results of our experiments.

Fig. 10 shows the consensus time for the first 20 simulations. We can see that the consensus time of BVM, CVM and GCVM with external complete network structure is in the same order of magnitude.

We can consider GCVM with incomplete external structure and compare the results with the previously described models. The results are summarized in Fig. 11.

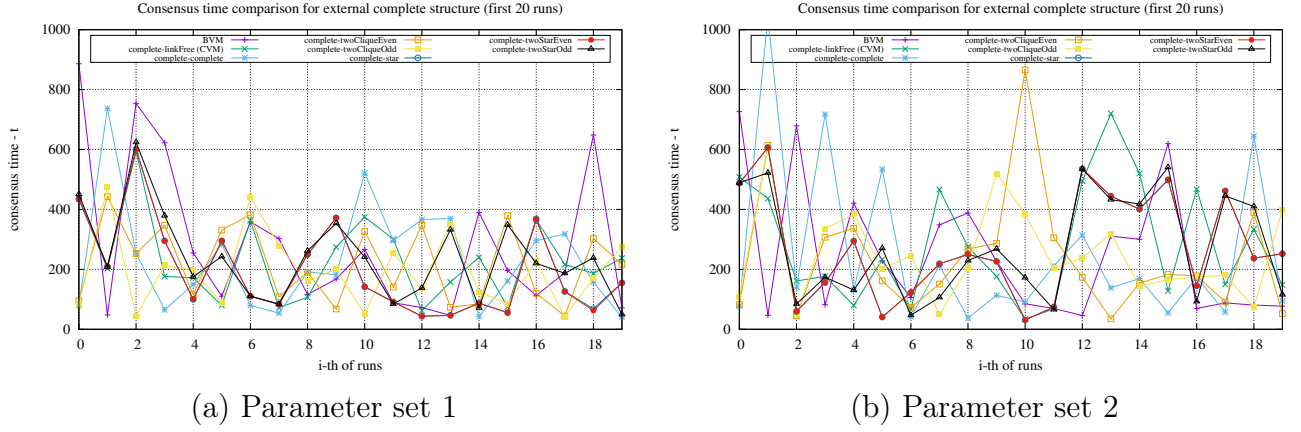


Figure 10: Consensus time for 20 runs, comparison between BVM, CVM and GCVN with external complete network structure

We highlight the surprising result: in the external layer the cycle structure is assumed to be much simpler than the complete structure, but it greatly prolongs the consensus time. One can observe the difference between the left and right subfigures — there is no “cycle-complete” model in the left subfigure. We do not put this model because in our simulations, there is no consensus on the model under parameter set 1. Even under parameter set 2, the consensus time for a structure “cycle-complete” is much longer than for other models.

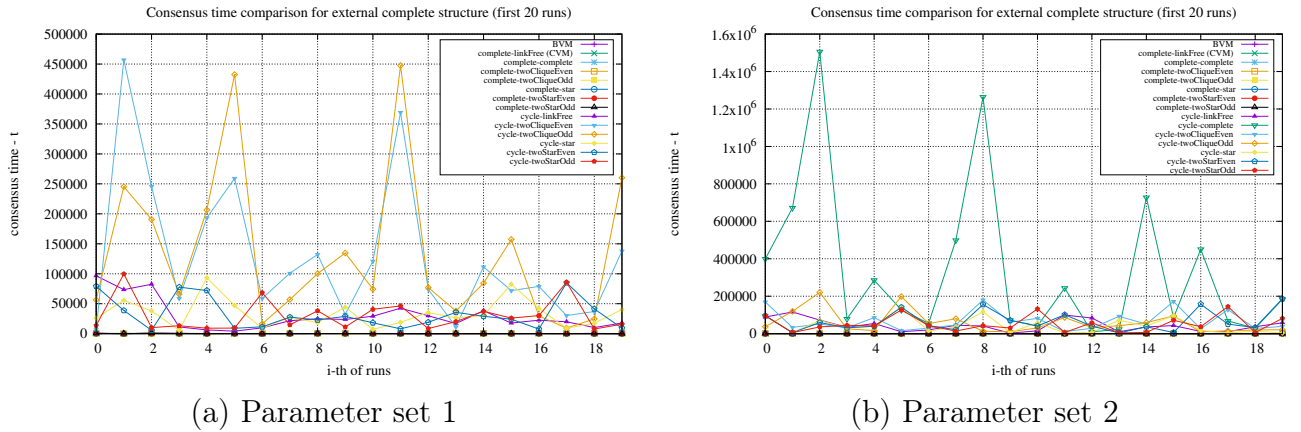


Figure 11: Consensus time for 20 runs, comparison between all models

Fig. 12 represents the observed average consensus time for all models. It is not too hard to make the same conclusion as the one based on Fig. 11 that the external layer cycle structure prolongs the consensus time significantly.

If we compare the observed average consensus time between BVM, CVM and GCVN with external complete structure, we can draw the following conclusions:

1. Multi-layer network structure of the society prolongs the consensus time in comparison with BVM, in which there is a unique communication layer;

2. Not all internal structures prolong the consensus time in CVM.

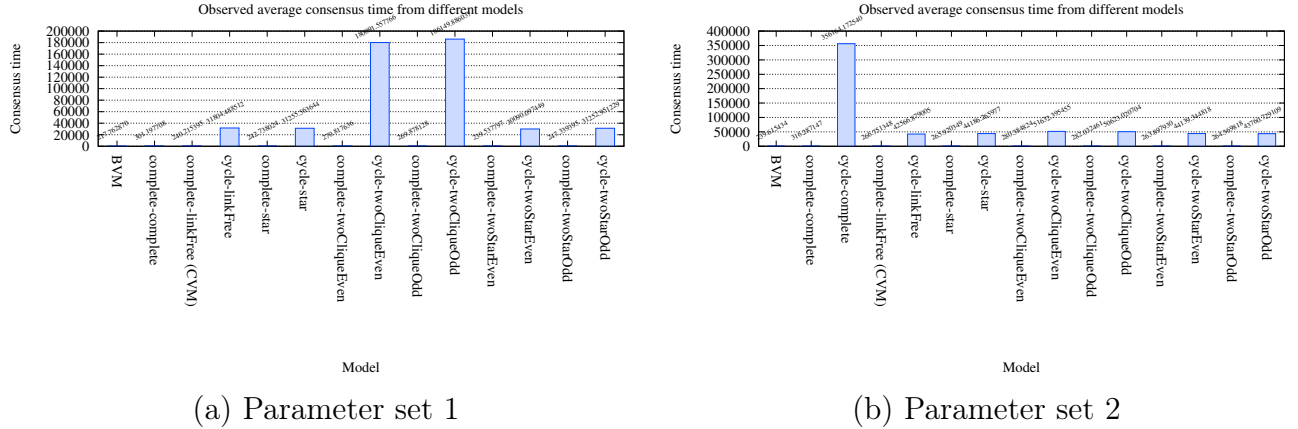


Figure 12: Observed average consensus time for all models

We present Fig. 13 based on 1000 simulations of each model. As the graph shows, BVM has the highest winning rate for both parameter sets. The case “cycle-complete” (external layer is cycle, internal is complete graph) is interesting because for parameter set 1 the consensus is not reached in our numerical simulations, therefore this structure is omitted in Fig. 12a. In the case “cycle-complete” with the parameter set 2, the consensus is reached the time to consensus is extremely high in comparison with all other structures of the layers. We also make an interesting observation: for the internal layer of a complex structure (complete, two-clique), the external cycle structure reduces the winning rate of the red opinion, which is close to 0, although the strength of the initial red opinion given by our parameters sets is very high. We can explain this as follows: when the external layer network structure is cyclic, the lower the average degree of nodes, the higher the consensus time, and the lower the winning rate. If we compare Fig. 13a with Fig. 13b, we can observe the higher strength of the red opinion and its higher winning rate. Therefore, our conclusion seems to be reasonable.

1.4 Conclusion to Chapter 1

In this chapter, we formularize the macroscopic general concealed voter model (macroscopic-GCVM) which is an extension of the concealed voter model. We use

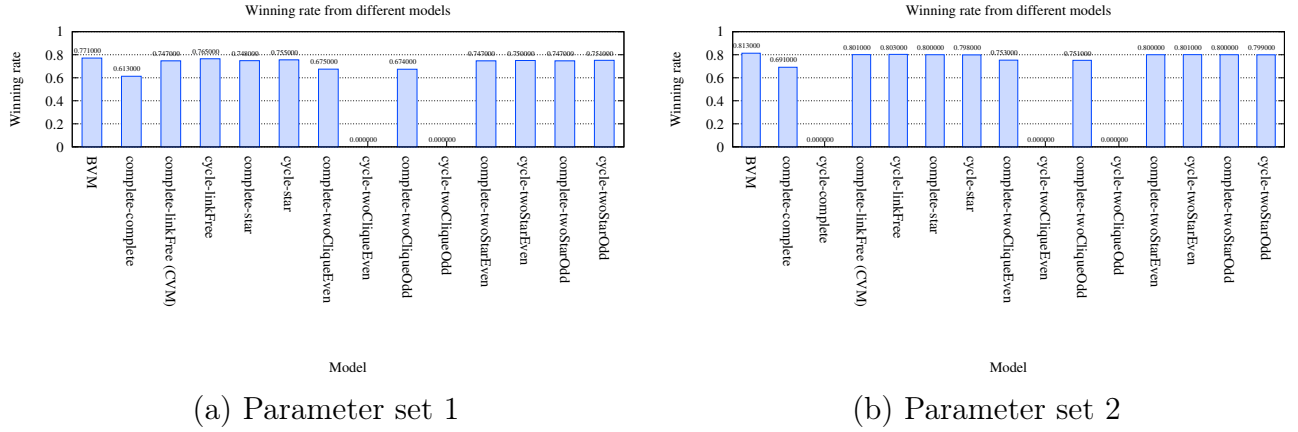


Figure 13: Winning rate of red opinion for all models

two-layer network structure to model the social network, the external layer represents the public communication network, and the internal layer represents the private communication network. If individuals show different opinions in the internal and external layers, they can change their opinions by externalization or internalization. We simulate the process of opinion transition for different network structures (i.e. complete, cycle, star, two-star-even, two-star-odd, two-clique-even, two-clique-odd for internal layer, and complete, cycle for external layer) with different parameters which represent different behaviors of individuals and compare the results with the basic voter model and concealed voter model. The main formulated conclusions based on the numerical simulations including some counter-intuitive ones are as follows:

1. For some simulations, a simple external layer network structure like cycle creates a problem for reaching a consensus;
2. If individuals in the social network are not good at expressing their opinions publicly (have a low value of parameter e), internal interaction does not have a great influence on consensus (including the winning rate and consensus time).

Chapter 2. Analysis of consensus time and winning rate in two-layer networks with hypocrisy of different structures

This chapter provides the definition for two-layer networks with replica nodes. We introduce the microscopic-GCVM which is more general than the macroscopic-GCVM discussed in Chapter 1 under the assumption that the internal interaction is allowed, the microscopic-GCVM is suitable for any given network structure that satisfies the definition. The impact of network properties and individual's behavior on the consensus time and winning rate is discussed. The results of this chapter is published in [25, 27].

2.1 Multi-layer network with replica nodes

A multilayer network is a network formed by several networks that evolve and interact with each other [56].

In a multilayer network with replica nodes there is a one-to-one mapping of the nodes in different layers and corresponding nodes are called replica nodes. Since there is a one-to-one mapping between the nodes in different layers, every layer is formed by the same number of nodes [56].

2.2 Two-layer network with replica nodes

We use the following notations to define a two-layer (external and internal) network with replica nodes:

1. N : number of nodes in each layer;

2. $a_i = (a_i^E, a_i^I)$: one-to-one mapping of node i in the external and internal layer, where a_i^E (a_i^I) is a representation of node i in the external (internal) layer;
3. $G_E(\mathcal{V}_E, \mathcal{E}_E)$: predefined external network, where $\mathcal{V}_E = \{a_i^E\}$ and \mathcal{E}_E represent a set of individuals and set of edges in the external layer;
4. $G_I(\mathcal{V}_I, \mathcal{E}_I)$: predefined internal network, where $\mathcal{V}_I = \{a_i^I\}$ and \mathcal{E}_I represent a set of individuals and set of edges in the internal layer;
5. $\mathcal{E}_C = \{(a_i^E, a_i^I) | i = 1, \dots, N\}$: edges between replica nodes.

A two-layer network with N individuals/agents can be defined as¹:

$$G(\mathcal{V}, \mathcal{E}), \tag{7}$$

where $\mathcal{V} = \mathcal{V}_E \cup \mathcal{V}_I$, $|\mathcal{V}_E| = |\mathcal{V}_I| = N$, and $\mathcal{E} = \mathcal{E}_E \cup \mathcal{E}_I \cup \mathcal{E}_C$. This definition is independent of a specific network structure, i.e. external/internal networks can be different.

2.3 Model

2.3.1 The general concealed voter model (macro version)

Zhao and Parilina [24] proposed GCVM based on CVM introduced in [21]. These papers use simulations to represent opinion transmission processes in two-layer networks. In the following section, we introduce GCVM in a micro version.

¹This definition and the corresponding opinion dynamic models was firstly introduced in [27] and further discussed in [25] and [26].

2.3.2 The general concealed voter model (micro version)

In the GCVM, we use R, B (r, b) to represent individuals' external (internal) red and blue opinions respectively. There is a list of notations:

- $S = \{Rr, Rb, Br, Bb\}$: set of all possible states of an individual;
- $\omega(a_i, t) \in S$: opinion of individual a_i at time t , where $i = 1, \dots, N$, and $t = 0, 1, \dots$;
- ρ_{r_e} : ratio of individuals having red opinion in external layer;
- ρ_{r_i} : ratio of individuals having red opinion in internal layer;
- ρ_r : ratio of individuals having red opinion in both internal and external layers;
- r_e : number of individuals having red opinion in external layer;
- r_i : number of individuals having red opinion in internal layer;
- r : number of individuals having red opinion in both internal and external layers;
- π_{c_e} : external copy rate, that is a probability of an individual to copy opinion of his/her external neighbor;
- π_{c_i} : internal copy rate, that is a probability of an individual to copy opinion of his/her internal neighbor;
- π_e : externalization rate, that is a probability of hypocrisy² choosing to publicly express his/her internal opinion;
- π_i : internalization rate, that is a probability of hypocrisy accepting his/her external opinion.

In Sections 2.3.2–2.3.2, we describe GCVM of opinion dynamics in a two-layer network. The description is organized so that to understand how numerical simulations presented in Section 2.4 are done.

²By hypocrisy we mean a node having different opinions in external and internal layers, i.e., the nodes in states Rb and Br .

Two-layer network structure initialization

We start by setting two networks G_E and G_I (we read these networks from the file, and an example of such a file representing external cycle and internal star structures is shown in Listing 1. Then we add the edges between external and internal representations of individuals).

Listing 1: Example of graph file with external cycle and internal star structure

```

>>>external
E0 E1
E1 E2
E2 E3
E3 E0
<<<external end
>>>internal
I00 I1
I00 I2
I00 I3
<<<internal end

```

This results in a two-layer network G we store as an adjacency list.

Initialization of individuals' initial states

The relationship between the number of individuals in each state is presented in equation (3) of Chapter 1. Assuming a uniform distribution for each agent to belong to any state $s \in S$ at the initial time, we adopt the following rule of setting the initial state $\omega(a_i, 0)$ for any agent a_i at time $t = 0$:

$$\omega(a_i, 0) := f(x) = \begin{cases} Rr, & 0 \leq x < \rho_r, \\ Rb, & \rho_r \leq x < \rho_{r_e}, \\ Br, & \rho_{r_e} \leq x < \rho_{r_e} + \rho_{r_i} - \rho_r, \\ Bb, & \rho_{r_e} + \rho_{r_i} - \rho_r \leq x \leq 1, \end{cases} \quad (8)$$

where $x \sim U(0, 1)$.

Opinion transmission process

We can divide individuals into hypocrites and non-hypocrites based on the consistency of their external and internal opinions. Hypocrites are individuals who

have different opinions in the internal and external layers, while non-hypocrites have the same opinions in both layers.

We focus on two measurements to analyze GCVM:

- *consensus time*: T_{cons} is consensus time in (G)CVM, that is, the time required for all individuals to form the same opinion in internal and external layers (i.e., $\rho_{r_e} = \rho_{r_i} = \rho_r = 0$ or 1 for T_{cons});
- *winning rate*: ρ is a winning rate of red opinion in a series of simulations. For the opinion, to win means that there is no other opinion that agents have in the whole network (i.e. in a series of simulations, the number of simulations, in which red opinion wins blue opinion divided by the number of simulations).

Before presenting an algorithm of GCVM, we briefly define the actions available for a randomly chosen individual a_i :

- *picking up a_i 's neighbor*: Randomly choose a neighbor among all a_i 's neighbors. Let it be individual a_j (this is a prerequisite action for external/internal copying);
- *external copying*: a_i copies a_j 's external opinion with probability π_{c_e} ;
- *internal copying*: a_i copies a_j 's internal opinion with probability π_{c_i} ;
- *externalization*: a_i expresses his/her internal opinion with probability π_e (this action is available only for hypocrite);
- *internalization*: a_i accepts his/her external opinion with probability π_i (this action is available only for hypocrite).

Externalization and internalization are meaningless for non-hypocrites, so they have only two possible actions (external and internal copies).

Algorithm of GCVM:

Step 1. Initialize $t = 0$.

Step 2. Choose an individual a_i , uniformly random from N individuals in two-layer network G ;

Step 3. Check all valid actions of individual a_i (depending on his/her state) and randomly choose one of the valid actions with equal probabilities:

- I) a_i is a hypocrite, then he/she has four possible actions: (i) external copying, (ii) internal copying, (iii) externalization, and (iv) internalization. Any action is chosen with a probability of 0.25;
- II) a_i is a non-hypocrite, then he/she can perform only external or internal copying. Any action is chosen with a probability of 0.5.

- Step 4. Generate random number $x \sim U(0, 1)$. Perform the action chosen in Step 3:
- a) if external copying is chosen in Step 3 and $x < \pi_{c_e}$, then a_i copies a_j 's external opinion;
 - b) if internal copying is chosen in Step 3 and $x < \pi_{c_i}$, then a_i copies a_j 's internal opinion;
 - c) if externalization is chosen in Step 3 and $x < \pi_e$, then a_i expresses his/her internal opinion;
 - d) if internalization is chosen in Step 3 and $x < \pi_i$, then a_i accepts his/her external opinion.

- Step 5. Increase t by 1. If consensus is reached³, stop iteration. Otherwise, go back to Step 2.

2.4 Experiments and results

2.4.1 General description

The experiment focuses on observing the effect of an external network structure on a winning rate of opinion and consensus time.

We consider two types of external structures: cycle and complete, and seven internal structures: cycle, complete, star, two-star (odd and even cases), and two-clique (odd and even cases). This gives us 14 different combinations of external-

³The algorithm will be stopped when all individuals in both layers hold the same opinion, i.e. consensus is reached.

internal structures, i.e. 14 two-layer networks. Most of the structures are shown in Figure 3 to Figure 9 of the Chapter 1, we do not present here to save space.

For our experiments, we fix the following parameters: $\rho_{r_e} = 0.75$, $\rho_{r_i} = 0.25$, $\rho_r = 0.2$, $\pi_{c_i} = \pi_{c_e} = 1$, $\pi_e = 0.01$, $\pi_i = 0.5$, $N = 100$. Then we observe the effect of external structure on the winning rate of opinion and consensus time (number of iterations) for the given internal structure. We conduct 100 simulations for each model and obtain the following statistical results. The models presented in Figures 14a and 14b below are named as: “external layer — internal layer — # of individuals”.

As shown⁴ in Figure 14a, we conclude that cyclic external structures prolong consensus time in comparison with complete external structure (the similar result is obtained in [24]). This conclusion is true for all internal structures we examine in the experiment. Additionally, we observe that the internal structure also has an impact on consensus time. For instance, consensus time for a two-star internal structure is less than that of a two-clique internal structure.

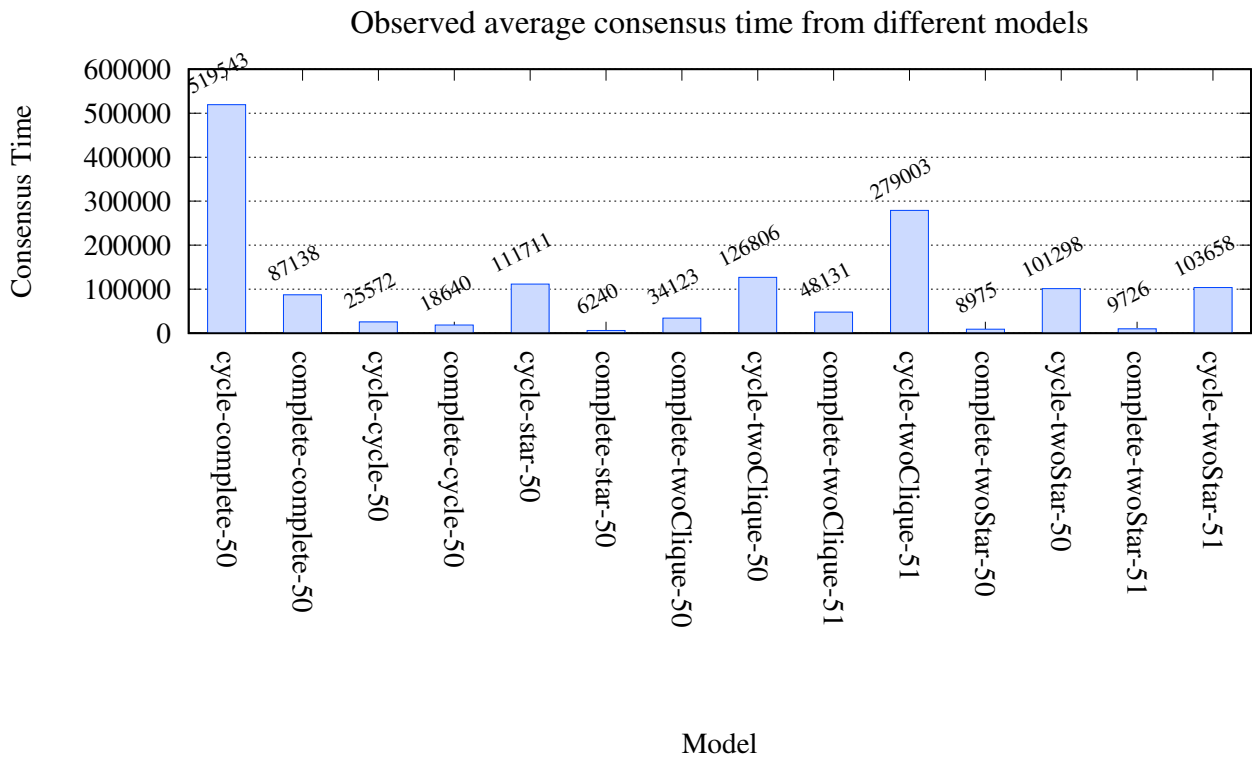
However, the observation results for the winning rate are quite different than in our previous work [24], as shown in Figure 14b. We can notice that except internal structures “twoStar-51” and “star-50”, a cyclic external layer decreases the winning rate. For all other models, a cyclic external structure has a positive impact on the winning rate. The possible reason is in specification of a microscopic model, i.e. actions that an individual/agent can take are related only to his current state.⁵ In a macroscopic version of GCVM, the probability of each possible action is related to the overall state of the system represented by a triple $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ (see [24]).

2.4.2 Main results and observations

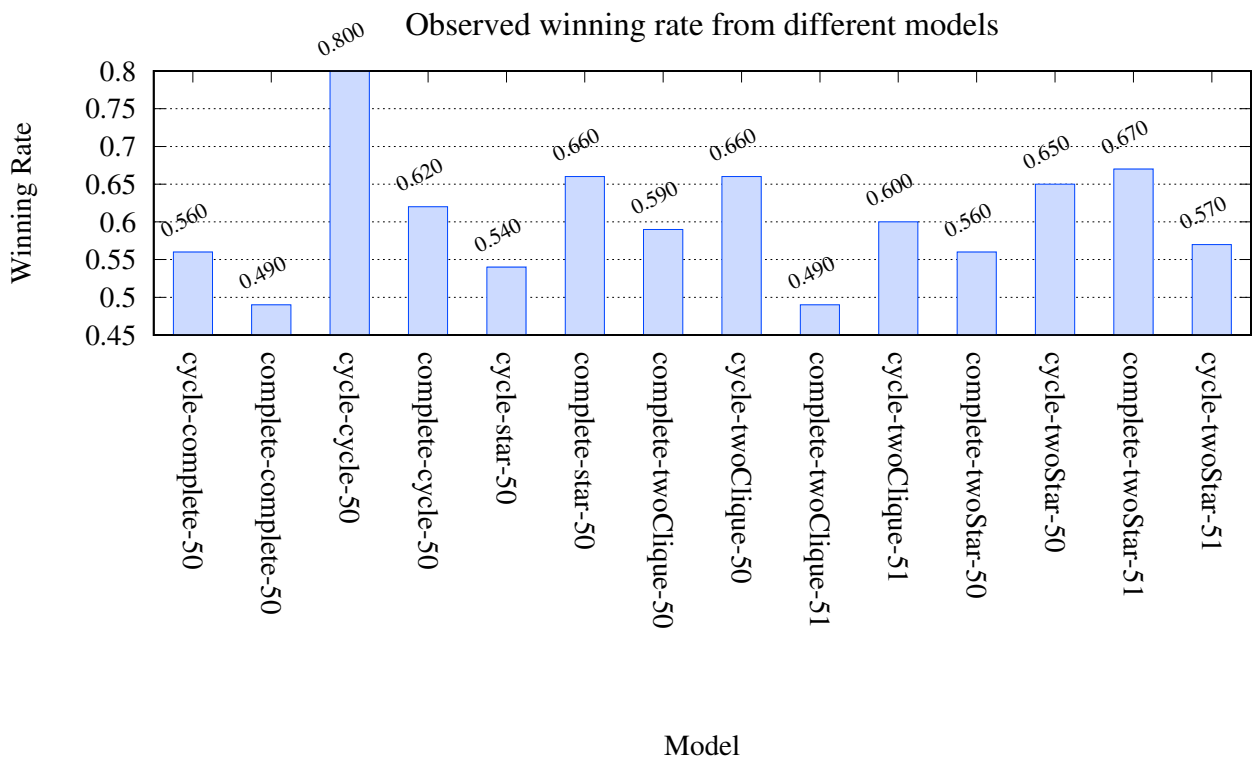
Based on the findings from Section 2.4.1 indicating that a cyclic structure has a positive impact on consensus time, we have formulated the following research questions. A series of experiments were designed and conducted in order to address these research questions:

⁴The number in the model name represents the number of individuals. e.g. 50 represents an even case, and 51 is an odd case.

⁵For non-hypocrites, an individual has two possible actions, and for hypocrites, an individual has four possible actions. The probability for each possible action of an individual/agent at the present moment is fixed.



(a) consensus time



(b) winning rate

Figure 14: Observed consensus time (a) and winning rate (b) for models with different two-layer network structures

1. How does an external structure influence KPIs, i.e. are there any features of a network that significantly affect KPIs? To address these question we do the following:
 - a) extend a cyclic structure to a complete one in different ways,⁶
 - b) reduce a cyclic structure to a line, and observe the KPIs,
 - c) conduct statistical tests to determine if there are significant differences in the distribution of KPIs for different ways of constructing a complete graph (see Item a),
 - d) check the correlation between network features and KPIs.
2. How do externalization and internalization rates influence KPIs?
 - a) vary π_e from 0.1 to 1.0 with a step of 0.1,
 - b) vary π_i from 0.3 to 1.0 with a step of 0.1 (note: when $\pi_i = 0.1$ or $\pi_i = 0.2$, consensus cannot be reached in some models).
3. How does a copying rate influence KPIs?
 - a) vary π_c from 0.1 to 1.0 for both layers with a step of 0.1.
4. How does a combination of parameters influence KPIs (i.e. which combinations maximize or minimize consensus time and winning rate)?
 - a) use a combination of parameters $(\pi_i, \pi_e, \pi_{c_i}, \pi_{c_e})$, where we vary $\pi_i \in [0.3, 1]$ and $\pi_e, \pi_{c_i}, \pi_{c_e} \in [0.1, 1]$ with a step of 0.1.

We start by extending a cyclic structure in the following three different ways:

- *normal*: find a set of edges presented in a complete graph but not in a cyclic graph, and add them sequentially to a cyclic graph until finally obtain a complete graph;
- *random*: find a set of edges presented in a complete graph but not in a cyclic graph, and add them randomly to a cyclic graph;
- *shortest*: find a set of edges presented in a complete graph but not in a cyclic graph, and add the edges from the list that minimizes d , where d is the average

⁶A cycle graph can be transformed into a complete graph by adding several nonexistent edges, or degenerate into a line by deleting an existing edge.

of the shortest paths among all pairs of nodes in external layer. Here d is calculated as follows:

$$d = \sum_{s,t \in \mathcal{V}_E} \frac{d(s,t)}{n_E(n_E - 1)}, \quad (9)$$

where $d(s,t)$ is the length of the shortest path between s and t , \mathcal{V}_E is a set of nodes in external layer, $n_E = |\mathcal{V}_E|$ is the number of nodes in external layer.

Obviously, for an undirected graph with N nodes, a cyclic structure has only N edges, and a complete structure has $N \times (N - 1)/2$ edges. Therefore, a cyclic graph can become a complete graph by adding $N \times (N - 3)/2$ edges. If we remove one edge from a cyclic structure, it will degenerate to a line.

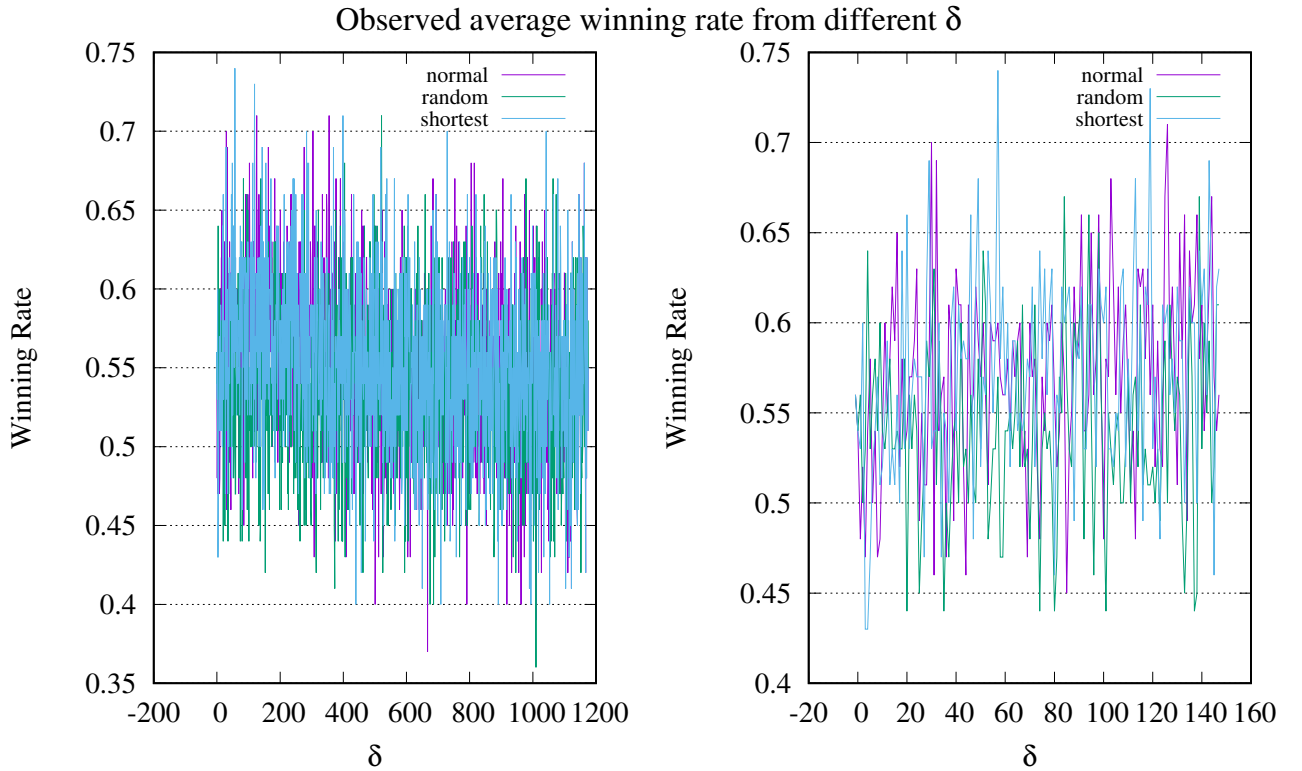
In our experiments, we extended a network structure from ‘cycle-complete-50’ to ‘complete-complete-50’, defined ‘cycle+ δ -complete-50’ as an intermediate network structure, where $\delta \in \mathbb{Z}$ means the number of edges which we have added to a cyclic structure by an iteration. Here $\delta \in [-1, 1175]$, and the value $\delta = -1$ corresponds to the case when we deleted an edge from a cycle degenerating this cycle into a line. When $\delta = 1175$, the cycle becomes a complete graph, i.e. the maximal number of edges that can be added into a cycle with 50 nodes is equal to 1175. In our experiments we examined the dynamics of consensus time and d when δ is increasing.

The results of simulations are shown in Figure 15. In particular, Figure 15a shows how d (δ) influences winning rate for different ways of extending a cyclic structure to a complete one. The right figure is an increase of the first 150 points from the left graph, we did the same in Figures 15b and 16. Looking at Figure 15a, we can notice that winning rate is almost white noise with changes of δ .⁷

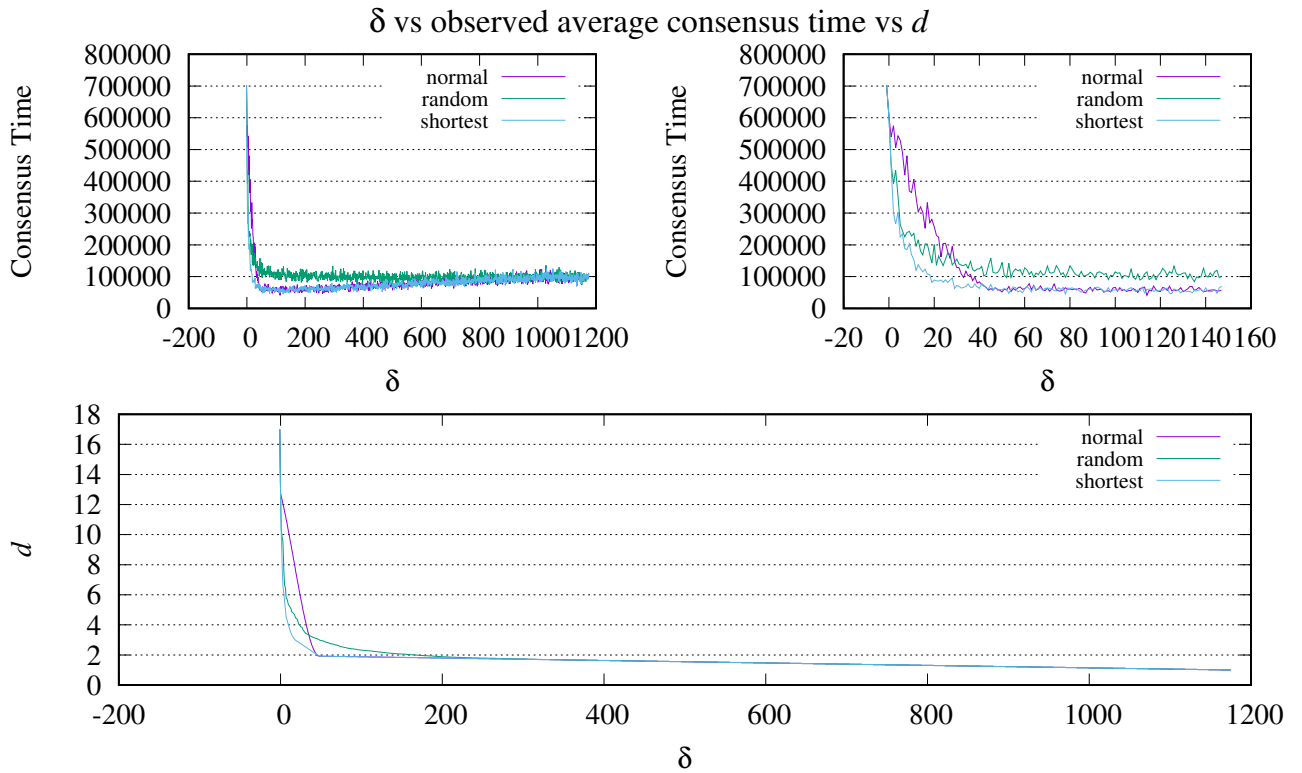
Figure 15b shows a trend of consensus time when δ is changing, and it is easy to recognize that the purple and green lines have almost the same trend. At the beginning, with an increase of the number of added edges, consensus time significantly decreases. After reaching a certain level, the increase in the number of edges has no significant impact on consensus time. First points in Figure 15b correspond to a network structure “cycle+-1-complete-50” having an external line structure. It is obvious that $d_{\text{line}} > d_{\text{cycle}} > d_{\text{complete}}$. Therefore, we formulate Hypothesis 1:

Hypothesis 1. *There is a significant correlation between consensus time and d .*

⁷The winning rate is white noise for “random” with lag 1–10, for “shortest” with lag 1–3, but the winning rate for “normal” extending way is not white noise which is confirmed by Ljung–Box test.



(a) Winning rate



(b) Consensus time

Figure 15: Winning rate (a) and consensus time (b) for the models with different extension ways

In order to verify Hypothesis 1, we should find d for each graph in Figure 15b. After calculating d for each graph, we construct the third subfigure in Figure 15b. Obviously, the first and third subfigures have a similar trend.

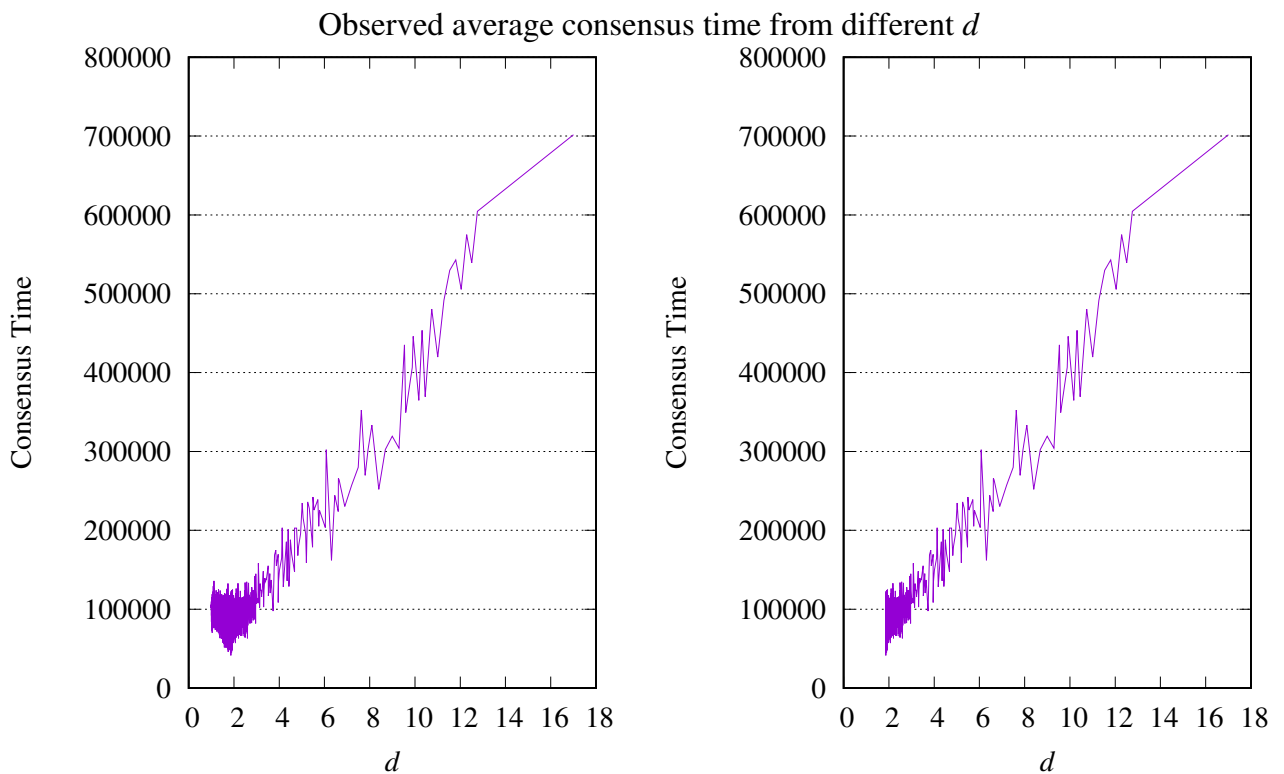
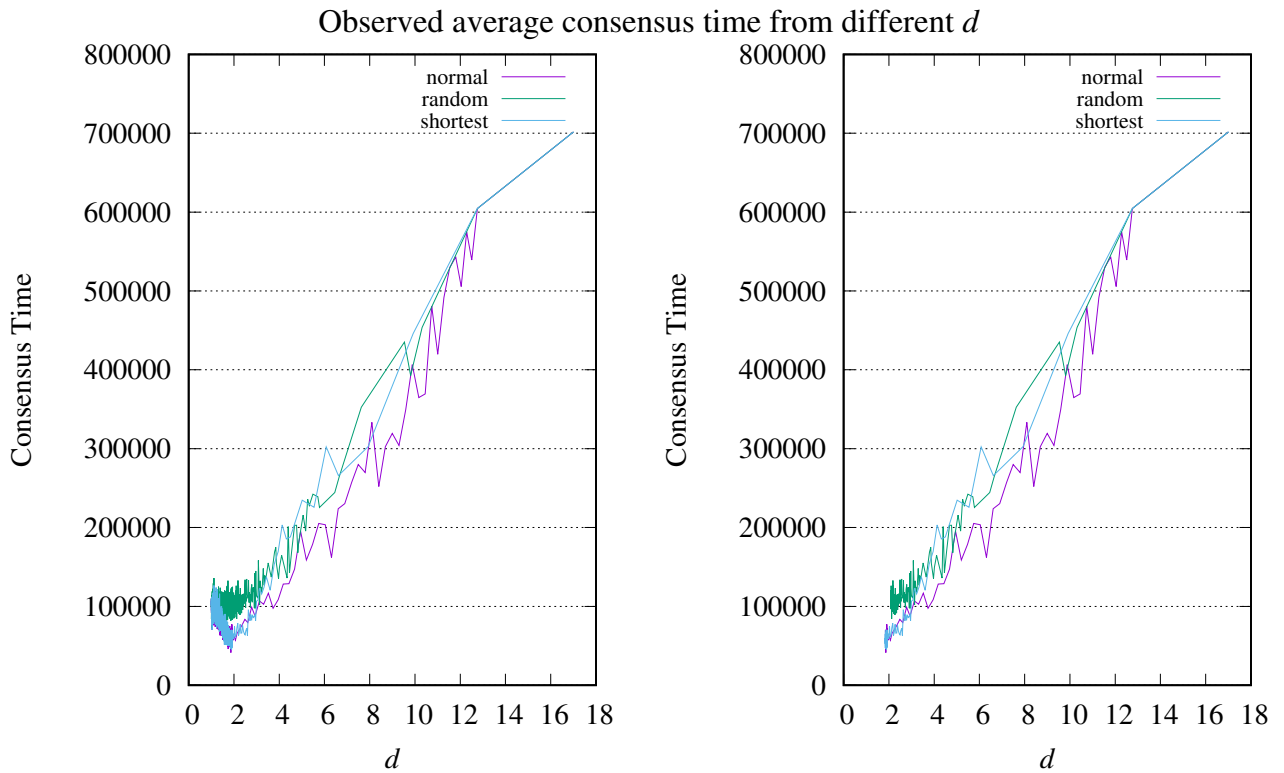


Figure 16: Consensus time vs d

We represent the relation between d and consensus time in Figure 16. The difference between Figures 16a and 16b is that in Figure 16a we group the data by

extending way first, and then draw the trend, while in Figure 16b, we do not specify the way of extending the graph, but only make an analysis based on different d and consensus time.

We can observe an approximately linear relationship in Figure 16. We can further use statistical methods for correlation analysis [57, 58]. The results of examining the Pearson correlation coefficient (PCC) [59] are shown in Figure 17. We make the following conclusions:

- the correlation between d and consensus time is significantly strong, and PCC is 0.78;
- for other pairs of KPIs, the correlation is not significant, and the absolute values of PCC are less than 0.15.

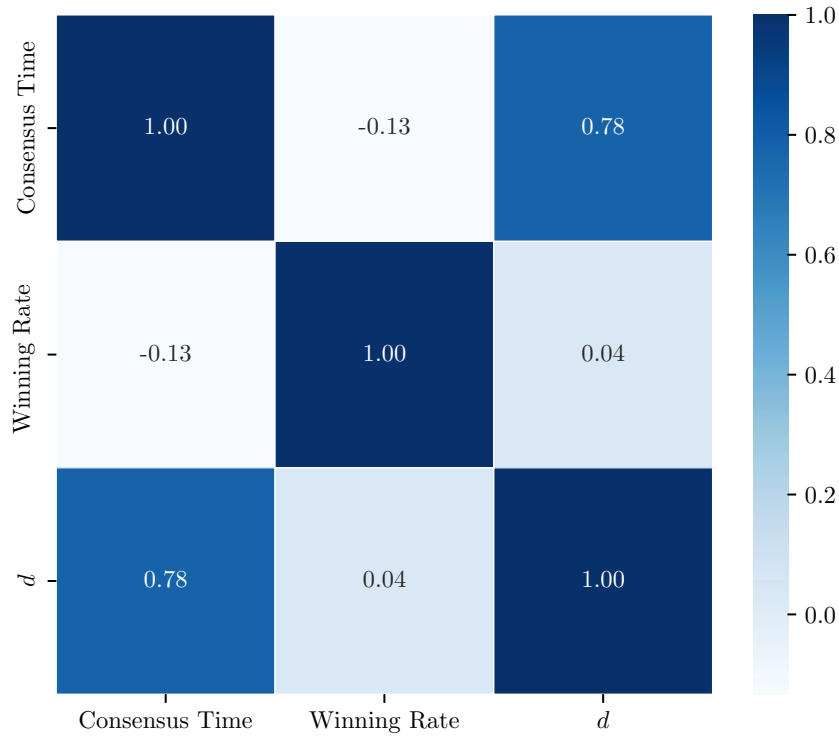


Figure 17: Pearson correlation coefficients

Hypothesis 2. *There are significant differences in KPI distributions for different ways of constructing a complete graph.*

We are interested in how a different extending way influences the distribution of KPIs. We show empirical distributions of KPIs in Figure 18. We should notice

that their distributions are significantly different for different ways of extending the graph from a circle to a complete one. But for ‘normal’ and ‘shortest’ extending ways, KPIs distributions are very similar. We use the Kolmogorov–Smirnov test for further analysis [60,61]. The results are shown in Table 8. From Table 8 we can see that p -values for all KPIs when we compare normal and shortest extending ways are larger than 0.05. We make the following conclusion: we should accept the null hypothesis that the distributions of KPIs for normal and shortest extending ways are identical.

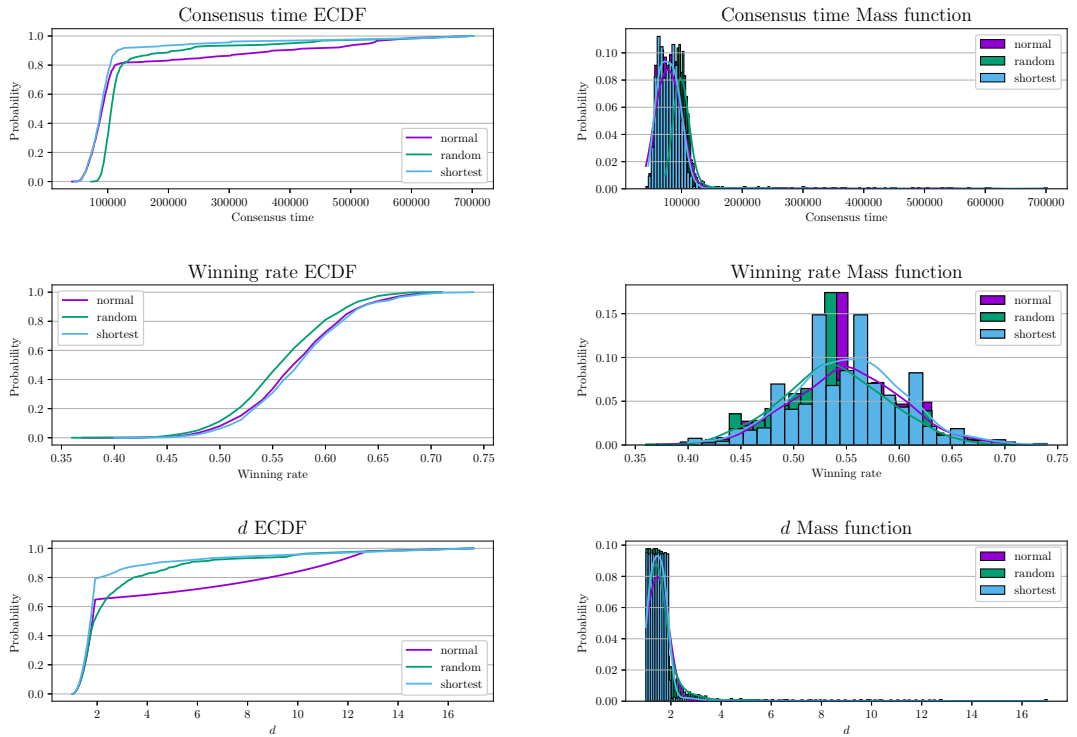


Figure 18: Frequency mass function and empirical cumulative distribution function (ECDF) for different KPIs

For now we find out that some ways of extending a circle to a complete graph have an impact on KPI distributions. But how significant is this impact? How are mean and variance affected? Therefore, we formulate next hypothesis:

Hypothesis 3. *Means and variances of KPIs are the same for different ways of extending a circle to a complete graph.*

We use some statistical tests to verify equity of variances and equity of means. Before doing this, we first run normality tests [62,63] since some statistical tests are

Table 8: Results of Kolmogorov–Smirnov tests

Pair	Statistics	<i>p</i>-values
Consensus time		
normal vs random	0.557	0.0
normal vs shortest	0.031	0.641
random vs shortest	0.574	0.0
Winning rate		
normal vs random	0.114	0.0
normal vs shortest	0.02	0.967
random vs shortest	0.121	0.0
<i>d</i>		
normal vs random	0.121	0.0
normal vs shortest	0.02	0.978
random vs shortest	0.12	0.0

parametric, i.e. they assume normality of the data. The results of normality tests are shown in Table 9, where we can see that the *p*-value for all KPIs are smaller than 0.05. Then we should reject the null hypothesis that any KPI is normally distributed.

In Table 10 and 11, we have two group of results, ‘ev/em test for all’ corresponds to whether the variances/means of three extension ways are all equal. The ‘pairwise ev/em test’ corresponds to two-sample equity test of variances/means. As none of KPIs is normally distributed, we use the Levene test for variance equity [64–67].

We make the following conclusions from Table 10:

- the variance of winning rate is the same for all extension ways (all *p*-values in the Levene tests are greater than 0.05);
- we reject the null hypothesis that the variances of the consensus time are equal for all extension ways (all *p*-values in the Levene tests are less than 0.05);
- variances of *d* are equal for normal and random extension ways.

Since none of KPIs is normally distributed, and not all KPIs are homoscedastic, to verify equity of means we use the Kruskal test [68–71]. The results of the tests are given in Table 11 and we conclude the following:

Table 9: Results of normality tests
normality testing overall

normality testing overall			
	KPIs	Statistics	<i>p</i>-values
	Consensus time	0.5	0.0
	Winning rate	0.996	0.0
	<i>d</i>	0.396	0.0
normality testing grouped via mode			
Mode	KPIs	Statistics	<i>p</i>-values
normal	Consensus time	0.379	0.0
normal	Winning rate	0.996	0.003
normal	<i>d</i>	0.32	0.0
random	Consensus time	0.347	0.0
random	Winning rate	0.996	0.003
random	<i>d</i>	0.469	0.0
shortest	Consensus time	0.442	0.0
shortest	Winning rate	0.995	0.001
shortest	<i>d</i>	0.369	0.0

- the means of d are equal for all extension ways, i.e. a way of extending a circle to a complete graph does not affect the mean of d (all p -values in the Kruskal tests are greater than 0.05). For consensus time and winning rate, the means are not all equal, i.e. they differ by extension ways;
- we accept the null hypothesis that the means of consensus time (and winning rate) are equal for normal and shortest extension ways (all p -values in KruskalResult are greater than 0.05).

Figures 19 and 20 show how parameters (π_{c_e} and π_{c_i} in Figure 19, and π_e and π_i in Figure 20) influence winning rate. We can see that winning rate fluctuates within a certain range, but not too much. Therefore, we temporarily think that an impact of parameters on winning rate is limited.

Figures 21 and 22 show how consensus time varies with a change of parameters (π_{c_e} and π_{c_i} in Figure 21, and π_e and π_i in Figure 22). We make these interesting observations:

- an increase of external copying rate π_{c_e} has a negative effect on consensus time. The interpretation is as follows: when an individual in a society is more

Table 10: Results of variance equity tests

Test	Description	KPIs	Statistics	p -values
EV test for all				
Fligner test	Distribution free when populations are identical	Consensus time	195.249	0.0
		Winning rate	1.206	0.547
		d	23.341	0.0
Levene test	More robust for significantly non-normal population	Consensus time	12.533	0.0
		Winning rate	1.074	0.342
		d	5.441	0.004
Bartlett test	More depends on normal population	Consensus time	320.877	0.0
		Winning rate	3.584	0.167
		d	351.795	0.0
Pairwise EV test				
Levene test for pairs	normal vs random	Consensus time	20.967	0.0
	normal vs shortest	Consensus time	6.579	0.01
	random vs shortest	Consensus time	7.651	0.006
	normal vs random	Winning rate	1.572	0.21
	normal vs shortest	Winning rate	0.003	0.959
	random vs shortest	Winning rate	1.71	0.191
	normal vs random	d	0.187	0.665
	normal vs shortest	d	8.533	0.004
	random vs shortest	d	10.099	0.002

Table 11: Results of mean equity tests

Test	Description	KPIs	Statistics	<i>p</i> -value
EM test for all				
f_ oneway test	Independent sample; each sample is from a normally distributed population; homoscedasticity	Consensus time	99.143	0.0
		Winning rate	23.205	0.0
		<i>d</i>	4.795	0.008
Kruskal test	Sample size should > 5	Consensus time	1100.711	0.0
		Winning rate	46.919	0.0
		<i>d</i>	1.596	0.45
Alexander Govern test	Independent sample; each sample is from a normally distributed population; heteroscedasticity	Consensus time	261.282	0.0
		Winning rate	47.419	0.0
		<i>d</i>	12.243	0.002
Pairwise EM test				
Kruskal test for pairs	normal vs random	Consensus time	795.081	0.0
	normal vs shortest	Consensus time	0.775	0.379
	random vs shortest	Consensus time	854.985	0.0
	normal vs random	Winning rate	32.215	0.0
	normal vs shortest	Winning rate	0.229	0.632
	random vs shortest	Winning rate	37.928	0.0
	normal vs random	<i>d</i>	1.092	0.296
	normal vs shortest	<i>d</i>	0.001	0.972
	random vs shortest	<i>d</i>	1.301	0.254

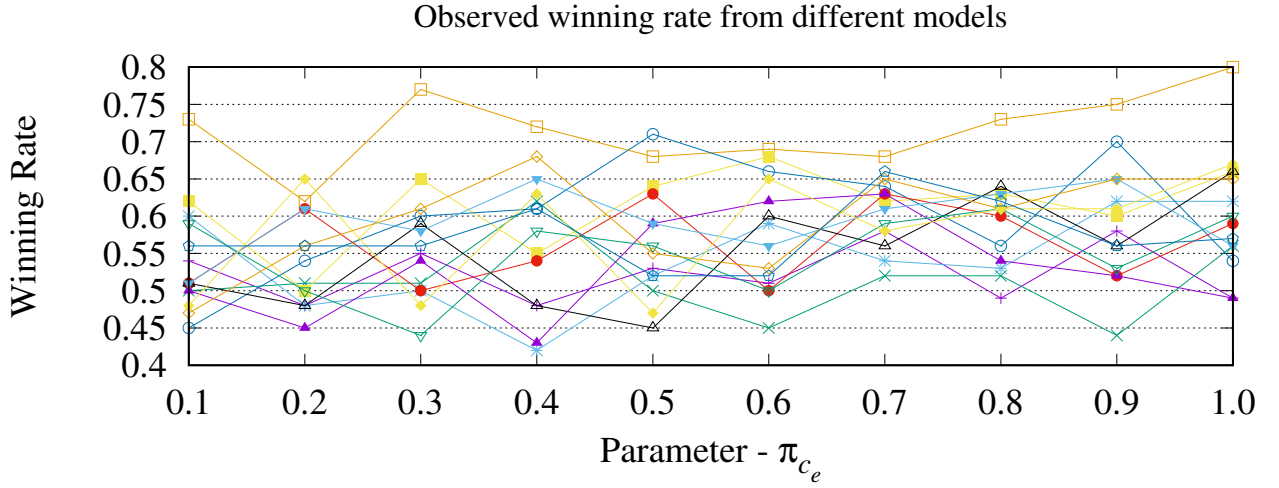
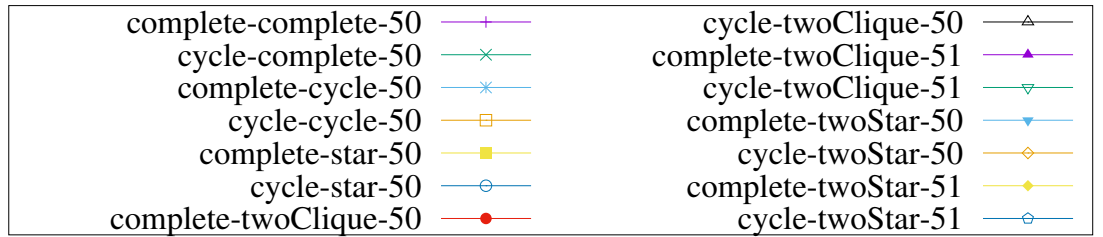
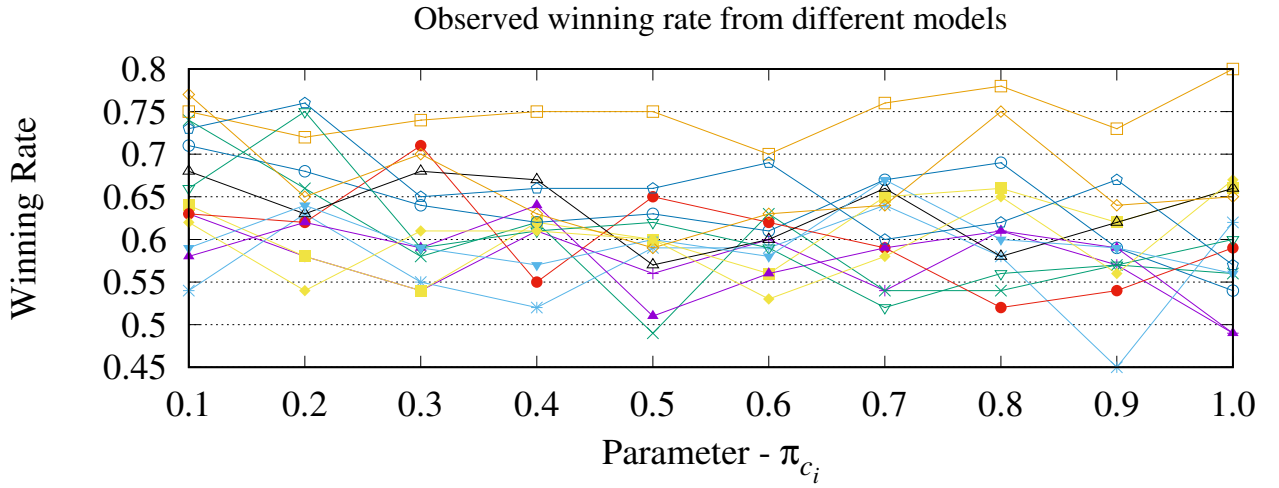
(a) Varying external copying rate π_{c_e} (b) Varying internal copying rate π_{c_i}

Figure 19: Winning rate for different copying rates

inclined to listen to the opinions of his/her external neighbors, it is helpful to reach consensus;

- with an increase of internal copying rate π_{c_i} , consensus time increases;
- with an increase of externalization rate π_e , consensus time first increases until it reaches the maximal value, and then decreases. The interpretation of this is as follows: expressing your true opinion to a certain extent is not effective

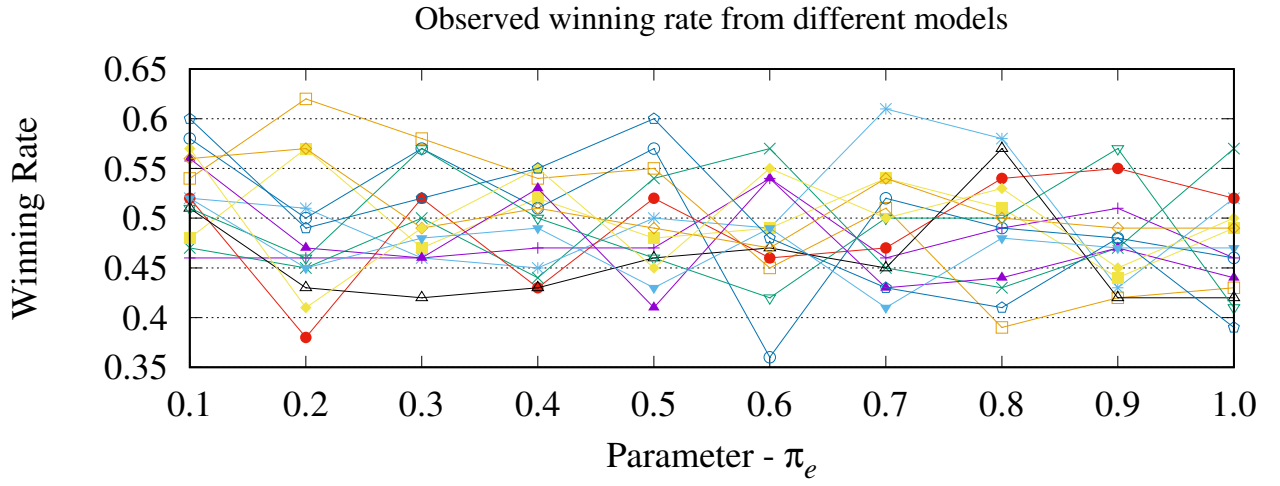
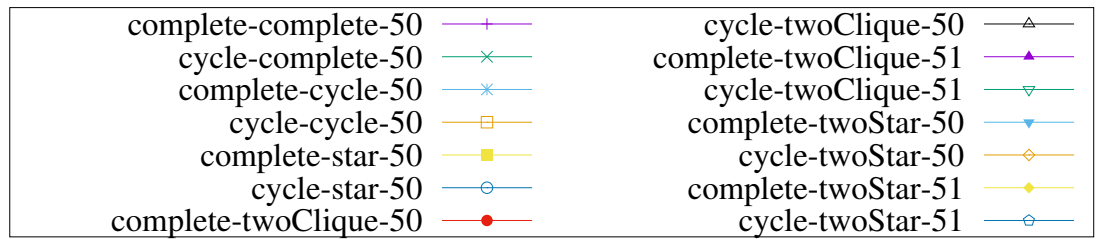
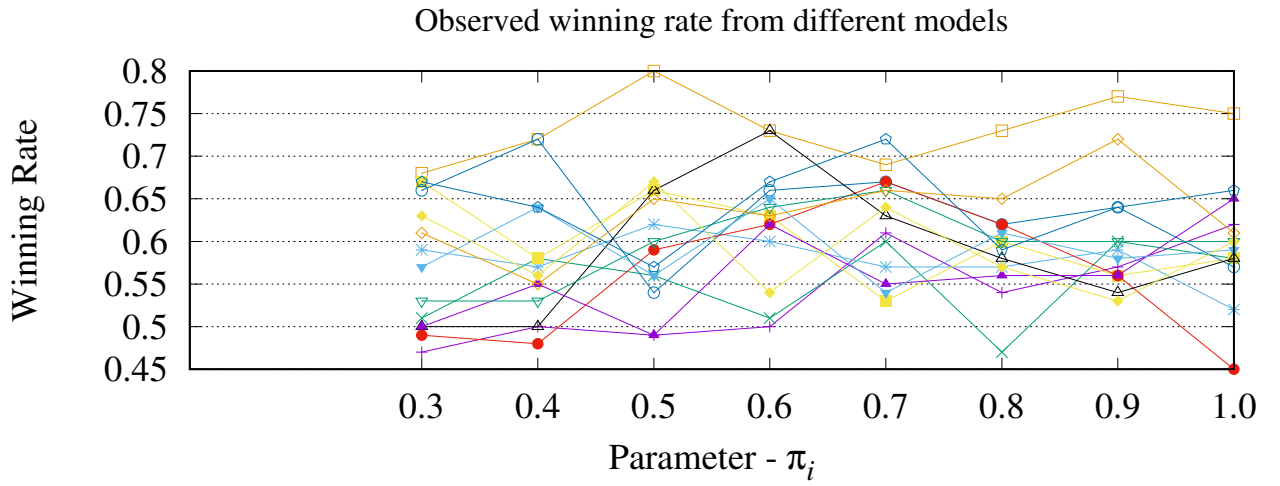
(a) Varying externalization rate π_e (b) Varying internalization rate π_i

Figure 20: Winning rate for different externalization and internalization rates

to reach consensus within the whole system, but beyond this threshold, along with an increase of desire to express your opinion, for the system, it is easier to reach consensus;

- with an increase of internalization rate π_i , consensus time decreases. This can be interpreted as follows: when people are more willing to accept their own external opinion, it will accelerate consensus of the whole system.

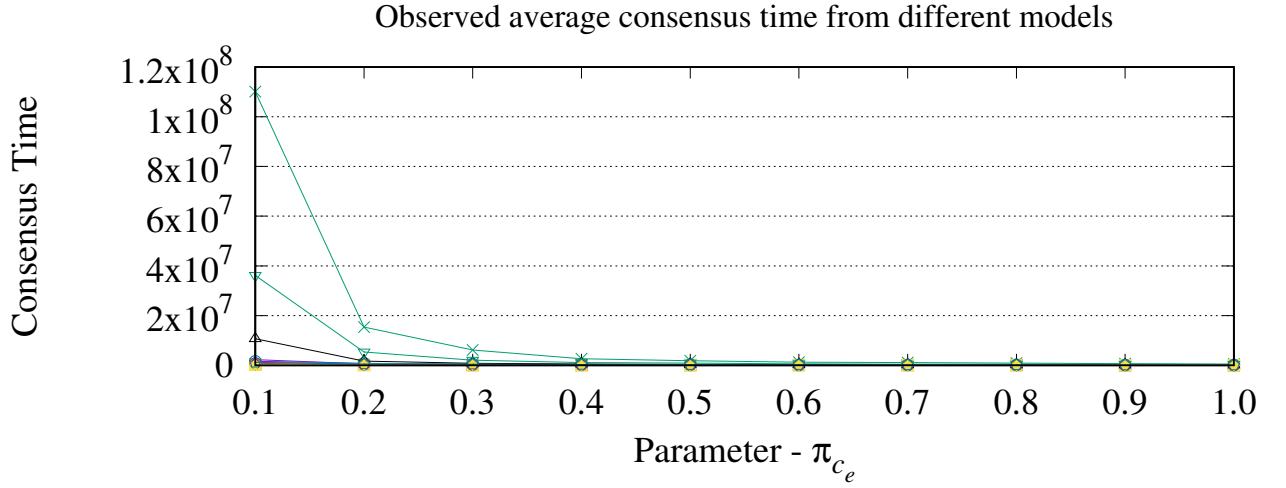
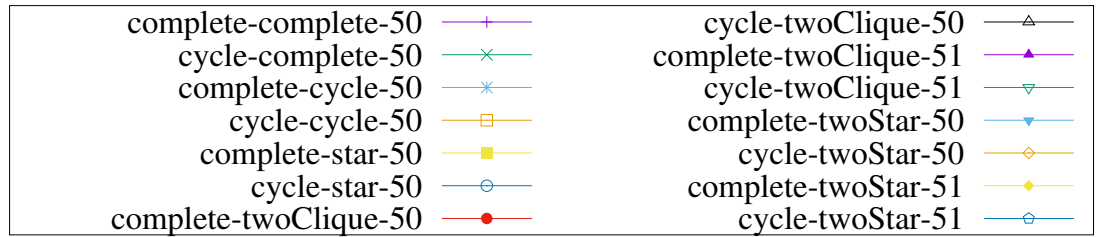
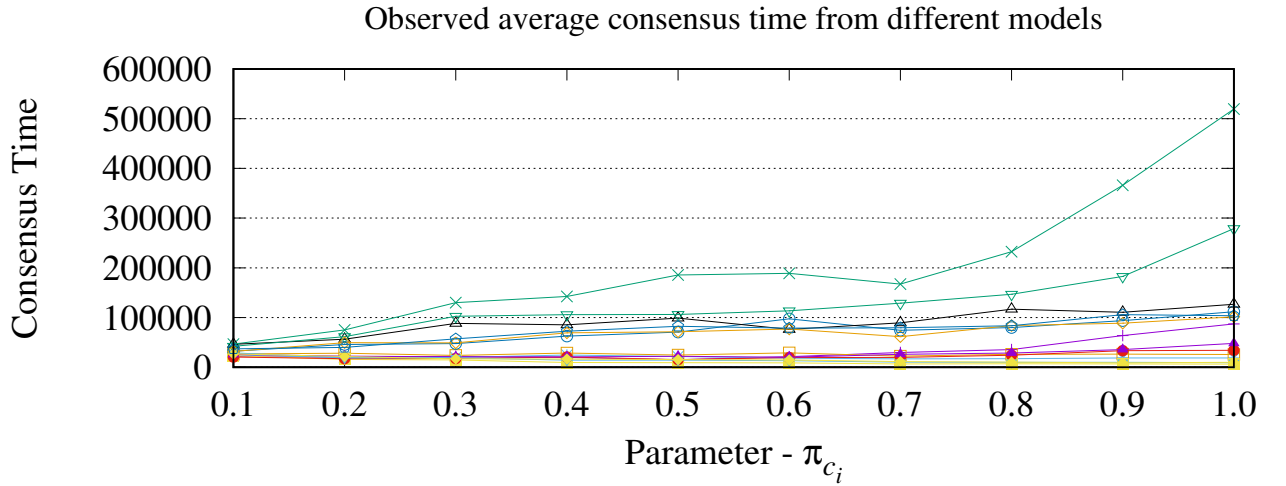
(a) Varying external copying rate π_{c_e} (b) Varying internal copying rate π_{c_i}

Figure 21: Consensus time for different copying rates

In Tables 12 and 13, we show minimal/maximal consensus time and winning rate respectively for different combinations of parameters. If we compare the left and right parts in Tables 12 and 13, there is a large difference between maximum and minimum values. We performed a clustering procedure using K -means method with input variables being consensus time and winning rate separately [72, 73]. The resulting cluster labels are then added to the original data. Based on these cluster

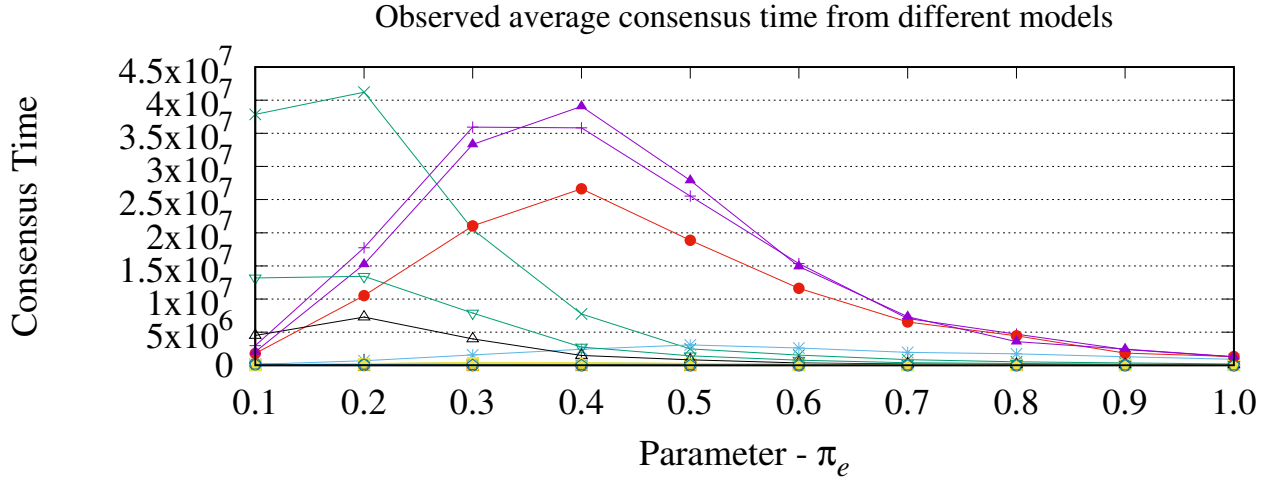
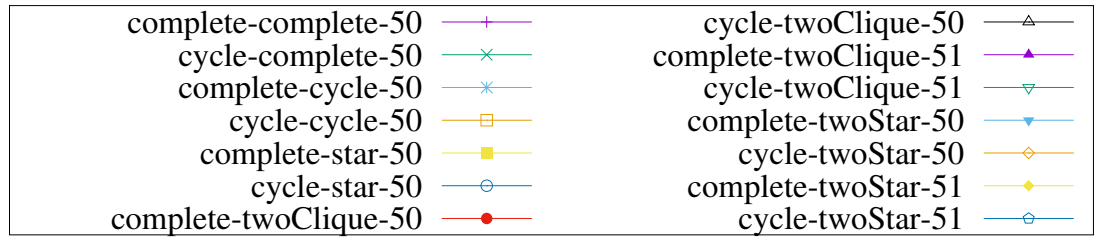
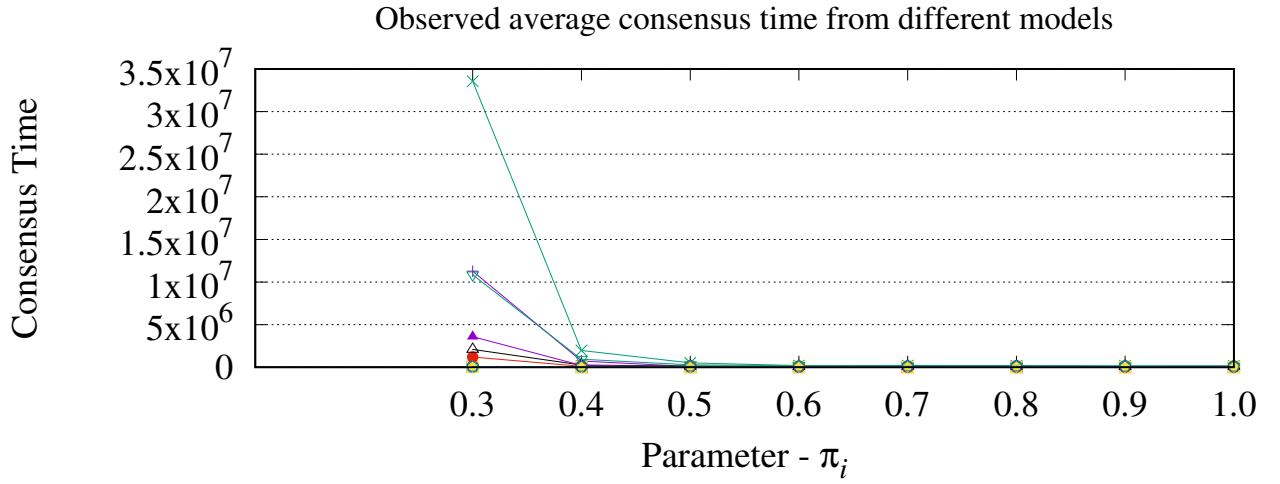
(a) Varying externalization rate π_e (b) Varying internalization rate π_i

Figure 22: Consensus time for different externalization and internalization rates

labels, we observed the distribution of the corresponding four parameters (π_{c_e} , π_{c_i} , π_e , and π_i).

Ideally, we prefer having two clusters since it allows us to determine which parameter combinations result in respectively large or small KPIs. In practice, we specify a range for the number of clusters k , from 1 to 20, and calculate the silhouette score [74]. In some sense an optimal value of k is the one maximizing the silhouette score.

Table 12: Observed minimum and maximum consensus time



































External	Internal	N	Max	π_{c_e}	π_{c_i}	π_e	π_i	Min	π_{c_e}	π_{c_i}	π_e	π_i
complete	complete	50	 1185261644	0.1	1.0	0.1	0.3	 28767	0.8	0.1	0.1	1.0
cycle	complete	50	 1144248878	0.6	1.0	0.1	0.3	 29387	0.9	0.1	0.3	1.0
complete	cycle	50	26467498	1.0	1.0	0.4	0.3	 22850	1.0	0.1	0.1	1.0
cycle	cycle	50	474773	0.1	0.1	0.6	0.9	 26842	1.0	0.9	0.1	1.0
complete	star	50	1602538	1.0	0.1	0.5	0.3	2018	0.1	1.0	1.0	0.3
cycle	star	50	275898	1.0	1.0	0.1	0.3	2021	0.1	1.0	1.0	0.3
complete	twoClique	50	 492816970	0.2	1.0	0.1	0.3	 28185	0.1	0.7	1.0	0.3
cycle	twoClique	50	151026217	0.7	1.0	0.1	0.3	 20923	0.1	1.0	1.0	0.3
complete	twoClique	51	 708728030	0.1	1.0	0.1	0.3	 30628	0.9	0.1	0.1	1.0
cycle	twoClique	51	 449651392	0.6	1.0	0.1	0.3	 26250	0.1	1.0	1.0	0.3
complete	twoStar	50	2027297	1.0	0.1	0.5	0.3	3795	0.1	0.9	1.0	0.3
cycle	twoStar	50	317571	1.0	0.9	0.1	0.3	3743	0.2	1.0	1.0	0.4
complete	twoStar	51	2347437	1.0	0.1	0.5	0.3	4201	0.1	0.9	1.0	0.3
cycle	twoStar	51	349638	0.7	1.0	0.1	0.3	4527	0.2	1.0	1.0	0.3

Table 13: Observed minimum and maximum winning rate

External	Internal	N	Max	π_{c_e}	π_{c_i}	π_e	π_i	Min	π_{c_e}	π_{c_i}	π_e	π_i
complete	complete	50	 0.7	0.1	0.1	0.2	1.0	 0.31	0.2	0.6	1.0	0.5
cycle	complete	50	 0.81	0.9	0.1	0.2	0.8	 0.33	0.4	0.1	0.8	0.3
complete	cycle	50	 0.71	0.2	0.1	0.4	1.0	 0.29	0.7	1.0	0.6	0.3
cycle	cycle	50	0.82	0.1	0.6	0.1	0.6	 0.25	0.2	0.9	1.0	0.6
complete	star	50	0.7	0.1	0.4	0.2	1.0	 0.24	0.3	0.4	1.0	0.4
cycle	star	50	0.82	0.2	0.1	0.1	0.7	 0.23	0.1	0.2	1.0	0.4
complete	twoClique	50	 0.68	0.5	0.7	0.4	0.8	 0.3	0.7	0.3	0.9	0.9
cycle	twoClique	50	 0.76	0.9	0.1	0.1	1.0	 0.28	0.2	0.8	0.6	0.5
complete	twoClique	51	0.73	0.1	0.1	0.1	0.9	 0.32	0.4	0.8	0.7	0.4
cycle	twoClique	51	0.77	0.7	0.2	0.1	0.9	 0.31	0.1	0.3	1.0	0.5
complete	twoStar	50	 0.72	0.1	0.4	0.2	1.0	 0.26	0.1	0.8	0.8	0.4
cycle	twoStar	50	0.77	0.8	0.2	0.2	0.9	 0.25	0.3	0.3	1.0	0.3
complete	twoStar	51	 0.72	0.1	0.1	0.1	0.7	 0.27	0.1	0.8	0.9	0.3
cycle	twoStar	51	0.8	0.8	0.2	0.1	1.0	 0.26	0.1	1.0	0.9	0.4

Since the distributions of winning rates are very close to a normal distribution, based on clustering results of winning rate, we cannot observe any significant differences in descriptive statistics of clusters (see Table 14). Therefore, in our future analysis we only focus on performing a clustering analysis of consensus time.

Table 14: Descriptive statistics of clusters within winning rate (complete-complete-50)

Parameters	Cluster	Count	Mean	Std	Min	25%	50%	75%	Max
π_{c_e}	0	169	0.484	0.282	0.1	0.3	0.4	0.7	1.0
	1	238	0.540	0.287	0.1	0.3	0.5	0.8	1.0
	2	186	0.573	0.274	0.1	0.3	0.6	0.8	1.0
	3	291	0.543	0.289	0.1	0.3	0.5	0.8	1.0
	:								
π_{c_i}	0	169	0.487	0.306	0.1	0.2	0.4	0.7	1.0
	1	238	0.591	0.293	0.1	0.3	0.6	0.9	1.0
	2	186	0.533	0.272	0.1	0.3	0.5	0.8	1.0
	3	291	0.547	0.290	0.1	0.3	0.5	0.8	1.0
	:								
π_e	0	169	0.509	0.290	0.1	0.3	0.5	0.7	1.0
	1	238	0.581	0.304	0.1	0.3	0.6	0.9	1.0
	2	186	0.525	0.292	0.1	0.3	0.5	0.8	1.0
	3	291	0.587	0.282	0.1	0.4	0.6	0.8	1.0
	:								
π_i	0	169	0.698	0.233	0.3	0.5	0.7	0.9	1.0
	1	238	0.632	0.230	0.3	0.4	0.6	0.8	1.0
	2	186	0.631	0.237	0.3	0.4	0.6	0.8	1.0
	3	291	0.622	0.230	0.3	0.4	0.6	0.8	1.0
	:								

Table 15 shows the number of elements in the clusters of consensus time, where we selected to have two clusters since it maximizes the silhouette score.

In Figure 23a, the distributions of consensus time are almost consistent for different multi-layer models. At the same time, in Figure 23b, we can clearly observe noticeable differences. This interesting result provides us with a valuable insight that the diverse parameter distributions can significantly prolong consensus time.

In the top right corner of Figure 23b, we observe that when the internal structure is complete or twoClique, parameter π_{c_i} differs significantly consensus time for these models in comparison with other models. In the lower left corner, we can see when the internal structure is complete or twoClique, a value of π_e in cluster 1

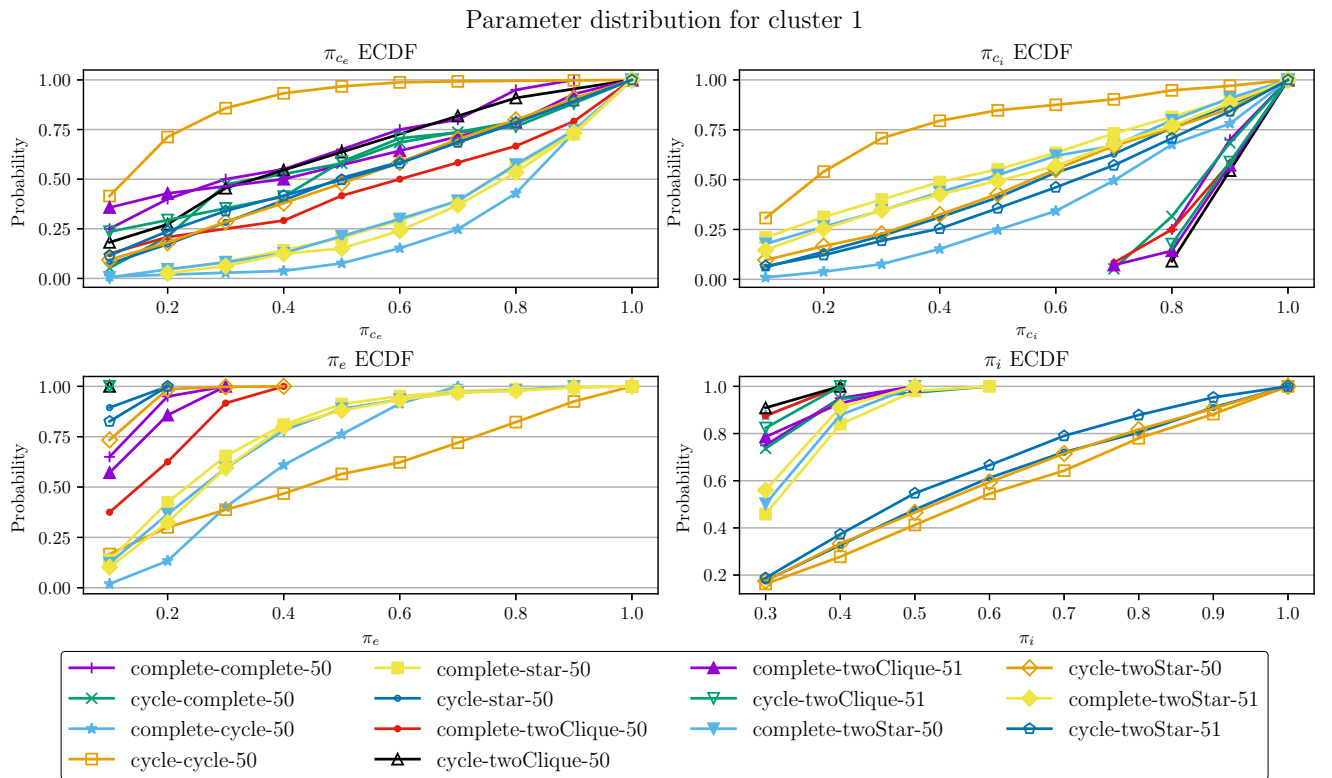
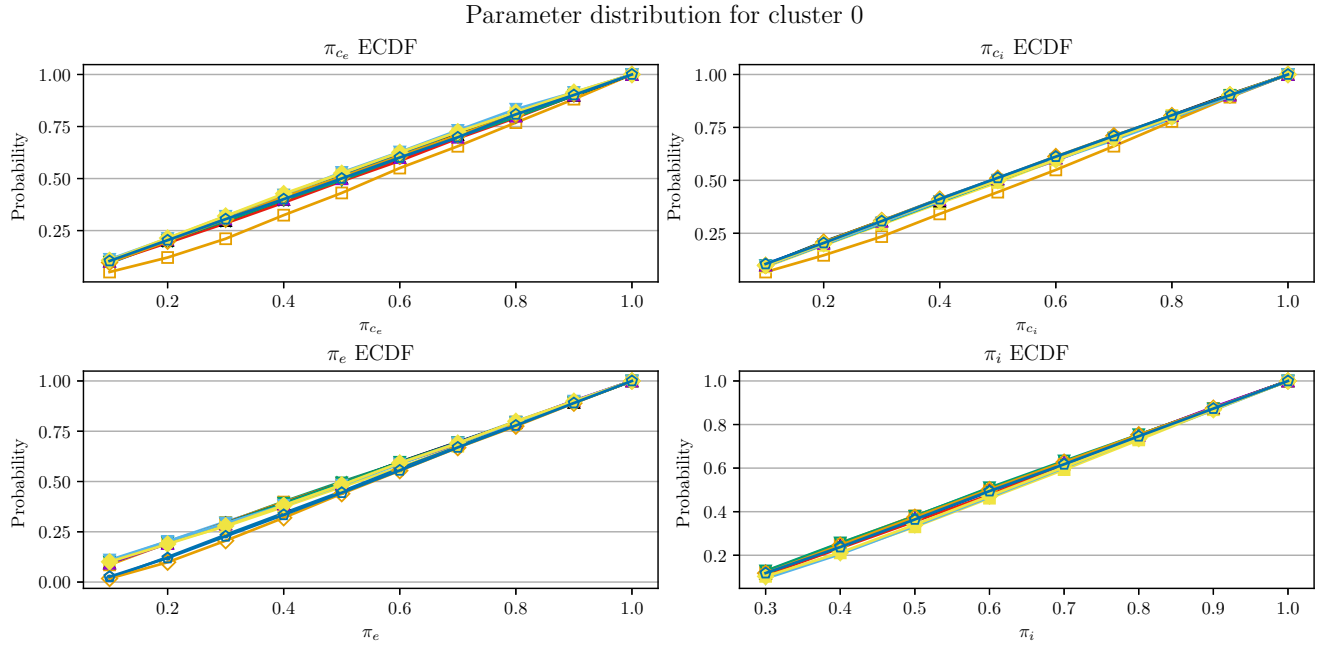


Figure 23: Empirical cumulative distribution function of consensus time with respect to parameters π_{c_e} , π_{c_i} , π_e , and π_i

Table 15: Consensus time cluster sizes for each model

	Model	Cluster 0	Cluster 1
0	complete-complete-50	3267	20
1	cycle-complete-50	3241	19
2	complete-cycle-50	3319	105
3	cycle-cycle-50	3027	400
4	complete-star-50	3219	205
5	cycle-star-50	3124	312
6	complete-twoClique-50	3239	24
7	cycle-twoClique-50	3348	11
8	complete-twoClique-51	3904	14
9	cycle-twoClique-51	3831	17
10	complete-twoStar-50	3171	197
11	cycle-twoStar-50	3100	372
12	complete-twoStar-51	3598	225
13	cycle-twoStar-51	3634	362

is always equal to 0.1. For the figures in the top left and lower right corners, we can get similar conclusions by comparison.

2.5 Conclusion to Chapter 2

This chapter introduces a novel approach, that is microscopic-GCVM, which simulate opinion dynamics process by creating a real network instead of using statistical-physical methods for its modeling as described in Chapter 1. Therefore, this approach is suitable for any two-layer networks that are represented by (7). Additionally, we use different ways to extend an external cyclic structure to a complete one. We highlight a hypothesis on how the way of extension of a circle to a complete graph influences consensus time and winning rate based on simulation results and use a statistical test to verify them. The main conclusions are as follows:

- cyclic external structure always increases consensus time;
- cyclic external structure has a positive impact on winning rate;

- cyclic external structure influences consensus time by an increase of d , i.e. there is a strong linear relationship between d and consensus time. The lower d is, the higher is consensus time;
- the way of extension of a circle to a complete graph has a significant impact on consensus time and winning rate;
- each parameter has a different impact on consensus time but almost has no impact on winning rate;
- the combination of parameters (different individual's behavior) has a significant impact on consensus time.

Chapter 3. Centrality measures and opinion dynamics in two-layer networks with replica nodes

In Chapters 1 and 2, we are separately introduced the macroscopic-GCVM and microscopic-GCVM. The network structures discussed in Chapters 1 and 2 are very simple structures – i.e. cyclic, complete, star, etc. In this chapter, we consider a Zachary’s karate club social network and extend it by adding the second (internal) layer of communication, and simulate the opinion dynamics on such two-layer networks based on microscopic-GCVM. The basic notations and concepts about microscopic-GCVM were introduced in Chapter 2. Therefore, we will not repeat them in this chapter.

A novel approach to simplify the two-layer network to one-layer weighted network is discussed. We also proposed two fast and accurate algorithms for one layer weighted network to approximate the game-theoretic centrality measures and examine connection between centrality measures and characteristics of opinion dynamic processes on such networks. The results of this chapter are published in papers [26, 51].

3.1 Two-layer network simplification

According to the definition of two-layer network in Chapter 2 (see equation (7)), two-layer network $G(\mathcal{V}, \mathcal{E})$ is composed of external network $G_E(\mathcal{V}_E, \mathcal{E}_E)$, internal network $G_I(\mathcal{V}_I, \mathcal{E}_I)$ with the set of edges \mathcal{E}_C connecting nodes between layers. Two-layer networks can also be represented by an adjacency matrix. The adjacency matrix of a two-layer network is a block matrix, where the diagonal blocks are the adjacency matrices of the external and internal layers, and the off-diagonal blocks are the adjacency matrices of the connections between external and internal layers.

The adjacency matrix of $G(\mathcal{V}, \mathcal{E})$ is as follows:

$$A = \begin{bmatrix} A_{EE} & A_{EI} \\ A_{IE} & A_{II} \end{bmatrix} \quad (10)$$

where A_{EE} is the adjacency matrix of the external layer, A_{EI} is the adjacency matrix of the connections between external and internal layers, A_{IE} is the adjacency matrix of the connections between internal and external layers, and A_{II} is the adjacency matrix of the internal layer. For undirected graphs in both layers, the adjacency matrix A is symmetric.

Keeping in mind an binary opinion dynamics model presented in the Chapter 1 and Chapter 2, we define the rates of coping opinions from one node to another:

- π_{c_e} : external copy rate with which an individual is copying opinion of his/her external neighbor if they both are randomly chosen;
- π_{c_i} : internal copy rate with which an individual is copying opinion of his/her internal neighbor;
- π_e : externalization rate with which an individual is behaving as a hypocrisy⁸ choosing to publicly express his/her internal opinion;
- π_i : internalization rate with which an individual being hypocrisy accepts his/her external opinion.

We propose a way how to transform a two-layer network $G(\mathcal{V}, \mathcal{E})$ with the given parameters $\pi_{c_e}, \pi_{c_i}, \pi_e, \pi_i$ of opinion dynamics to a one-layer weighted network. We can define a matrix of weights as follows:

$$W' = \pi_{c_e} \cdot A_{EE} + \pi_{c_i} \cdot A_{II} + \pi_i \cdot \Lambda_E + \pi_e \cdot \Lambda_I, \quad (11)$$

where Λ_E and Λ_I are $N \times N$ diagonal matrices, and the elements on the diagonal represent the degrees of nodes in external and internal layers, respectively. Furthermore, we use w'_{ij} to represent the elements of matrix W' and define a new weighted network $G'(\mathcal{V}', \mathcal{E}', W')$, where $\mathcal{V}' = \{1, 2, \dots, N\}$ is the set of nodes, $\mathcal{E}' = \{(i, j) \mid w'_{ij} \neq 0, i, j \in \mathcal{V}'\}$ is the set of edges, and W' is the matrix of weights.

Based on a new weighted network $G'(\mathcal{V}', \mathcal{E}', W')$, we proposed two game-theoretic centrality measures which will be discussed in Section 3.3.3.

⁸By hypocrisy we mean a node having different opinions in external and internal layers.

3.2 Zachary's karate club network in two-layer setting

As an example of a social network, we consider Zachary's karate club network representing friendship relations among 34 members of a karate club at the US university in the 1970s [75]. The study became famous in data and network analytical literature since it highlighted a conflict between manager (Node 0) and director (Node 33), which eventually led to the split of the club into two groups. One-layer Zachary's karate club network is represented in Fig. 24. The blue and red colors of nodes represent two opinions in the social network.

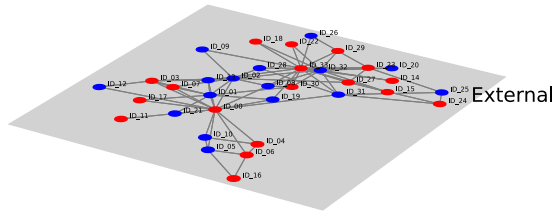


Figure 24: One-layer Zachary's karate club network

Fig. 25 shows how one-layer Zachary's Karate Club network can be extended into a two-layer network if we add an internal layer of communication between agents. If we consider binary opinion dynamics models, in the concealed voter model (CVM) [21, 22], the nodes in the internal layer are not connected, i.e. internal layer is represented by an empty graph (see Fig. 25a), while in the general concealed voter model (GCVM) [24, 25, 27] there may be nonempty network representing internal communication of agents. In Fig. 25b we represent a star internal structure. The colors in Fig 25 represent individuals' opinions.

The color, blue or red, is randomly initialized for the given parameters, these are (i) probability of having red initial opinion for the basic voter model on one-layer network, (ii) probabilities of having red initial opinion in external, internal, and in both layers for CVM and GCVM models.

In the basic voter model (BVM, see [4]), there is only one layer, as shown in Fig. 24, and each individual holds one of two opinions (red or blue). At each step, a random individual selects a random neighbor and adopts his/her neighbor's opinion with the copying rate π_{c_e} . This process repeats until everyone in the network holds the same opinion, i.e. reaching a consensus. The opinion dynamics process in

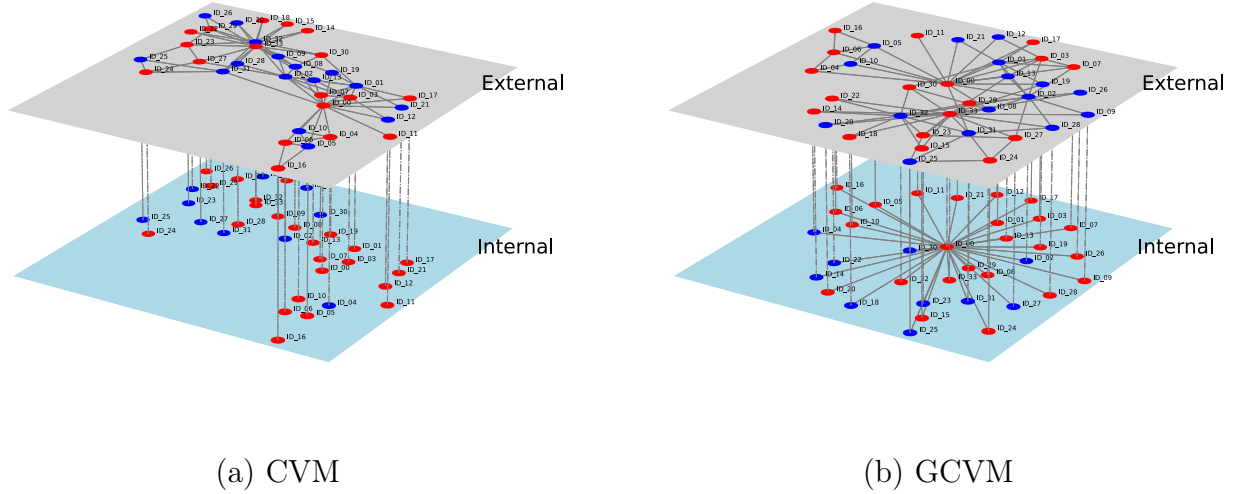


Figure 25: Two-layer networks used in CVM and GCVM: (a) CVM: two-layer network with external Zachary's karate club and empty internal layer, (b) GCVM: two-layer network with external Zachary's karate club and star internal layer.

CVM [21,22] and GCVM [24–27] is implemented on a two-layer network (see Fig. 25a and 25b). In the CVM, individuals in the internal layer do not communicate, while in GCVM, individuals in the internal layer can communicate. The opinion dynamics process in CVM and GCVM is similar to BVM, but have more options: in the CVM, individuals can his/her internal opinion publicly with externalization rate π_e or accept his/her external opinion with internalization rate π_i . These two options do not exist in the BVM, while in GCVM, individuals can copy their internal neighbors' opinion with internal copy rate π_{c_i} , which is not allowed in the CVM.

3.3 Centrality measures in one- and two-layer networks

In this section, we represent several centrality measures. Some of them are defined for one-layer networks and can be applied for two layers separately, some of them take into account the multi-layer structure of a network. We also introduce

game-theoretical centrality measures and provide an algorithm to calculate their approximation when the network contains large number of nodes.

Definition 1. *A pairwise average shortest path for an external layer d_E in a two-layer network is*

$$d_E = \sum_{s,t \in \mathcal{V}_E} \frac{d_E(s,t)}{n_E(n_E - 1)}, \quad (12)$$

where $d_E(s,t)$ is a length of the shortest path between nodes s and t in the external layer, \mathcal{V}_E is a set of nodes in the external layer, $n_E = |\mathcal{V}_E|$ is a number of nodes in the external layer. Similarly, we can define pairwise average shortest path for an internal layer denoted by d_I .

Definition 2. *A graph density is a ratio of the number of edges $|\mathcal{E}|$ with respect to the maximal number of edges. Since internal layer is represented by an undirected graph, we define an internal graph density as in [76]:*

$$D_I = \frac{2|\mathcal{E}_I|}{|\mathcal{V}_I|(|\mathcal{V}_I| - 1)}. \quad (13)$$

The definitions above we will use in the experiments to analyze the influence of the network structure on the opinion dynamics process (See section 3.4.2).

3.3.1 Classical centrality measures

In this section we briefly introduce some (most well-known) centrality measures defined for one-layer networks. In the rest of the paper, we use V to denote the set of nodes in a one-layer network.

Betweenness centrality

Betweenness centrality of a node introduced in [29] gives the number of geodesics between all nodes that contain this node. It reflects the level of node participation in the dissemination of information between other nodes in a graph.

It is calculated by the formula:

$$C_b(v) = \frac{1}{n_b} \sum_{s,t \in V} \frac{\sigma_{s,t}(v)}{\sigma_{s,t}}, \quad (14)$$

where $\sigma_{s,t}$ indicates the number of shortest paths between nodes s and t , and $\sigma_{s,t}(v)$ is the number of shortest paths between nodes s and t containing node v . Normalization coefficient is $n_b = (|V| - 1)(|V| - 2)$ for $v \notin \{s, t\}$, otherwise $n_b = |V|(|V| - 1)$, where $|V|$ is the number of nodes in a one-layer network [28]. If $s = t$, $\sigma_{s,t} = 1$ and if $v \in \{s, t\}$, then $\sigma_{s,t}(v) = 0$.

Group betweenness centrality

Group betweenness centrality measure indicates a proportion of shortest paths connecting pairs of nongroup members that pass through the group (see [77]), and it is defined by formula:

$$C_{gb}(X) = \frac{1}{n_{gb}} \sum_{s,t \in V \setminus X} \frac{\sigma_{s,t}(X)}{\sigma_{s,t}}, \quad (15)$$

where $\sigma_{s,t}(X)$ is the number of shortest paths between nodes s and t passing through some nodes in group X . Normalization coefficient is $n_{gb} = (|V| - |X|)(|V| - |X| - 1)$, where $|X|$ is the number of nodes in group X .

Closeness centrality

In a connected graph, closeness centrality of node u is the reciprocal of a sum of lengths of the shortest paths between u and all other nodes in the graph [30–32]. When calculating closeness centrality, its normalized form is usually referred to as the one representing the average length of the shortest path instead of their sum, and it is calculated like this:

$$C_c(u) = \frac{n_c}{\sum_{v \in V \setminus \{u\}} d(v, u)}, \quad (16)$$

where normalization coefficient is $n_c = |V| - 1$.

Group closeness centrality

Group closeness centrality is the reciprocal of the sum of the shortest distances from the group to all nodes outside the group [77–79]. It is calculated as follows:

$$C_{gc}(X) = \frac{n_{gc}}{\sum_{v \in V \setminus X} d(v, X)}, \quad (17)$$

where $d(v, X)$ is the shortest distance between group X and v . Normalization coefficient is $n_{gc} = |V - X|$.

Degree centrality

Degree centrality of node v [33] is defined as

$$C_d(v) = \frac{v_d}{n_d}, \quad (18)$$

where v_d is a degree of node v , and normalization coefficient is $n_d = |V| - 1$.

Group degree centrality

Group degree centrality is the number of nodes outside the group connected with the nodes from this group [77, 78]. Normalized group degree centrality for group X is given by the formula:

$$C_{gd}(X) = \frac{|\{v_i \in V \setminus X | v_i \text{ is connected to } v_j \in X\}|}{n_{gd}}, \quad (19)$$

where normalization coefficient is $n_{gd} = |V| - |X|$.

3.3.2 Random walk based centralities

The second group of centrality measures is based on random walks, simple dynamical process that can occur on a network. Random walks can be also used to approximate other types of diffusion processes [34, 80, 81].

Random walk occupation centrality

The random walk occupation centrality [34] of node v is the probability of that node v being visited by a random walker during an infinitely long walk, and it is defined as

$$C_{rwoc}(v) = \lim_{t \rightarrow \infty} \frac{n_v(t)}{t}, \quad (20)$$

where $n_v(t)$ is the number of times node v is visited by a random walker during time interval t . Different exploration strategies can be used to calculate the occupation centrality, we use the uniform exploration strategy in this paper (i.e. each node jumps to its neighbor with the equal probability). In the weighted networks, jumping probabilities are proportional to the weights of the edges.

The analytical expressions of random walk occupation centrality with a uniform exploration strategy in interconnected multilayer networks are presented in [34].

Random walk betweenness centrality

The most common betweenness is the shortest path betweenness [29,34], where the centrality of a node v is relative to the number of shortest paths between all pairs of nodes passing through v . However, in real networks, entities (rumors, messages, or internet packets) that travel the network do not always follow the shortest path [34,82,83]. Therefore, the random walk betweenness centrality of node v is defined as the number of random walks between any pair (s, d) of nodes that pass through node v [35]:

$$C_{rwbc}(v) = \frac{1}{n_{rwbc}} \sum_{\substack{s,t \in V \\ s \neq t \\ v \neq s, v \neq t}} \mathbf{1}_{v \in \text{Path}_{s \rightarrow t}}, \quad (21)$$

where $n_{rwbc} = 2N(N - 1)$ is the normalization coefficient. The indicator function $\mathbf{1}_{v \in \text{Path}_{s \rightarrow t}}$ is equal to 1 if node v is in the path between nodes s and t , and 0 otherwise. The $\text{Path}_{s \rightarrow t}$ is the random path between nodes s and t in the network. Repeating the random walk process few times to get different random paths we get the average random walk betweenness centrality.

It will be useful to get the analytical expression of random walk betweenness centrality for nodes by absorbing random walk, where the absorbing state is selected to be the destination node d [35,81]. An extended analytical expression of random walk betweenness centrality for interconnected multilayer networks can be found in [34].

Random walk closeness centrality

A variant of closeness centrality is random walk closeness centrality, the computation of which is based on the mean first-passage time (MFPT). The MFPT is defined as the average number of steps to reach node d starting from a given node s . The lower average MFPT indicates that a node is on average more quickly accessible from other nodes. Therefore, a node with a lower average MFPT to all other nodes is considered more “central” in the network. The random walk closeness centrality is defined as the reciprocal of the average MFPT, and it is calculated by the formula:

$$C_{rwcc}(v) = \frac{n - 1}{\sum_{u \in V \setminus \{v\}} \tau_{uv}}, \quad (22)$$

where τ_{uv} is the MFPT from node u to node v . The MFPT matrix can be calculated analytically by means of Kemeny-Snell fundamental matrix Z [84, 85] or by means of absorbing random walks [81, 86].

The analytical expressions of random walk closeness centrality in interconnected multilayer networks can be found in [34].

3.3.3 Game-theoretic centrality measures

Shapley value based centrality

The Shapley value is a solution concept in cooperative game theory introduced by Lloyd Shapley in 1953 [37]. It is a measure of the average marginal contribution of a player to all possible coalitions. Shapley value is a solution concept assigning a singleton solution to the players as allocation given by formula:

$$\phi(i) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n - |S| - 1)!}{n!} (v(S \cup \{i\}) - v(S)), \quad (23)$$

where $S \subseteq V$ represents a coalition, the value of coalition S can be denoted by $v(S)$. We define characteristic function $v(S)$ as half of the sum of the weighted degrees of

all nodes in the subgraph induced by S , that is,

$$v(S) = \frac{1}{2} \sum_{\{i,j\} \subseteq S} W(i,j), \quad (24)$$

where $W(i,j)$ is the weight of the edge between nodes i and j within the subgraph induced by S , coefficient $\frac{1}{2}$ is used for correction in case of an undirected graph, each edge is counted twice when summing up over all pairs of nodes.

Algorithm 1 Calculation of the Shapley Values based on weighted graph

Require: A graph $G(V, E, W)$ with $n = |V|$ nodes

Ensure: Shapley value component $\phi(i)$ for each node $i \in V$

```

1: for all nodes  $i \in V$  do
2:   Initialize  $\phi(i) \leftarrow 0$ 
3: end for
4: for all nodes  $i \in V$  do
5:   for all subsets  $S \subseteq V \setminus \{i\}$  do
6:     Compute  $v(S) \leftarrow \sum_{\{j,k\} \subseteq S} W(j,k)$  within subgraph induced by  $S$ 
7:     Compute  $v(S \cup \{i\})$  within subgraph induced by  $S \cup \{i\}$ 
8:      $\Delta v(S, i) \leftarrow v(S \cup \{i\}) - v(S)$ 
9:     coeff  $\leftarrow \frac{|S|! \cdot (n - |S| - 1)!}{n!}$ 
10:     $\phi(i) \leftarrow \phi(i) + \text{coeff} \cdot \Delta v(S, i)$ 
11:   end for
12: end for
   return  $\phi(i)$  for all  $i \in V$ 

```

Algorithm 1 describes how to calculate the Shapley value based on a weighted graph. However, the Shapley value is computationally expensive, especially, for large networks with the large number of coalitions.⁹ Therefore, we propose a new approach to calculate an approximated Shapley value described in the next section.

Approximated Shapley value

According to the fact that the influence from other nodes is decreasing with an increase of the path length, we propose several ideas to fasten the calculation of the Shapley value:

1. *Depth limitation:* By limiting the depth of the reachable nodes considered, the number of subsets that need to be considered is reduced.

⁹For a network with n nodes, the total number of coalitions is equal to 2^n .

2. *Local subset iteration*: Iterating over subsets only within the reachable nodes, rather than the entire graph, decreases the number of iterations.
3. *Sampling from reachable nodes*: For a large number of reachable nodes, computational complexity can be reduced by random sampling, thereby decreasing the number of subsets iterated over.

We first define $\psi(i, d_{max})$ as the set of reachable nodes of node i up to given depth d_{max} excluding node i . We can calculate an approximated Shapley value of node i based on set $\psi(i, d_{max})$ by formula:

$$\phi_a(i) = \begin{cases} \sum_{S \subseteq \psi(i, d_{max})} \frac{v(S \cup \{i\}) - v(S)}{2^{|\psi(i, d_{max})|}} & \text{if } |\psi(i, d_{max})| < m, \\ \beta \sum_{S \subseteq \psi(i, d_{max})} \frac{v(S \cup \{i\}) - v(S)}{2^{|\psi(i, d_{max})|}} & \text{if } |\psi(i, d_{max})| \geq m, \end{cases} \quad (25)$$

where $\beta = \frac{|\psi(i, d_{max})| + 1}{m + 1}$ is a scaling factor, and m is the maximal number of reachable nodes considered.

For $|\psi(i, d_{max})| \geq m$, we create a random sample of m nodes from set $\psi(i, d_{max})$ several times and calculate the components of the Shapley value based on these samples. The sampling time $H_{|\psi(i, d_{max})|, m}$ is given by formula (26) (see [87]):

$$H_{|\psi(i, d_{max})|, m} = \left(\frac{|\psi(i, d_{max})| + \frac{1}{2}}{m} - \frac{1}{2} \right) (\ln |\psi(i, d_{max})| + \gamma) + \frac{1}{2}, \quad (26)$$

where $\gamma \approx 0.5772156649$ is the Euler-Mascheroni constant. This formula is the mathematical expectation of the number of samples for collecting m nodes from set $\psi(i, d_{max})$ until all reachable nodes are collected.¹⁰

Equation (25) gives a good approximation of the Shapley value, especially when the density of the graph is not very high (less than 0.7). We should highlight that our algorithm provide an accurate estimation of the ratio of the approximated component of the Shapley value to the sum of all its components. Knowing this ratio and exact value $v(N)$, it is easy to calculate the approximated Shapley value using the scaling factor ξ :

$$\xi = \frac{v(N)}{\sum_{i \in V} \phi_a(i)}. \quad (27)$$

The steps of calculation of the approximated Shapley value in weighted graphs is described in Algorithm 2. We use the same characteristic function as in the original

¹⁰We can consider this problem as the generalized coupon collector's problem [87].

Shapley value calculation, but we reduce the number of calculations following the ideas listed above.

We can get a more accurate approximated Shapley value by multiplying its components $\phi_a(i)$ given by equation (25) with factor ξ defined by (27). We include this step in the benchmark of Algorithm 2. The results of its work are presented in Section 3.4.1.

Algorithm 2 Calculation of approximated Shapley Value in weighted graph

Require: A weighted graph $G = (V, E, W)$, depth limit d_{max} , maximal size m of the set of reachable considered

Ensure: Approximated Shapley value $(\phi_a(i), i \in V)$

```

1: Initialize  $\phi_a(i) \leftarrow 0$  for each  $i \in V$ 
2: for  $i \in V$  do
3:    $\psi(i, d_{max}) \leftarrow$  Calculate or retrieve all reachable nodes of  $i$  up to depth  $d_{max}$ 
4:   if  $|\psi(i, d_{max})| < m$  then
5:     for each subset  $S \subseteq \psi(i, d_{max}) \setminus \{i\}$  do
6:       Compute  $v(S) \leftarrow \sum_{\{j,k\} \subseteq S} W(j, k)$  within subgraph induced by  $S$ 
7:       Compute  $v(S \cup \{i\})$  within subgraph induced by  $S \cup \{i\}$ 
8:        $\Delta v(S, i) \leftarrow v(S \cup \{i\}) - v(S)$ 
9:        $\phi_a(i) \leftarrow \phi_a(i) + \Delta v(S, i)$ 
10:    end for
11:     $\text{coeff} \leftarrow \frac{1}{2^{|\psi(i, d_{max})|}}$ 
12:     $\phi_a(i) \leftarrow \phi_a(i) \cdot \text{coeff}$ , normalize  $\phi_a(i)$  based on the number of subsets
13:  else
14:    Pick up  $m$  nodes randomly from  $\psi(i, d_{max})$  and repeat  $H_{|\psi(i, d_{max})|, m}$  times
15:    for  $i = 1$  to  $H_{|\psi(i, d_{max})|, m}$  do
16:       $s_{reachable} \leftarrow$  Randomly select a sample of  $m$  nodes from  $\psi(i, d_{max})$ ,
17:      for each subset  $S \subseteq s_{reachable} \setminus \{i\}$  do
18:        Calculate  $v(S)$  and  $v(S \cup \{i\})$  as before
19:         $\Delta v(S, i) \leftarrow v(S \cup \{i\}) - v(S)$ 
20:         $\phi_a(i) \leftarrow \phi_a(i) + \Delta v(S, i)$ 
21:      end for
22:    end for
23:     $\text{coeff} \leftarrow 1/2^{|\psi(i, d_{max})|} / H_{|\psi(i, d_{max})|, m} \cdot \frac{|\psi(i, d_{max})|+1}{m+1}$ 
24:     $\phi_a(i) \leftarrow \phi_a(i) \cdot \text{coeff}$ 
25:  end if
26: end for
27: Define scaling factor  $\xi \leftarrow \frac{v(N)}{\sum_{i \in V} \phi_a(i)}$  ▷ For accurate results
28: return  $\phi_a(i) \leftarrow \xi \cdot \phi_a(i)$  for all  $i \in V$ 

```

Myerson value based centrality

The Myerson value was introduced by Roger Myerson in 1977 [38], and it is an allocation rule when players are connected by a network structure. By modifying the calculation method of the Shapley value, Myerson takes into account connections in

the network, thereby reflecting the influence of network structure on the cooperative game. Consider a game where graph G is a tree, which consists on N nodes and characteristic function is determined by the scheme proposed in [88]. Every direct connection gives to coalition S a value r , where $0 \leq r \leq 1$. Players also obtain an impact from non-direct connections. This kind of impact will decrease with the increase of the path length. The characteristic function is defined as follows [40]:

$$v(S) = a_1 r + a_2 r^2 + \dots + a_k r^k + \dots + a_L r^L = \sum_{k=1}^L a_k r^k, \quad (28)$$

where L is a maximal distance between two nodes in the coalition; a_k is the number of paths of length k in this coalition; $v(i) = 0, \forall i \in N$.

Mazalov et al. in [89] proved that an allocation rule, that is, the Myerson value for unweighted graphs:

$$Y_i(v, g) = \frac{\sigma_1(i)}{2} r + \frac{\sigma_2(i)}{3} r^2 + \dots + \frac{\sigma_L(i)}{L+1} r^L = \sum_{k=1}^L \frac{\sigma_k(i)}{k+1} r^k, \quad (29)$$

where $\sigma_k(i)$ is a number of the paths of the length k including node i . The same approach to define an allocation can be applied to weighted graphs by converting the weight of the edge to the number of paths between two nodes, i.e. by converting a weighted graph to a multigraph [40].

Approximated Myerson value

However, calculation of the Myerson value is also computationally expensive, especially, for large networks. We can consider the rule of “six degrees separation” [90], implementing the idea that any two persons in the world who do not know each other only need a few intermediaries to establish a contact. Based on this idea, we can reduce the computational expense by limiting the maximal depth L of nodes with whom the node is connected. For a social network, the higher the density, the lower intermediate nodes are needed to connect two nodes. We redefine L as follows:

$$L = \begin{cases} 6, & \text{if } D \leq 0.2, \\ 2, & \text{if } 0.2 < D \leq 0.3, \\ 1, & \text{if } D > 0.3, \end{cases} \quad (30)$$

where D is the density of the network.

The algorithm for calculation of Myerson values in weighted graphs is presented in Algorithm 3. We use the same characteristic function as given by equation (29), but we cutoff the maximal depth to approximate the Myerson value given by formula (30).

Algorithm 3 Calculation of the approximated Myerson value for weighted graph

Require: A weighted graph $G(V, E, W)$, discount factor r (default 0.5), boolean *weight* for considering edge weights (default True), boolean *approximate* for approximation (default True), boolean *scale* for scaling (default False)

Ensure: Components of Myerson value $Y_i(v, g)$, $i \in V$

```

1:  $Y_i(v, g) \leftarrow 0$  for each  $i \in V$  ▷ Initialize Myerson values
2: if approximate then
3:    $L \leftarrow \begin{cases} 1, & \text{if } \text{density}(G) > 0.3, \\ 2, & \text{if } \text{density}(G) > 0.2, \\ 6, & \text{otherwise} \end{cases}$  ▷ Adjust  $L$  based on  $\text{density}(G)$ .
4: else
5:    $L \leftarrow |V| - 1$  ▷ Without approximation
6: end if
7: for all  $i \in V$  do
8:    $l2c \leftarrow$  Initialize a length- $\rightarrow$ count map for paths through  $i$ 
9:   for all pairs  $(start, end)$  in  $V \times V$  do
10:    for all  $path$  in all simple paths from  $start$  to  $end$  with  $length \leq L$  do
11:     if  $node \in path$  then
12:       $length \leftarrow \text{len}(path) - 1$ 
13:      if weight then
14:         $l2c[length] \leftarrow l2c[length] + \min_{(u,v) \in path} w(u, v)$ 
15:      else
16:         $l2c[length] \leftarrow l2c[length] + 1$ 
17:      end if
18:    end if
19:  end for
20: end for
21: for all  $(length, count) \in l2c$  do
22:    $count \leftarrow count/2$  ▷ Correct for double counting
23:    $Y_i(v, g) \leftarrow Y_i(v, g) + \left( count \cdot \frac{r^{length}}{length+1} \right)$ 
24: end for
25: end for
26: if scale then
27:   Define  $\xi \leftarrow \frac{v(N)}{\sum_{i \in V} Y_i(v, g)}$  ▷ For more accurate results
28:   return  $Y_i(v, g) \leftarrow \xi \cdot Y_i(v, g)$  for all  $i \in V$ 
29: else
30:   return  $Y_i(v, g)$  for all  $i \in V$ 
31: end if

```

Similarly to equation (27), we can also define the scaling factor ξ for the approximated Myerson value as follows:

$$\xi = \frac{v(N)}{\sum_{i \in V} Y_i(v, g)}. \quad (31)$$

In both formulae (27) and (31), $v(N)$ is used, and to calculate it for the Myerson value we need to count the number of paths for all lengths, i.e. a_1, a_2, \dots, a_L . It is much more computationally expensive than in the case of the Shapley value. But after the ξ scaling, we obtain a more accurate approximation which will be shown in Section 3.4.1.

3.4 Experiments

Network structure has a huge impact on key performance indicators (KPIs) of opinion dynamics realized on this network. Therefore, we define several characteristics of a network which, in our opinion, have most significant correlation with KPIs of opinion dynamics. Our experiments are organized as follows: in Section 3.4.1 we provide the series of experiments in which we calculated the centralities based on approximated Shapley and Myerson values (realizations of Algorithms 2 and 3) for the graphs with varying density. We analyze correlation between KPIs of opinion dynamics and network properties in Section 3.4.2.

3.4.1 Centralities based on the Shapley and Myerson values

Due to the computational complexity of the Shapley value and Myerson value, we run our experiments on networks composed by 20 nodes for a given density. We create a network as follows: randomly and repeatedly take two different nodes from the set of nodes and add a connection between them until the density reaches a

desired value. We designed the following experiments to evaluate the performance of the proposed centrality measures:

1. **Shapley-value based centrality:** The density of a network takes the values: $0.1, 0.2, \dots, 1.0$. For each weighted or unweighted graph, we calculate the exact Shapley value and an approximated Shapley value. We compare (i) these two values and (ii) computation time for these two methods.
2. **Myerson-value based centrality:** The density of a network takes the values: $0.1, 0.11, \dots, 0.2$.¹¹ For each weighted or unweighted graph, we calculate the exact Myerson value and an approximated Myerson value. We compare: (i) these two values and (ii) computation time for these two methods. We also make the series of experiments on networks composed by 10 nodes with density from 0.05 to 1.0 to show efficiency of our algorithm to approximate the Myerson value.
3. **Comparison with classical centrality measures:** Based on the real social network dataset “Zachary’s karate club”, we created a two-layer network by adding different internal network structures. We reduce a two-layer network to a one-layer weighted network using opinion dynamics parameters by the method described in Section 3.1. The most important nodes in the network are nodes 0 (instructor — Mr Hi) and 33 (manager — John A). We define the coefficient of accuracy or simply accuracy of centrality measures as follows:

$$Ac = \frac{|\text{Top 2 nodes according to a centrality measure} \cap \{0, 33\}|}{2} \cdot 100\%. \quad (32)$$

The meaning of Ac is the percentage of important nodes (0 and 33) in the top two nodes identified by considered centrality measures. We also compare the accuracy Ac of the proposed centrality measures with the accuracy Ac of classical centrality measures (betweenness and closeness centralities).

The results of this part of experiments are presented in Tables 16–23. In Table 16 we provide the results for the graphs with 20 nodes and different density (from 0.1 to 1.0), weighted and unweighted graphs. We compare the computation time for the exact Shapley value (column “SV time”) and for an approximated Shapley value (column “ASV time”). In Table 16 we also present a root mean square error

¹¹We limit the density to the set $\{0.1, 0.11, \dots, 0.2\}$ because even a small increase in density significantly increases computation time. This makes the realization of the series of experiments quite complicated.

for an approximated Shapley value (column “RMSE ASV”) and for the ratio of the approximated Shapley value component to the sum of all its components (column “RMSE ratio ASV”). The lower the value of RMSE, the more accurate approximation we obtain. We can make the following conclusions based on the results from Table 16:

1. The computation time of an approximated Shapley value is much faster than the time of the exact Shapley value (see columns “SV time” and “ASV time”).
2. The root mean square error of the approximated Shapley value is very small (see column “RMSE ASV”), and root mean square error of the ratio is even smaller than the first one (see column “RMSE ratio ASV”). The ratio here is referred to the normalized values. The RMSE of ratio is the RMSE between the exact normalized Shapley values and the approximated normalized Shapley values. RMSE increases with an increase of the graph density because of the sampling procedure.
3. The weighted graph introduces greater uncertainty in the sampling process when graph density is high (≥ 0.7), leading to a slightly higher RMSE. However, even in the worst case (see graph “20-1.0” in Table 16), our algorithm still demonstrates a high accuracy.

Table 16: Results on the Shapley value when ξ scaling factor is applied

Graph	Weighted	SV time	ASV time	RMSE ASV	RMSE ratio ASV
20-0.1	True	$7.2 \cdot 10^2$	$2.2 \cdot 10^{-3}$	$2.6 \cdot 10^{-22}$	$5.5 \cdot 10^{-27}$
20-0.2	True	$8.7 \cdot 10^2$	$1.9 \cdot 10^{-2}$	$3.1 \cdot 10^{-22}$	$5.6 \cdot 10^{-27}$
20-0.3	True	$1 \cdot 10^3$	$1.5 \cdot 10^{-1}$	$1.6 \cdot 10^{-22}$	$1.2 \cdot 10^{-27}$
20-0.4	True	$1.2 \cdot 10^3$	$9.3 \cdot 10^{-1}$	$1.7 \cdot 10^{-22}$	$6.1 \cdot 10^{-28}$
20-0.5	True	$1.3 \cdot 10^3$	$2.2 \cdot 10^0$	$7.1 \cdot 10^{-23}$	$2 \cdot 10^{-28}$
20-0.6	True	$1.5 \cdot 10^3$	$1.9 \cdot 10^1$	$7.7 \cdot 10^{-23}$	$1.8 \cdot 10^{-28}$
20-0.7	True	$1.6 \cdot 10^3$	$4.7 \cdot 10^1$	$3.3 \cdot 10^{-2}$	$6 \cdot 10^{-8}$
20-0.8	True	$1.7 \cdot 10^3$	$1.7 \cdot 10^2$	$2 \cdot 10^0$	$2.9 \cdot 10^{-6}$
20-0.9	True	$1.8 \cdot 10^3$	$2.5 \cdot 10^2$	$6.9 \cdot 10^0$	$7.9 \cdot 10^{-6}$
20-1.0	True	$1.9 \cdot 10^3$	$3.2 \cdot 10^2$	$9.9 \cdot 10^0$	$9 \cdot 10^{-6}$
20-0.1	False	$7.2 \cdot 10^2$	$3.4 \cdot 10^{-3}$	$4.5 \cdot 10^{-23}$	$1.1 \cdot 10^{-26}$
20-0.2	False	$8.8 \cdot 10^2$	$1.6 \cdot 10^{-2}$	$1.1 \cdot 10^{-22}$	$7.8 \cdot 10^{-27}$
20-0.3	False	$1 \cdot 10^3$	$9.6 \cdot 10^{-2}$	$1.1 \cdot 10^{-22}$	$6.6 \cdot 10^{-27}$
20-0.4	False	$1.2 \cdot 10^3$	$5.5 \cdot 10^{-1}$	$1.5 \cdot 10^{-22}$	$1.6 \cdot 10^{-26}$
20-0.5	False	$1.3 \cdot 10^3$	$2.9 \cdot 10^0$	$1.1 \cdot 10^{-22}$	$8.6 \cdot 10^{-27}$
20-0.6	False	$1.5 \cdot 10^3$	$2.5 \cdot 10^1$	$1.4 \cdot 10^{-22}$	$9.4 \cdot 10^{-27}$
20-0.7	False	$1.6 \cdot 10^3$	$6.2 \cdot 10^1$	$3.8 \cdot 10^{-4}$	$2.1 \cdot 10^{-8}$
20-0.8	False	$1.7 \cdot 10^3$	$1.7 \cdot 10^2$	$1.8 \cdot 10^{-3}$	$7.7 \cdot 10^{-8}$
20-0.9	False	$1.8 \cdot 10^3$	$2.5 \cdot 10^2$	$2 \cdot 10^{-3}$	$6.9 \cdot 10^{-8}$
20-1.0	False	$1.9 \cdot 10^3$	$3.2 \cdot 10^2$	$4.1 \cdot 10^{-21}$	$8.6 \cdot 10^{-34}$

In Tables 17 and 18 we present the results on computation of the Myerson value (exact and approximated) without and with scaling factor ξ defined by (31),

respectively. In both tables we provide the results for the graphs with 20 nodes and different density (from 0.1 to 0.2), weighted and unweighted graphs. We compare the computation time for the exact Myerson value (column “MV time”) and for an approximated Myerson value (column “AMV time”). In these two tables we also present a root mean square error for an approximated Myerson value (column “RMSE AMV”) and for the ratio of the approximated Myerson value component to the sum of all its components (column “RMSE ratio AMV”). Conclusions from Tables 17 and 18 are

1. The computational complexity of the exact Myerson-value based centrality grows rapidly with the increase of the network density (column “MV time”).
2. The root mean square error of an approximated Myerson value grows with an increase of the network density (see column “RMSE AMV”), but the root mean square error of the ratio is very small. The RMSE of the ratio means the RMSE between the normalized approximated and normalized exact Myerson values. Therefore, we can recommend to use an approximation of the Myerson value as an approximated centrality measure.
3. If we estimate the effect of scaling factor ξ on the results, we can compare “AMV time” in Tables 17 and 18, and conclude that without ξ the computation time is much smaller, but the RMSE AMV is higher. While the RMSE ratio AMV between Tables 17 and 18 are almost the same.
4. Comparing Tables 17 and 18, we cannot see a significant difference in the results referred to weighted and unweighted graphs.

Next, we demonstrate the efficiency of our algorithm to approximate the Myerson value in high-density graphs. To do this, we examine graphs with 10 nodes and with density from 0.05 to 1.0 with step 0.05, weighted and unweighted graphs. The results of our experiments are presented in Tables 19 and 20 for the cases without and with scaling factor, respectively.

By comparing the pairs of Tables 17 and 18, and Tables 19 and 20, we can conclude that the scaling factor ξ has a very limited effect on reducing the error of an approximated Myerson value and moreover using ξ significantly increases computation time. But the RMSE of ratio is very small. Therefore, we could recommend to use the Myerson-value based centrality approximation without scaling factor ξ and better its ratio as an approximation of the Myerson value.

Table 17: Results on the Myerson value without ξ scaling

Graph	Weighted	MV time	AMV time	RMSE AMV	RMSE ratio AMV
20-0.1	True	$1.03 \cdot 10^{-1}$	$1.62 \cdot 10^{-1}$	$4.37 \cdot 10^{-4}$	$1.48 \cdot 10^{-8}$
20-0.11	True	$1.91 \cdot 10^{-1}$	$1.67 \cdot 10^{-1}$	$8.75 \cdot 10^{-4}$	$1.7 \cdot 10^{-8}$
20-0.12	True	$4.17 \cdot 10^{-1}$	$2.23 \cdot 10^{-1}$	$1.54 \cdot 10^{-1}$	$5.28 \cdot 10^{-7}$
20-0.13	True	$1.02 \cdot 10^0$	$3.56 \cdot 10^{-1}$	$5.28 \cdot 10^{-1}$	$8.15 \cdot 10^{-7}$
20-0.14	True	$2.21 \cdot 10^0$	$6.15 \cdot 10^{-1}$	$2.14 \cdot 10^0$	$1.93 \cdot 10^{-6}$
20-0.15	True	$8.78 \cdot 10^0$	$9.24 \cdot 10^{-1}$	$1.89 \cdot 10^1$	$4.3 \cdot 10^{-6}$
20-0.16	True	$1.8 \cdot 10^1$	$1.48 \cdot 10^0$	$1.27 \cdot 10^2$	$1.25 \cdot 10^{-5}$
20-0.17	True	$2.87 \cdot 10^1$	$2.14 \cdot 10^0$	$4.06 \cdot 10^2$	$3.24 \cdot 10^{-5}$
20-0.18	True	$6.34 \cdot 10^1$	$2.77 \cdot 10^0$	$1.55 \cdot 10^3$	$4.7 \cdot 10^{-5}$
20-0.19	True	$1.45 \cdot 10^2$	$3.58 \cdot 10^0$	$4.82 \cdot 10^3$	$8.66 \cdot 10^{-5}$
20-0.2	True	$2.94 \cdot 10^2$	$4.45 \cdot 10^0$	$1.33 \cdot 10^4$	$1.35 \cdot 10^{-4}$
20-0.1	False	$3.3 \cdot 10^{-1}$	$1.96 \cdot 10^{-1}$	$1.12 \cdot 10^{-2}$	$1.43 \cdot 10^{-6}$
20-0.11	False	$3.45 \cdot 10^{-1}$	$2.16 \cdot 10^{-1}$	$1.73 \cdot 10^{-2}$	$1.41 \cdot 10^{-6}$
20-0.12	False	$4.87 \cdot 10^{-1}$	$2.17 \cdot 10^{-1}$	$3.43 \cdot 10^{-2}$	$1.52 \cdot 10^{-6}$
20-0.13	False	$1.37 \cdot 10^0$	$3.79 \cdot 10^{-1}$	$3.95 \cdot 10^{-1}$	$5.73 \cdot 10^{-6}$
20-0.14	False	$3.69 \cdot 10^0$	$5.81 \cdot 10^{-1}$	$2.71 \cdot 10^0$	$1.53 \cdot 10^{-5}$
20-0.15	False	$7.65 \cdot 10^0$	$1.05 \cdot 10^0$	$1.38 \cdot 10^1$	$2.75 \cdot 10^{-5}$
20-0.16	False	$1.71 \cdot 10^1$	$1.38 \cdot 10^0$	$5.94 \cdot 10^1$	$4.19 \cdot 10^{-5}$
20-0.17	False	$4.28 \cdot 10^1$	$1.62 \cdot 10^0$	$2.97 \cdot 10^2$	$6.29 \cdot 10^{-5}$
20-0.18	False	$8.25 \cdot 10^1$	$1.88 \cdot 10^0$	$8.04 \cdot 10^2$	$6.73 \cdot 10^{-5}$
20-0.19	False	$1.69 \cdot 10^2$	$2.56 \cdot 10^0$	$2.72 \cdot 10^3$	$7.6 \cdot 10^{-5}$
20-0.2	False	$3.66 \cdot 10^2$	$3.57 \cdot 10^0$	$8.75 \cdot 10^3$	$1.31 \cdot 10^{-4}$

Table 18: Results on the Myerson value when ξ scaling is applied

Graph	Weighted	MV time	AMV time	RMSE AMV	RMSE ratio AMV
20-0.1	True	$1.23 \cdot 10^{-1}$	$1.1 \cdot 10^{-1}$	$1.98 \cdot 10^{-4}$	$1.48 \cdot 10^{-8}$
20-0.11	True	$1.57 \cdot 10^{-1}$	$2.11 \cdot 10^{-1}$	$3.21 \cdot 10^{-4}$	$1.7 \cdot 10^{-8}$
20-0.12	True	$4.53 \cdot 10^{-1}$	$2.21 \cdot 10^{-1}$	$1.55 \cdot 10^{-2}$	$5.28 \cdot 10^{-7}$
20-0.13	True	$9.1 \cdot 10^{-1}$	$4.33 \cdot 10^{-1}$	$3.15 \cdot 10^{-2}$	$8.15 \cdot 10^{-7}$
20-0.14	True	$2.09 \cdot 10^0$	$6.7 \cdot 10^{-1}$	$1.08 \cdot 10^{-1}$	$1.93 \cdot 10^{-6}$
20-0.15	True	$7.83 \cdot 10^0$	$1.24 \cdot 10^0$	$4.81 \cdot 10^{-1}$	$4.3 \cdot 10^{-6}$
20-0.16	True	$1.65 \cdot 10^1$	$2.24 \cdot 10^0$	$3.85 \cdot 10^0$	$1.25 \cdot 10^{-5}$
20-0.17	True	$2.63 \cdot 10^1$	$3.28 \cdot 10^0$	$2.36 \cdot 10^1$	$3.24 \cdot 10^{-5}$
20-0.18	True	$5.91 \cdot 10^1$	$5.91 \cdot 10^0$	$6.15 \cdot 10^1$	$4.7 \cdot 10^{-5}$
20-0.19	True	$1.41 \cdot 10^2$	$1.15 \cdot 10^1$	$2.07 \cdot 10^2$	$8.66 \cdot 10^{-5}$
20-0.2	True	$2.92 \cdot 10^2$	$2.06 \cdot 10^1$	$6.48 \cdot 10^2$	$1.35 \cdot 10^{-4}$
20-0.1	False	$2.75 \cdot 10^{-1}$	$2.43 \cdot 10^{-1}$	$1.59 \cdot 10^{-3}$	$1.43 \cdot 10^{-6}$
20-0.11	False	$3.64 \cdot 10^{-1}$	$1.94 \cdot 10^{-1}$	$1.81 \cdot 10^{-3}$	$1.41 \cdot 10^{-6}$
20-0.12	False	$4.56 \cdot 10^{-1}$	$2.47 \cdot 10^{-1}$	$2.65 \cdot 10^{-3}$	$1.52 \cdot 10^{-6}$
20-0.13	False	$1.34 \cdot 10^0$	$4.13 \cdot 10^{-1}$	$2.01 \cdot 10^{-2}$	$5.73 \cdot 10^{-6}$
20-0.14	False	$3.33 \cdot 10^0$	$6.97 \cdot 10^{-1}$	$1.18 \cdot 10^{-1}$	$1.53 \cdot 10^{-5}$
20-0.15	False	$7.17 \cdot 10^0$	$1.29 \cdot 10^0$	$5.64 \cdot 10^{-1}$	$2.75 \cdot 10^{-5}$
20-0.16	False	$1.55 \cdot 10^1$	$2.01 \cdot 10^0$	$1.92 \cdot 10^0$	$4.19 \cdot 10^{-5}$
20-0.17	False	$3.93 \cdot 10^1$	$3.3 \cdot 10^0$	$7.44 \cdot 10^0$	$6.29 \cdot 10^{-5}$
20-0.18	False	$7.78 \cdot 10^1$	$5.58 \cdot 10^0$	$1.6 \cdot 10^1$	$6.73 \cdot 10^{-5}$
20-0.19	False	$1.64 \cdot 10^2$	$1.06 \cdot 10^1$	$4.58 \cdot 10^1$	$7.6 \cdot 10^{-5}$
20-0.2	False	$3.43 \cdot 10^2$	$1.99 \cdot 10^1$	$2.06 \cdot 10^2$	$1.31 \cdot 10^{-4}$

Table 19: Results on the Myerson value without ξ scaling for networks with 10 nodes

Graph	Weighted	MV time	AMV time	RMSE AMV	RMSE ratio AMV
10-0.05	True	$1.24 \cdot 10^{-3}$	$1.59 \cdot 10^{-2}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.1	True	$1.79 \cdot 10^{-3}$	$1.58 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.15	True	$1.27 \cdot 10^{-2}$	$3.17 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.2	True	$7.33 \cdot 10^{-3}$	$1.05 \cdot 10^{-2}$	$9.65 \cdot 10^{-6}$	$7.63 \cdot 10^{-10}$
10-0.25	True	$1.49 \cdot 10^{-2}$	$2.89 \cdot 10^{-3}$	$3.27 \cdot 10^1$	$1.64 \cdot 10^{-4}$
10-0.3	True	$6.69 \cdot 10^{-2}$	$3.35 \cdot 10^{-3}$	$2.79 \cdot 10^2$	$1.91 \cdot 10^{-4}$
10-0.35	True	$1.17 \cdot 10^{-1}$	$1.78 \cdot 10^{-3}$	$1.85 \cdot 10^3$	$5.12 \cdot 10^{-4}$
10-0.4	True	$2.6 \cdot 10^{-1}$	$1.2 \cdot 10^{-3}$	$4.58 \cdot 10^3$	$4.08 \cdot 10^{-4}$
10-0.45	True	$5.53 \cdot 10^{-1}$	$1.21 \cdot 10^{-3}$	$1.05 \cdot 10^4$	$6.26 \cdot 10^{-4}$
10-0.5	True	$1.21 \cdot 10^0$	$1.37 \cdot 10^{-3}$	$1.56 \cdot 10^4$	$5.53 \cdot 10^{-4}$
10-0.55	True	$2.56 \cdot 10^0$	$1.26 \cdot 10^{-3}$	$3.92 \cdot 10^4$	$4.49 \cdot 10^{-4}$
10-0.6	True	$5.7 \cdot 10^0$	$1.62 \cdot 10^{-3}$	$1.11 \cdot 10^5$	$6.17 \cdot 10^{-4}$
10-0.65	True	$1.05 \cdot 10^1$	$1.33 \cdot 10^{-3}$	$2.31 \cdot 10^5$	$7.47 \cdot 10^{-4}$
10-0.7	True	$1.99 \cdot 10^1$	$1.33 \cdot 10^{-3}$	$5.19 \cdot 10^5$	$6.84 \cdot 10^{-4}$
10-0.75	True	$3.55 \cdot 10^1$	$6.44 \cdot 10^{-3}$	$1 \cdot 10^6$	$5.96 \cdot 10^{-4}$
10-0.8	True	$7.36 \cdot 10^1$	$1.38 \cdot 10^{-3}$	$3.42 \cdot 10^6$	$5.9 \cdot 10^{-4}$
10-0.85	True	$1.15 \cdot 10^2$	$1.38 \cdot 10^{-3}$	$8.11 \cdot 10^6$	$3.76 \cdot 10^{-4}$
10-0.9	True	$1.73 \cdot 10^2$	$1.39 \cdot 10^{-3}$	$1.63 \cdot 10^7$	$3.92 \cdot 10^{-4}$
10-0.95	True	$2.48 \cdot 10^2$	$1.37 \cdot 10^{-3}$	$2.74 \cdot 10^7$	$3.7 \cdot 10^{-4}$
10-1.0	True	$4.33 \cdot 10^2$	$1.37 \cdot 10^{-3}$	$7.31 \cdot 10^7$	$2.79 \cdot 10^{-4}$
10-0.05	False	$1.15 \cdot 10^{-3}$	$1.06 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.1	False	$2.09 \cdot 10^{-3}$	$1.99 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.15	False	$3.4 \cdot 10^{-3}$	$2.84 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.2	False	$7.57 \cdot 10^{-3}$	$1.08 \cdot 10^{-2}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.25	False	$3.14 \cdot 10^{-2}$	$8.37 \cdot 10^{-3}$	$2.56 \cdot 10^0$	$1.57 \cdot 10^{-4}$
10-0.3	False	$3.12 \cdot 10^{-2}$	$3.11 \cdot 10^{-3}$	$8.85 \cdot 10^0$	$9.04 \cdot 10^{-5}$
10-0.35	False	$8.49 \cdot 10^{-2}$	$1.13 \cdot 10^{-3}$	$5.98 \cdot 10^1$	$1.53 \cdot 10^{-4}$
10-0.4	False	$2.18 \cdot 10^{-1}$	$1.33 \cdot 10^{-3}$	$2.28 \cdot 10^2$	$1.29 \cdot 10^{-4}$
10-0.45	False	$4.74 \cdot 10^{-1}$	$1.16 \cdot 10^{-3}$	$6.23 \cdot 10^2$	$9.58 \cdot 10^{-5}$
10-0.5	False	$9.61 \cdot 10^{-1}$	$1.17 \cdot 10^{-3}$	$1.81 \cdot 10^3$	$8.71 \cdot 10^{-5}$
10-0.55	False	$1.95 \cdot 10^0$	$1.21 \cdot 10^{-3}$	$4.91 \cdot 10^3$	$7.4 \cdot 10^{-5}$
10-0.6	False	$4.15 \cdot 10^0$	$1.3 \cdot 10^{-3}$	$1.97 \cdot 10^4$	$4.98 \cdot 10^{-5}$
10-0.65	False	$7.11 \cdot 10^0$	$1.24 \cdot 10^{-3}$	$4.79 \cdot 10^4$	$5.33 \cdot 10^{-5}$
10-0.7	False	$1.14 \cdot 10^1$	$1.33 \cdot 10^{-3}$	$1.07 \cdot 10^5$	$9.37 \cdot 10^{-5}$
10-0.75	False	$1.86 \cdot 10^1$	$3.51 \cdot 10^{-3}$	$2.46 \cdot 10^5$	$1.6 \cdot 10^{-4}$
10-0.8	False	$3.77 \cdot 10^1$	$1.39 \cdot 10^{-3}$	$8.53 \cdot 10^5$	$1.07 \cdot 10^{-4}$
10-0.85	False	$5.26 \cdot 10^1$	$1.29 \cdot 10^{-3}$	$1.73 \cdot 10^6$	$1.35 \cdot 10^{-4}$
10-0.9	False	$7.45 \cdot 10^1$	$1.31 \cdot 10^{-3}$	$3.64 \cdot 10^6$	$9.85 \cdot 10^{-5}$
10-0.95	False	$1.05 \cdot 10^2$	$1.3 \cdot 10^{-3}$	$7.74 \cdot 10^6$	$4.85 \cdot 10^{-5}$
10-1.0	False	$1.71 \cdot 10^2$	$1.28 \cdot 10^{-3}$	$2.16 \cdot 10^7$	$0 \cdot 10^0$

In Table 16, the results for the weighted graph with density 1.0 show the highest RMSE, and in Table 18 the results for the weighted graph with density 0.2 also show the highest RMSE. Therefore, we decide to carefully examine the high-density case and examine the Shapley value for the weighted graph with density 1.0 and the Myerson value for the weighted graph with density 0.2. To make analysis, we show the components of the exact and approximated Shapley values in Table 21 for the former graph and the exact and approximated Myerson values in Table 22 for the latter graph. In these tables we provide a relative error which is calculated by taking the difference between the approximated value and the exact value dividing this difference by the exact value, that is, $\frac{\text{approximated value} - \text{exact value}}{\text{exact value}}$.

Analyzing Table 21, we can say that our algorithm provides a very accurate approximation for the Shapley value such that only for two nodes (Nodes 5 and 19) the absolute relative error is larger than 5%.

Table 20: Results on the Myerson value when ξ scaling is applied for networks with 10 nodes

Graph	Weighted	MV time	AMV time	RMSE AMV	RMSE ratio AMV
10-0.05	True	$1.24 \cdot 10^{-3}$	$1.22 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.1	True	$3.08 \cdot 10^{-3}$	$1.99 \cdot 10^{-3}$	$6.24 \cdot 10^{-32}$	$0 \cdot 10^0$
10-0.15	True	$4.67 \cdot 10^{-3}$	$3.73 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.2	True	$1.2 \cdot 10^{-2}$	$1.74 \cdot 10^{-2}$	$3.25 \cdot 10^{-6}$	$7.63 \cdot 10^{-10}$
10-0.25	True	$1.46 \cdot 10^{-2}$	$4.48 \cdot 10^{-3}$	$1.67 \cdot 10^0$	$1.64 \cdot 10^{-4}$
10-0.3	True	$4.97 \cdot 10^{-2}$	$8.24 \cdot 10^{-3}$	$6.05 \cdot 10^0$	$1.91 \cdot 10^{-4}$
10-0.35	True	$1.05 \cdot 10^{-1}$	$1.69 \cdot 10^{-2}$	$3.5 \cdot 10^1$	$5.12 \cdot 10^{-4}$
10-0.4	True	$2.57 \cdot 10^{-1}$	$2.63 \cdot 10^{-2}$	$6.41 \cdot 10^1$	$4.08 \cdot 10^{-4}$
10-0.45	True	$5.52 \cdot 10^{-1}$	$7.07 \cdot 10^{-2}$	$2.14 \cdot 10^2$	$6.26 \cdot 10^{-4}$
10-0.5	True	$1.18 \cdot 10^0$	$1.27 \cdot 10^{-1}$	$2.91 \cdot 10^2$	$5.53 \cdot 10^{-4}$
10-0.55	True	$2.5 \cdot 10^0$	$3.8 \cdot 10^{-1}$	$5.78 \cdot 10^2$	$4.49 \cdot 10^{-4}$
10-0.6	True	$5.81 \cdot 10^0$	$5.76 \cdot 10^{-1}$	$2.15 \cdot 10^3$	$6.17 \cdot 10^{-4}$
10-0.65	True	$1.06 \cdot 10^1$	$1.24 \cdot 10^0$	$5.36 \cdot 10^3$	$7.47 \cdot 10^{-4}$
10-0.7	True	$1.98 \cdot 10^1$	$2.14 \cdot 10^0$	$1.12 \cdot 10^4$	$6.84 \cdot 10^{-4}$
10-0.75	True	$3.57 \cdot 10^1$	$3.66 \cdot 10^0$	$1.9 \cdot 10^4$	$5.96 \cdot 10^{-4}$
10-0.8	True	$7.43 \cdot 10^1$	$6.77 \cdot 10^0$	$6.41 \cdot 10^4$	$5.9 \cdot 10^{-4}$
10-0.85	True	$1.16 \cdot 10^2$	$1.11 \cdot 10^1$	$9.71 \cdot 10^4$	$3.76 \cdot 10^{-4}$
10-0.9	True	$1.75 \cdot 10^2$	$1.71 \cdot 10^1$	$2.03 \cdot 10^5$	$3.92 \cdot 10^{-4}$
10-0.95	True	$2.48 \cdot 10^2$	$2.45 \cdot 10^1$	$3.22 \cdot 10^5$	$3.7 \cdot 10^{-4}$
10-1.0	True	$4.33 \cdot 10^2$	$4.33 \cdot 10^1$	$6.49 \cdot 10^5$	$2.79 \cdot 10^{-4}$
10-0.05	False	$1.23 \cdot 10^{-3}$	$1.29 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.1	False	$2.28 \cdot 10^{-3}$	$2.55 \cdot 10^{-3}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.15	False	$4.91 \cdot 10^{-3}$	$1.73 \cdot 10^{-2}$	$0 \cdot 10^0$	$0 \cdot 10^0$
10-0.2	False	$2.15 \cdot 10^{-2}$	$7.96 \cdot 10^{-3}$	$2.16 \cdot 10^{-31}$	$2.65 \cdot 10^{-33}$
10-0.25	False	$1.64 \cdot 10^{-2}$	$9.18 \cdot 10^{-3}$	$5.35 \cdot 10^{-2}$	$1.57 \cdot 10^{-4}$
10-0.3	False	$4.56 \cdot 10^{-2}$	$6.48 \cdot 10^{-3}$	$7.23 \cdot 10^{-2}$	$9.04 \cdot 10^{-5}$
10-0.35	False	$6.45 \cdot 10^{-2}$	$6.84 \cdot 10^{-3}$	$3.23 \cdot 10^{-1}$	$1.53 \cdot 10^{-4}$
10-0.4	False	$2.43 \cdot 10^{-1}$	$1.99 \cdot 10^{-2}$	$1.02 \cdot 10^0$	$1.29 \cdot 10^{-4}$
10-0.45	False	$4.54 \cdot 10^{-1}$	$3.73 \cdot 10^{-2}$	$2 \cdot 10^0$	$9.58 \cdot 10^{-5}$
10-0.5	False	$9.12 \cdot 10^{-1}$	$8.23 \cdot 10^{-2}$	$5.1 \cdot 10^0$	$8.71 \cdot 10^{-5}$
10-0.55	False	$1.86 \cdot 10^0$	$2.11 \cdot 10^{-1}$	$1.15 \cdot 10^1$	$7.4 \cdot 10^{-5}$
10-0.6	False	$4.29 \cdot 10^0$	$3.61 \cdot 10^{-1}$	$3.03 \cdot 10^1$	$4.98 \cdot 10^{-5}$
10-0.65	False	$7.36 \cdot 10^0$	$6.8 \cdot 10^{-1}$	$7.83 \cdot 10^1$	$5.33 \cdot 10^{-5}$
10-0.7	False	$1.14 \cdot 10^1$	$1.51 \cdot 10^0$	$3.05 \cdot 10^2$	$9.37 \cdot 10^{-5}$
10-0.75	False	$1.94 \cdot 10^1$	$2.03 \cdot 10^0$	$1.23 \cdot 10^3$	$1.6 \cdot 10^{-4}$
10-0.8	False	$3.74 \cdot 10^1$	$3.41 \cdot 10^0$	$2.9 \cdot 10^3$	$1.07 \cdot 10^{-4}$
10-0.85	False	$5.27 \cdot 10^1$	$5.04 \cdot 10^0$	$7.35 \cdot 10^3$	$1.35 \cdot 10^{-4}$
10-0.9	False	$7.45 \cdot 10^1$	$6.5 \cdot 10^0$	$1.13 \cdot 10^4$	$9.85 \cdot 10^{-5}$
10-0.95	False	$1.05 \cdot 10^2$	$9.53 \cdot 10^0$	$1.19 \cdot 10^4$	$4.85 \cdot 10^{-5}$
10-1.0	False	$1.71 \cdot 10^2$	$1.63 \cdot 10^1$	$0 \cdot 10^0$	$0 \cdot 10^0$

Analyzing the results in Table 22, we can find that some nodes (Nodes 1, 2, 8, and 9) have a high relative error in approximation. The reason is that in our algorithm we count the number of paths not for all lengths ($length \leq L$). By increasing L we can obtain a more accurate result, but it will increase the computational complexity.

The last item in analysis of this series of experiments is to compare the accuracy defined by (32) for the approximated Shapley and Myerson values, and for the classical centrality measures (betweenness and closeness). We summarize these results in Table 23. Both centrality measures proposed in this paper have a higher accuracy than the classical centrality measures. In particular, an approximated Shapley-value based centrality can identify the most important nodes with 100% accuracy for all examined graphs. For the approximated Myerson value based cen-

Table 21: Exact Shapley value vs an approximated Shapley value for “graph-1.0”

Nodes	SV	SV ratio	ASV	ASV ratio	Rel. error ASV/SV
0	52.0000	4.96%	53.3579	5.09%	2.61%
1	46.5000	4.43%	46.5090	4.43%	$1.83 \cdot 10^{-2}\%$
2	50.5000	4.81%	49.0574	4.68%	-2.86%
3	53.0000	5.05%	53.6764	5.12%	1.28%
4	50.5000	4.81%	50.0131	4.77%	-0.96%
5	50.0000	4.77%	54.3135	5.18%	8.63%
6	50.0000	4.77%	51.2873	4.89%	2.57%
7	63.0000	6.01%	62.5960	5.97%	-0.64%
8	55.0000	5.24%	53.3579	5.09%	-2.98%
9	44.0000	4.19%	44.4384	4.24%	0.99%
10	49.5000	4.72%	50.3316	4.80%	1.68%
11	51.0000	4.86%	51.6058	4.92%	1.19%
12	64.0000	6.10%	62.9145	6.00%	-1.70%
13	49.0000	4.67%	49.0574	4.68%	0.12%
14	48.0000	4.58%	47.9425	4.57%	-0.12%
15	63.5000	6.05%	63.0738	6.01%	-0.67%
16	54.0000	5.15%	54.4728	5.19%	0.87%
17	57.0000	5.43%	57.0213	5.44%	$3.66 \cdot 10^{-2}\%$
18	53.0000	5.05%	52.0837	4.97%	-1.73%
19	45.5000	4.34%	41.8899	3.99%	-7.93%

Table 22: Exact Myerson value vs an approximated Myerson value for “graph-0.2”

Nodes	MV	MV ratio	AMV	AMV ratio	Rel. error AMV/MV
0	143.2672	6.53%	144.0222	6.56%	0.53%
1	68.7002	3.13%	49.3370	2.25%	-28.18%
2	68.0767	3.10%	81.6335	3.72%	19.91%
3	104.6799	4.77%	109.9763	5.01%	5.06%
4	164.0880	7.48%	157.1853	7.16%	-4.21%
5	114.6388	5.22%	106.1499	4.84%	-7.40%
6	16.7705	0.76%	17.8578	0.81%	6.48%
7	228.8444	10.43%	251.4307	11.46%	9.87%
8	33.7682	1.54%	24.3128	1.11%	-28.00%
9	36.1832	1.65%	28.1827	1.28%	-22.11%
10	184.1740	8.39%	192.6532	8.78%	4.60%
11	79.7901	3.64%	69.8602	3.18%	-12.45%
12	56.7906	2.59%	53.0066	2.42%	-6.66%
13	48.8678	2.23%	46.0495	2.10%	-5.77%
14	107.3196	4.89%	93.9069	4.28%	-12.50%
15	63.4039	2.89%	66.5861	3.03%	5.02%
16	186.6146	8.50%	200.2029	9.12%	7.28%
17	229.3778	10.45%	258.1879	11.77%	12.56%
18	162.6473	7.41%	152.4821	6.95%	-6.25%
19	96.4034	4.39%	91.3826	4.16%	-5.21%

trality, it has 100% accuracy for all cases except a case of a network “karate-twoStar-34”, for which the accuracy is 50%. The mistake in prediction of one of the most important nodes is that another node (Node 17) is the central one in the internal layer by construction. For the betweenness centrality, the accuracy is zero for high-density networks. Closeness centrality also has the worst accuracy for these networks.

Table 23: Accuracy defined by (32) for the proposed in this paper and classical centrality measures

graph \ accuracy	betweenness	closeness	Shapley value	Myerson value
karate-34	100%	50%	100%	100%
karate-empty-34	100%	50%	100%	100%
karate-karate-34	100%	50%	100%	100%
karate-star-34	100%	100%	100%	100%
karate-twoStar-34	100%	50%	100%	50%
karate-cycle-34	100%	100%	100%	100%
karate-twoClique-34	0%	0%	100%	100%
karate-complete-34	0%	0%	100%	100%

3.4.2 Experiments to examine correlation of network properties and opinion dynamics

We have done simulations of opinion dynamics in one-layer Zachary’s karate club network following BVM and two-layer network following GCVM with Zachary’s karate club network being an external layer and different internal network structures. There is a list of internal layers we use in our analysis:

1. **karate**: Zachary’s karate club network;
2. **star**: star structure with node 0 being the center;
3. **two-star**: two central nodes 0 and 17, nodes 1–16 are linked with node 0, nodes 18–33 are linked with node 17. Moreover, nodes 0 and 17 are linked;

4. **cycle**: node 0 is linked with node 1, node 1 is linked with node 2, and so on. Finally, node 33 is linked with node 0;
5. **two-clique**: nodes 0–16 belong to the first clique, nodes 17–33 — to the second clique, and these two cliques are connected through link between nodes 0 and 17;
6. **complete**: all nodes are linked with each other.

We start with simulations of opinion dynamics of BVM for one-layer network, and then CVM and GCVM for two-layer networks implementing different internal structures and examine their affect on consensus time and winning rate. Network properties (in this work, centrality measures) will also change with changes of the network structure.

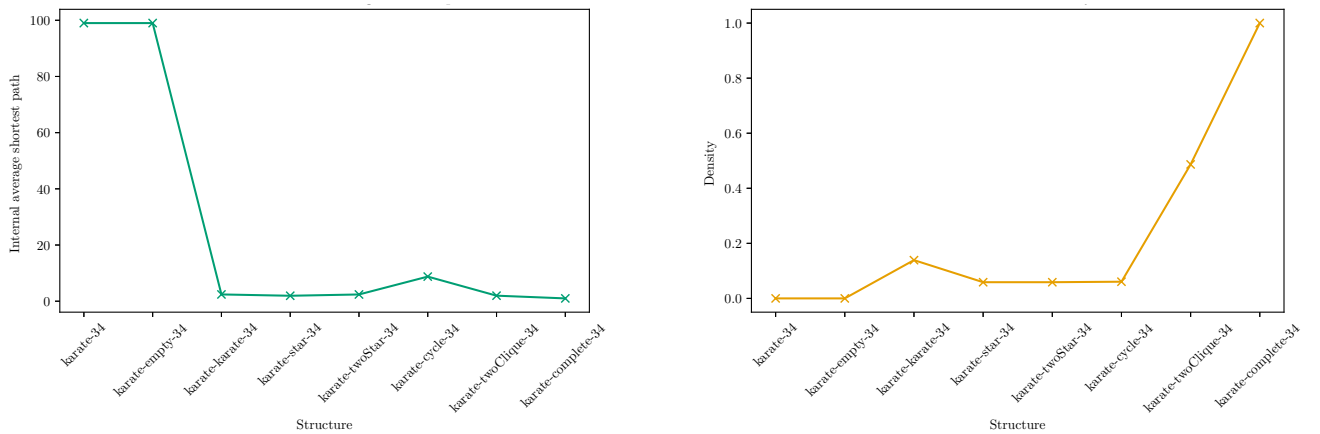
We describe the results of our experiments:

- Fig. 26a shows how internal average shortest path d_I varies depending on the network structure.¹²
- Fig. 26b shows how internal density varies depending on the network structure. By comparing Fig. 26a and 26b, we can notice that the internal density D_I and internal average shortest path d_I are negatively correlated.
- Fig. 27 shows different centralities for Nodes 0 and 33 for different network structures. In the left part of Fig. 27, we present the betweenness centrality, closeness centrality, approximated Shapley value, and approximated Myerson value for Nodes 0 and 33 on the simplified one-layer weighted network with weights calculated by formula (11). In the right part of Fig. 27, we present the group degree centrality, group closeness centrality, group betweenness centrality, and two different random walk centralities for the two-layer network with different internal structures. By comparing the centrality trend of Node 0 and Node 33 in the left and right parts of Fig. 27, respectively, we can observe that both have a similar trend, and this result demonstrates the validity of our approach of simplifying the two-layer network to a one-layer weighted network.

¹²“Karate-34” and “karate-empty-34” refer to a one-layer Zachary’s karate club network and to a two-layer network with Zachary’s karate club network in external layer and empty internal layer respectively, i.e. d_I does not exist for these two structures, in particular, it is equal to infinity. But in Fig. 26a, we replace infinity by number 99.

- Fig. 28 shows how KPIs vary with different network structures. Looking at Fig. 27, we may notice that some centralities trend of Node 33 are very similar to the trend in Fig. 26a. Fig. 26b also demonstrates the similar trend as in Fig. 28b.
- Fig. 28a shows that network structure has a great impact on winning rate.
- From Fig. 28b and 26b, we can notice that there exists a relationship between internal density D_I and consensus time T_{cons} . Networks with higher density, like “karate-complete-34”, take more time to reach consensus, while networks with less density, like “karate-empty-34”, reach consensus faster. This can be explained by the fact that networks with higher density have more connections which makes it more difficult for a single opinion to dominate quickly.

The main conclusions from the above results are: (i) there is a negative correlation between internal average shortest path d_I and both internal density D_I and consensus time T_{cons} , (ii) the approach of simplifying the two-layer network into a one-layer weighted network is valid.



(a) structure vs average shortest path

(b) structure vs density

Figure 26: Internal average shortest path and density for different network structures

Our next step is to examine dependence of network structure and KPIs of opinion dynamics. To do this, we conducted correlation tests on the above observation results by SciPy [57, 58]. We calculated three coefficients for each pair of characteristics: Pearson [59], Kendall [91] Spearman [92] correlation coefficients. They are presented in Table 24, where *, **, *** represent a level of significance 0.05, 0.01, 0.001 of the correlation coefficient, respectively.¹³

¹³We choose node 33 as the input for centrality. Actually, if we choose another node as an input, the conclusions are still valid.

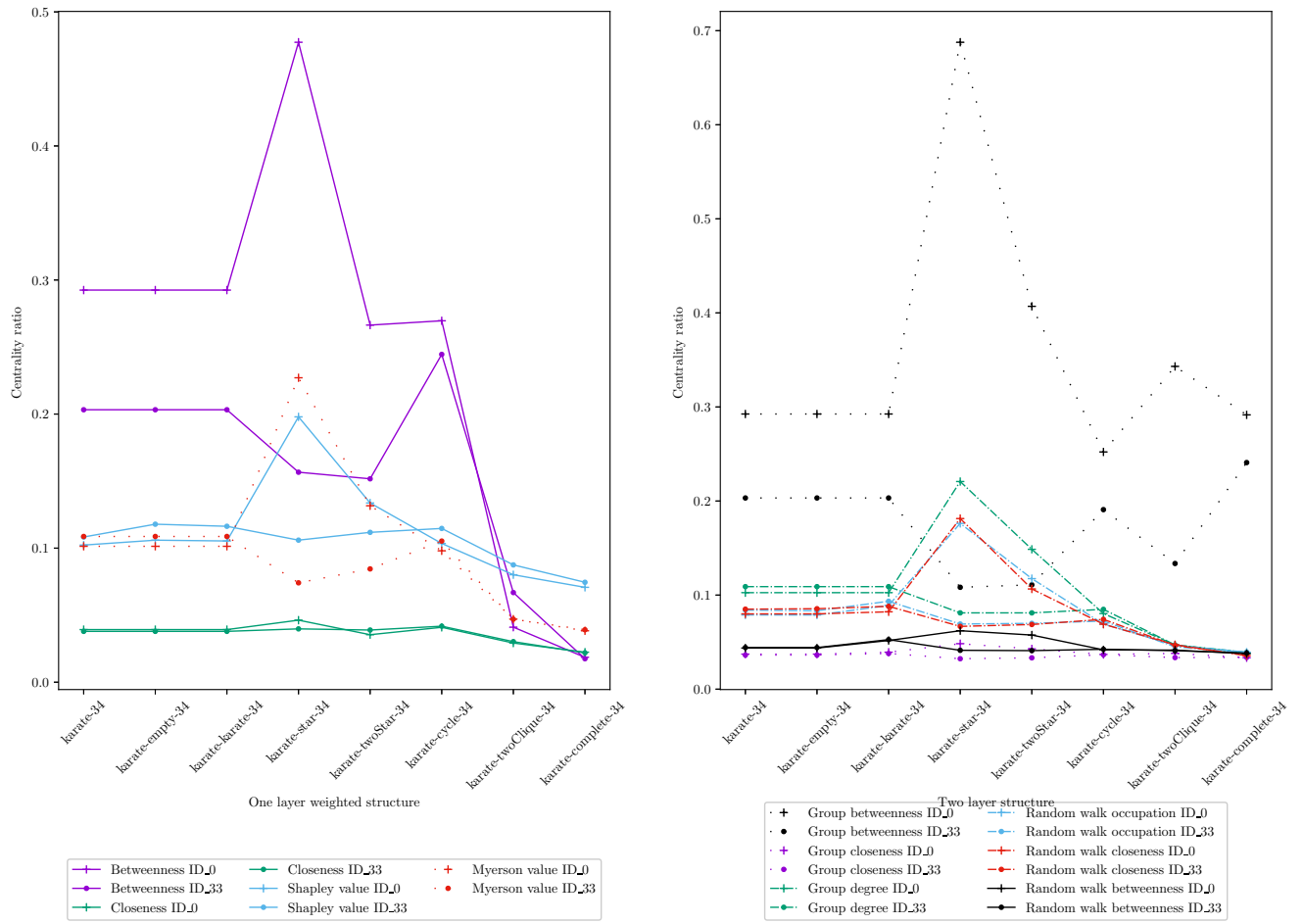
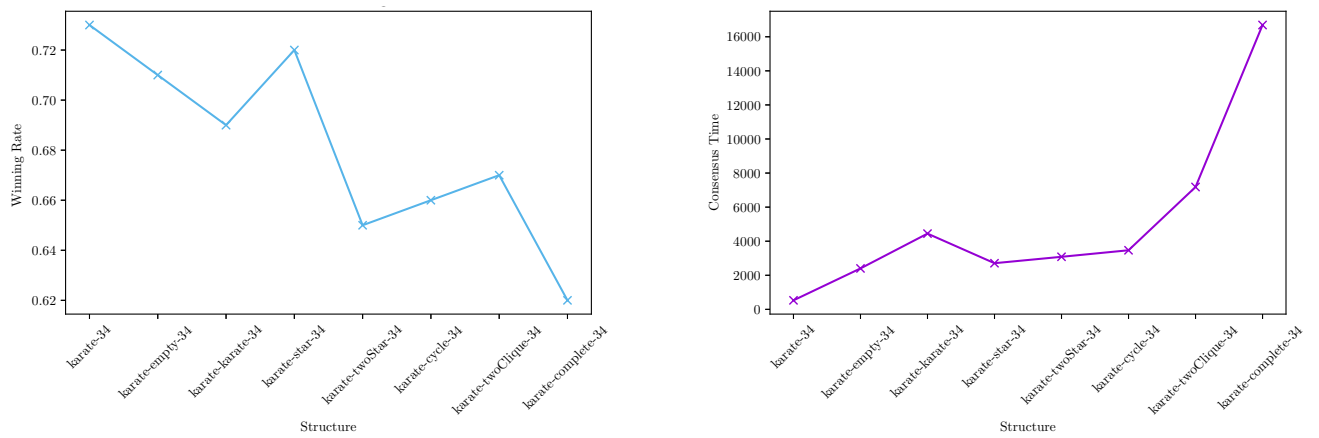


Figure 27: Different centralities for different structures



(a) structure vs winning rate

(b) structure vs consensus time

Figure 28: Winning rate and consensus time for different structures

We can make several conclusions from Table 24:

1. *High positive significant correlation:* (i) The internal density D_I exhibits a very strong highly significant positive correlation with consensus time across all correlation coefficients (Pearson, Kendall, and Spearman), they all are larger than 0.95. This indicates that when D_I increases, consensus time T_{cons} significantly increases; (ii) The internal average shortest path d_I is significantly positively correlated with most centrality measures which is confirmed by Kendall and Spearman correlation coefficients, they are significant.
2. *Negative significant correlation:* T_{cons} shows strong and significant negative correlation with network centrality measures like Betweenness and Closeness, especially with Closeness (with Pearson correlation coefficient equal to -0.928). So we can conclude that higher centrality scores are associated with shorter consensus time T_{cons} , which is expected.
3. *Variability in correlations:* Different metrics show varying levels of correlation strength across the Pearson, Kendall, and Spearman correlation coefficients. This variability indicates that the strength and significance of correlations can depend on the correlation method used, likely influenced by the underlying data distributions.

To sum up, Table 24 highlights significant relationships between specific network properties and characteristics of opinion dynamics like consensus time. Centrality of authoritative nodes and network density play a crucial role for consensus time.

Table 24: Correlation coefficients

	Pearson	Kendall	Spearman
d_I vs Betweenness	0.414	0.654*	0.810*
d_I vs Closeness	0.231	0.192	0.270
d_I vs Shapley value	0.363	0.618*	0.766*
d_I vs Myerson value	0.548	0.808**	0.908**
d_I vs Group betweenness	0.366	0.185	0.157
d_I vs Group closeness	0.408	0.333	0.614
d_I vs Group degree	0.610	0.830**	0.933***
d_I vs Random walk occupation	0.475	0.691*	0.826*
d_I vs Random walk closeness	0.553	0.691*	0.826*
d_I vs Random walk betweenness	0.161	0.618*	0.778*
T_{cons} vs Betweenness	-0.841**	-0.491	-0.537
T_{cons} vs Closeness	-0.928***	-0.340	-0.464
T_{cons} vs Shapley value	-0.872**	-0.286	-0.452
T_{cons} vs Myerson value	-0.791*	-0.491	-0.659
T_{cons} vs Group betweenness	0.407	0.255	0.299
T_{cons} vs Group closeness	-0.291	0.109	0.036
T_{cons} vs Group degree	-0.807*	-0.593*	-0.771*
T_{cons} vs Random walk occupation	-0.785*	-0.429	-0.548
T_{cons} vs Random walk closeness	-0.834*	-0.357	-0.524
T_{cons} vs Random walk betweenness	-0.424	-0.500	-0.571
D_I vs T_{cons}	0.983***	0.964**	0.988***

3.5 Conclusion to Chapter 3

In this chapter, we examined the correlation of several characteristics of opinion dynamics (including BVM, CVM, and GCVM) realized on two-layer networks with characteristics of these networks. As a network in one layer we consider a Zachary's karate club. We examined how internal network structure affects consen-

sus time and winning rate, and if these key performance indicators correlate with network centrality measures.

We proposed two fast and accurate algorithms to calculate centrality measures based on game-theoretic approach. Our algorithms can efficiently approximate the theoretical values of these measures for the networks, for which exact values are computationally difficult to find. Both of our algorithms can identify the most important nodes in the network, which is tested on different examples. The ideas of finding approximated centrality measures in the graphs implemented in our algorithms can be easily transferred to other fields, such as explainable artificial intelligence.

Chapter 4. ShapG: new feature importance method based on the Shapley value

In this chapter, we introduce one of the practical application of the developed centrality measures in Chapter 3 — explainable artificial intelligence. We consider the features in the machine learning models as nodes in a graph and consider centrality as feature importance. Based on the approximation algorithm of the Shapley value proposed in Chapter 3, we develop a new XAI method called ShapG. Most of results presented in this chapter are published in paper [52].

4.1 Explainable Artificial Intelligence (XAI) methods

In a general formulation, we assume that there is a sample of observations of features X_1, \dots, X_M which are used to construct a model $f(X_1, \dots, X_M)$ to explain/predict/classify a target variable Y . Due to nontransparency of black-box models in data analysis, the functional form of f is not known and there is a crucial requirement to estimate features importance in AI models applied for tabular data. There are three most popular explainable AI methods: feature importance, LIME (Local Interpretable Model-agnostic Explanations), and SHAP (SHapley Additive exPlanations). We do not describe them here but provide corresponding references where these methods are introduced or well described.

We use following methods in comparison with our novel XAI method described in Section 4.2.2:

- **Feature Importance:** Feature importance is a built-in method applied to tree models such as decision trees, random forests, gradient boosting trees, etc. When constructing a tree model, the algorithm automatically calculates the contribution of each feature to the model's prediction measuring the impact of each feature on the predicted outcome and evaluating its importance [93].

- **Permutation Feature Importance:** Permutation feature importance, a method for evaluating the importance of features, is proposed in [94]. If a feature is important, then the model performance will be greatly reduced when randomly shuffled, while if a feature is unimportant, then it will have very little impact on the model performance when it is randomly shuffled.
- **LIME:** It is a model-agnostic explanation method proposed in [47] and used to explain importance of variables in predictions of machine learning models [95]. The method is called agnostic if it is not specifically related to one particular machine learning method but can be applied to most of machine learning models.
- **SHAP (SHapley Additive exPlanations):** It is an explainable method developed based on cooperative game theory and proposed in [48]. SHAP provides a model-agnostic explanation mechanism that can theoretically be applied to any machine learning model.
- **KernelSHAP:** The method is based on LIME and the Shapley value. The method is using the following steps to simplify calculations: (i) generation of a random number of samples of features, (ii) defining the sample data for each subset of features in a special way, (iii) calculation of weights for each subset of features, and (iv) solution of a specially defined weighted least square optimization problem to find the vector of features importance, that is, an approximated Shapley value.
- **Sampling SHAP:** It computes Shapley values under the assumption of feature independence and it is an extension of the algorithm proposed in [96]. The calculations are based on a well-known alternative formulation of the Shapley value [97].

4.2 ShapG: a novel XAI method

4.2.1 The Shapley value

The cooperative game is defined by (\mathcal{M}, v) , where $\mathcal{M} = \{1, \dots, M\}$ is the set of players¹⁴ and $v : 2^{\mathcal{M}} \rightarrow \mathbb{R}$ is a characteristic function defining a “strength” of any coalition of players that is a subset of \mathcal{M} , i.e., for any coalition or collection of players $\mathcal{S} \subset \mathcal{M}$, $S = |\mathcal{S}|$. The value $v(\mathcal{S})$ represents the payoff or power of coalition \mathcal{S} . One of the main problems which the theory of cooperative games is solving is to find a “fair” allocation of the total payoff of the grand coalition $v(\mathcal{M})$ among its members. One imputation was proposed by Shapley [98] to allocate $v(\mathcal{M})$, and it is a vector $\boldsymbol{\phi} = (\phi_1, \dots, \phi_M)$, where ϕ_i is a payoff (part of $v(\mathcal{M})$) to player $i \in \mathcal{M}$ defined by

$$\phi_i = \sum_{\mathcal{S} : \mathcal{S} \subset \mathcal{M} \setminus \{i\}} \frac{(M - S - 1)! S!}{M!} (v(\mathcal{S} \cup \{i\}) - v(\mathcal{S})), \quad (33)$$

where $(v(\mathcal{S} \cup \{i\}) - v(\mathcal{S}))$ is a marginal contribution of player i if he joins coalition \mathcal{S} .

The vector with the components defined by (33) is called the Shapley value and it is a unique vector satisfying four axioms (efficiency, symmetry, null player, and additivity). The efficiency axiom means that the sum of the components of the Shapley value is equal to the payoff of grand coalition \mathcal{M} , i.e. $\sum_{i \in \mathcal{M}} \phi_i = v(\mathcal{M})$.

We also provide a probabilistic interpretation (see [99]) of the Shapley value to better understanding why this vector can be applied to measure the feature importance in complex machine learning models. Consider the i -th component of the Shapley value defined by (33). Player i gets the payoff $(v(\mathcal{S} \cup \{i\}) - v(\mathcal{S}))$ when he joins to the randomly formed coalition $\mathcal{S} : \mathcal{S} \subset \mathcal{M} \setminus \{i\}$. The probability that coalition \mathcal{S} containing S players is formed is equal to $\frac{1}{M} \binom{M-1}{S}$. It is assumed that all coalition \mathcal{S} 's sizes from 0 to $M - 1$ are equally probable and for a given coalition size S , the subsets of S players are also equally probable. Then the value ϕ_i given by (33) is player i 's expected payoff in such a probabilistic scheme.

¹⁴In the problem of measuring features importance, a feature is considered as a player in the game, so we use the same notation M for the number of players and number of features.

To make a connection between the Shapley value and the vector of feature importance, we can associate the set of players with the set of features and the characteristic function with the some quantitative characteristic of the prediction made by a machine learning model using subset of features. Then, the difference $(v(\mathcal{S} \cup \{i\}) - v(\mathcal{S}))$ can be interpreted as a contribution or payoff in the prediction quality if we add feature i to the subset of features \mathcal{S} . The expected value of such payoff is associated with feature i 's importance in a testing prediction ML model.

The main idea described in this section is borrowed from the theory of cooperative games and implemented in the SHAP method, but due to the complexity of the Shapley value calculation by formula (33) because the number of features and complexity of a prediction model f , the algorithms such as KernelSHAP and SamplingSHAP are proposed to approximate the Shapley value by reduction of the number of calculations [48, 100].

4.2.2 ShapG (explanations based on the Shapley value for graphs)

We describe a new XAI method called *ShapG* to calculate the feature importance in machine learning models based on the Shapley value defined on an undirected weighted graph constructed in a special way.

The Shapley value for undirected weighted graphs

The calculation of the Shapley value for an undirected weighted graph can be divided into following steps:

1. We define the undirected weighted graph $G = (\mathcal{M}, \mathcal{E})$, where \mathcal{M} is the set of nodes which are associated with features from set $\mathcal{M} = \{1, \dots, M\}$ and the set of edges \mathcal{E} without loops. The weight of an edge (j, k) , $j \neq k$, is equal to the Pearson correlation coefficient $W(j, k)$ between features j and k calculated by a given sample.
2. For any subset of features $\mathcal{S} \subset \mathcal{M}$ we define subgraph $G_{\mathcal{S}}$ of graph G .

3. For any subset of features $\mathcal{S} \subset \mathcal{M}$, we define the value of function v as follows:¹⁵

$$v(\mathcal{S}) = \sum_{\{j,k\} \subseteq G_{\mathcal{S}}} W(j,k). \quad (34)$$

4. We calculate the Shapley value by formula (33). As a result, the algorithm gives the Shapley value centrality for each node (feature).

Since the set of features \mathcal{M} may be large, we propose an approach for approximating the Shapley value with the high accuracy presented in [51], and the proposed method is based on calculation of weights of the edges and function (34), where \mathcal{S} is a subset of features from set \mathcal{M} , and $G_{\mathcal{S}}$ is the subgraph induced by \mathcal{S} . With a very minor modification, we can apply this approach to define a new XAI method presented in the next section.

Description and algorithm of the ShapG method

The ShapG method can be divided into following steps:

1. We define an undirected weighted graph $G = (\mathcal{M}, \mathcal{E})$, where \mathcal{M} is the set of nodes which are associated with the features $\mathcal{M} = \{1, \dots, M\}$ in the prediction model and \mathcal{E} is the set of all possible edges without loops, i.e. $\mathcal{E} = \{(i, j) : i \in \mathcal{M}, j \in \mathcal{M}, i \neq j\}$ is a complete graph without loops. The weight of an edge (j, k) , $j \neq k$, is equal to the Pearson correlation coefficient $W(j, k)$ between features j and k calculated on a given sample.
2. The matrix of weights $W = \{W(j, k)\}_{(j,k) \in \mathcal{E}}$ is usually a very dense matrix, therefore, we need to reduce the density of graph G to reduce the number of further calculations. We implement the idea of keeping all features of the dataset while minimizing the number of edges in the graph to reduce the density. The corresponding method is realized in Algorithm 4. The idea is straightforward: we construct graph G' starting from the empty graph by iteratively selecting the edges with largest Pearson correlation coefficients given in matrix W , and adding these edges into graph G' ensuring each node is included in G' at least once and graph G' is a connected graph (the latter

¹⁵The characteristic function (34) was proposed in the paper [39], the authors consider a cooperative game where the characteristic function is defined by the group degree centrality [77] of each coalition [36].

condition is a stopping rule in Algorithm 4). The output of Algorithm 4 is a new graph G' . In the following steps, we do not use matrix of weights W .¹⁶

3. In graph G' , we define subgraph G'_S for any subset of features $\mathcal{S} \subset \mathcal{M}$.
4. We define characteristic function $f(\mathcal{S})$ assigning the R^2 score (for regression models) or $F1$ score (for classification models) for any subset of features \mathcal{S} :

$$v(\mathcal{S}) = f(\mathcal{S}). \quad (35)$$

The characteristic function used for XAI can be defined by (35), where $f(\mathcal{S})$ is calculated as a prediction given by a ML model trained by only features from subset \mathcal{S} . For our purpose, we can use R^2 or $F1$ score as a measure of “power” of subset \mathcal{S} .

5. We calculate the Shapley value by formula (33). We use Algorithm 5 to find values of the Shapley value using exact formula (33). If the number of features is large, we use Algorithm 6 to find the approximated Shapley value.

Algorithm 4 Data preprocessing for ShapG method

Require: Dataset with M features

Ensure: The adjacency matrix $A \in \mathbb{R}^{M \times M}$.

- 1: Compute the Pearson correlation matrix W for all features. ▷ This is an initial matrix of weights for graph G
 - 2: Initialize the adjacency matrix A with zero matrix.
 - 3: $\mathcal{E} \leftarrow$ List of tuples $(i, j, W(i, j))$, where $i < j$.
 - 4: $\mathcal{E} \leftarrow$ Sort \mathcal{E} by $|W(i, j)|$ in descending order.
 - 5: Initialize index $k = 0$
 - 6: **while** G' is not connected **do** ▷ G' represented by A
 - 7: $i, j, weight = \mathcal{E}[k]$
 - 8: $A(i, j) \leftarrow 1$
 - 9: $A(j, i) \leftarrow 1$
 - 10: $k \leftarrow k + 1$
 - 11: **end while**
- return** The adjacency matrix A to represent the feature graph.
-

¹⁶We only use weights to reduce graph G to G' , we do not use it to calculate the Shapley value.

The Algorithm 5 describes the calculation of the Shapley value of features $1, \dots, M$ based on characteristic function defined by (35) following the steps described above.¹⁷

Algorithm 5 Calculation of the Shapley Value based on graph G'

Require: A graph $G'(\mathcal{M}, \mathcal{E})$ with $M = |\mathcal{M}|$ nodes

Ensure: Shapley value component $\phi(i)$ for each node $i \in \mathcal{M}$

```

1: for all nodes  $i \in \mathcal{M}$  do
2:   Initialize  $\phi(i) \leftarrow 0$ 
3: end for
4: for all nodes  $i \in \mathcal{M}$  do
5:   for all subsets  $\mathcal{S} \subseteq \mathcal{M} \setminus \{i\}$  do
6:     Compute  $v(\mathcal{S}) \leftarrow f(\mathcal{S})$ 
7:     Compute  $v(\mathcal{S} \cup \{i\}) \leftarrow f(\mathcal{S} \cup \{i\})$ 
8:      $\Delta v(\mathcal{S}, i) \leftarrow v(\mathcal{S} \cup \{i\}) - v(\mathcal{S})$ 
9:      $\text{coeff} \leftarrow \frac{S!(M-S-1)!}{M!}$ 
10:     $\phi(i) \leftarrow \phi(i) + \text{coeff} \cdot \Delta v(\mathcal{S}, i)$ 
11:   end for
12: end for
return  $\phi(i)$  for all  $i \in \mathcal{M}$ 

```

However, the Shapley value is pretty computationally expensive, especially, for a large number of features and consequently large number of feature subsets. Therefore, we provide a modified algorithm based on the fact that an influence on a particular node from other nodes is decreasing with an increase of the path length connecting them. Algorithm 6 can be used instead of Algorithm 5 having G' as an input. Algorithm 6 implements the following ideas to speed up the calculation of the Shapley value:

1. *Depth limitation:* By limiting the depth of the reachable nodes considered, the number of subsets that need to be considered is reduced.

We set parameter d_{max} which is the depth in the graph for any node $i \in \mathcal{M}$ to form the set of reachable nodes. Define $\psi(i, d_{max})$ as the set of reachable nodes of node i up to depth d_{max} excluding node i . Then we calculate the Shapley value of a node/feature i based on $\psi(i, d_{max})$ by equation (25), where $\beta = \frac{|\psi(i, d_{max})|+1}{m+1}$ is a scaling factor, and m is the maximal number of reachable nodes considered.

¹⁷For this algorithm, it's not necessary to reduce the graph density, because regardless of the structure of the graph, Algorithm 5 always needs to traverse all possible coalitions. Algorithm 5 needs to iterate for all possible coalitions, so for graph G and graph G' , the number of iterations is the same. For Algorithm 6, we do not need to consider all the coalitions, but only any node and its reachable nodes. Therefore, Algorithm 5 is graph independent, but Algorithm 6 is graph dependent.

2. *Local subset iteration*: Iterating over subsets only within the reachable nodes, rather than the entire graph, decreases the number of iterations. To be specific, we calculate only the marginal contribution of the current node to its reachable nodes from set $\psi(i, d_{max})$, where d_{max} is set (see Item 1).

In Algorithm 6, we use formula (35) to define the values of the characteristic function, but we limit the depth of the reachable nodes considered (Item 2) to reduce the number of calculations.

3. *Sampling from reachable nodes*: For a large number of reachable nodes, computational complexity can be reduced by random sampling, thereby decreasing the number of subsets iterated over.

When $|\psi(i, d_{max})| \geq m$, we choose a random sample of m nodes from the set $\psi(i, d_{max})$ several times and calculate the Shapley value based on these samples. The sampling time $H_{|\psi(i, d_{max})|, m}$ is given by formula (26). The value $H_{|\psi(i, d_{max})|, m}$ is the mathematical expectation of the number of samples each time collecting m nodes from the set $\psi(i, d_{max})$ until all reachable nodes are collected.¹⁸ As shown in Algorithm 6, we repeat sampling process $H_{|\psi(i, d_{max})|, m}$ times and then take the average value of the Shapley values.

Items 1–3 from this list reduce the number of calculations and the computational complexity of an algorithm. Based on these three items, the modified Algorithm 6 of the ShapG¹⁹ method is presented.

¹⁸This probabilistic scenario is known as a generalized coupon collector’s problem, which was introduced and examined in [87].

¹⁹The corresponding code for ShapG can be found in the GitHub repository at <https://github.com/vectorsss/shapG>.

Algorithm 6 Approximated Shapley Value based on graph of features

Require: A graph $G' = (\mathcal{M}, \mathcal{E})$, depth limit d_{max} , maximal size m of the set of reachable considered

Ensure: Shapley value $\phi_a(i)$ for each node $i \in \mathcal{M}$

```

1: Initialize  $\phi(i)_a \leftarrow 0$  for each  $i \in \mathcal{M}$ 
2: for  $i \in \mathcal{M}$  do
3:    $\psi(i, d_{max}) \leftarrow$  Calculate or retrieve all reachable nodes of  $i$  up to depth  $d_{max}$ 
4:   if  $|\psi(i, d_{max})| < m$  then
5:     for each subset  $\mathcal{S} \subseteq \psi(i, d_{max}) \setminus \{i\}$  do
6:       Compute  $v(\mathcal{S}) \leftarrow f(\mathcal{S})$ 
7:       Compute  $v(\mathcal{S} \cup \{i\}) \leftarrow f(\mathcal{S} \cup \{i\})$ 
8:        $\Delta v(\mathcal{S}, i) \leftarrow v(\mathcal{S} \cup \{i\}) - v(\mathcal{S})$ 
9:        $\phi_a(i) \leftarrow \phi_a(i) + \Delta v(\mathcal{S}, i)$ 
10:    end for
11:     $\text{coeff} \leftarrow \frac{1}{2^{|\psi(i, d_{max})|}}$ 
12:     $\phi_a(i) \leftarrow \phi_a(i) \cdot \text{coeff}$ , normalize  $\phi_a(i)$  based on the number of subsets
13:  else
14:    Pick up  $m$  nodes randomly from  $\psi(i, d_{max})$  and repeat  $H_{|\psi(i, d_{max})|, m}$  times
15:    for  $i = 1$  to  $H_{|\psi(i, d_{max})|, m}$  do
16:       $s_{reachable} \leftarrow$  Randomly select a sample of  $m$  nodes from  $\psi(i, d_{max})$ ,
17:      for each subset  $\mathcal{S} \subseteq s_{reachable} \setminus \{i\}$  do
18:        Calculate  $v(\mathcal{S})$  and  $v(\mathcal{S} \cup \{i\})$  as before
19:         $\Delta v(\mathcal{S}, i) \leftarrow v(\mathcal{S} \cup \{i\}) - v(\mathcal{S})$ 
20:         $\phi_a(i) \leftarrow \phi_a(i) + \Delta v(\mathcal{S}, i)$ 
21:      end for
22:    end for
23:     $\text{coeff} \leftarrow 1/2^{|\psi(i, d_{max})|} / H_{|\psi(i, d_{max})|, m} \cdot \frac{|\psi(i, d_{max})|+1}{m+1}$ 
24:     $\phi_a(i) \leftarrow \phi_a(i) \cdot \text{coeff}$ 
25:  end if
26: end for
27: return  $\phi_a(i)$  for all  $i \in \mathcal{M}$ 

```

4.3 Experiments

4.3.1 Description of datasets

To demonstrate the work of our XAI method ShapG, we consider two datasets to construct the prediction models: (i) the “housing price” dataset for regression prediction, and (ii) the “H1N1 flu vaccine” dataset for classification prediction.

We briefly describe datasets:

1. The “housing price” dataset was collected by the U.S. Census Bureau for housing information in the Boston, Massachusetts area. The dataset contains 13 features including “per capita crime rate by town”, “average number of rooms per dwelling”, and “lower status of the population”, as well as a target variable, that is, the “median value of owner-occupied homes.”
2. The “H1N1 flu vaccine” dataset is provided by the National Center for Health Statistics and borrowed from the DrivenData website. The dataset contains 35 features including an “individual’s age”, “gender”, “education level”, and “knowledge of the H1N1 flu vaccine”, as well as a target variable, that is, a binary value of whether or not an individual received the H1N1 flu vaccine.

These two datasets are used for different prediction tasks, but they are both modeled and predicted by machine learning algorithms. For the regression task, we use R^2 to define the characteristic function (35) in our ShapG XAI method, while for classification task, we use $F1$ score to define the values of characteristic function (35).

We should mention that the “housing price” dataset has much less number of features, than the “H1N1 flu vaccine” dataset. We will compare our ShapG method with other XAI methods not only quantitatively comparing their explanations but also comparing running time. Such a comparison helps to evaluate efficiency of different XAI methods in processing large-scale data.

4.3.2 Preprocessing data for ShapG

We follow Algorithm 4 to create graph G' based on the original complete graph connecting nodes representing features. Algorithm 4 starts from the empty graph and consequently adds the edges, these are the pairs of features with the strongest correlation coefficients. It stops when all features are connected (graph G' should be connected), ensuring that the feature graph has important structural information. Figures 29 and 30 represent the feature correlation coefficients heatmaps for the “housing price” and “H1N1” datasets, respectively. On both figures, figure (a) shows the original correlation coefficient heatmap, and figure (b) — the heatmap of correlation coefficients between features in the reduced graph G' computed by Algorithm 4.

We highlight that we use Algorithm 4 to reduce the number of edges while preserving all features, thus ignoring “unimportant” relationships between features when constructing graph G' . This will reduce the number of iterations to compute the components of the Shapley value measuring importance of the features.

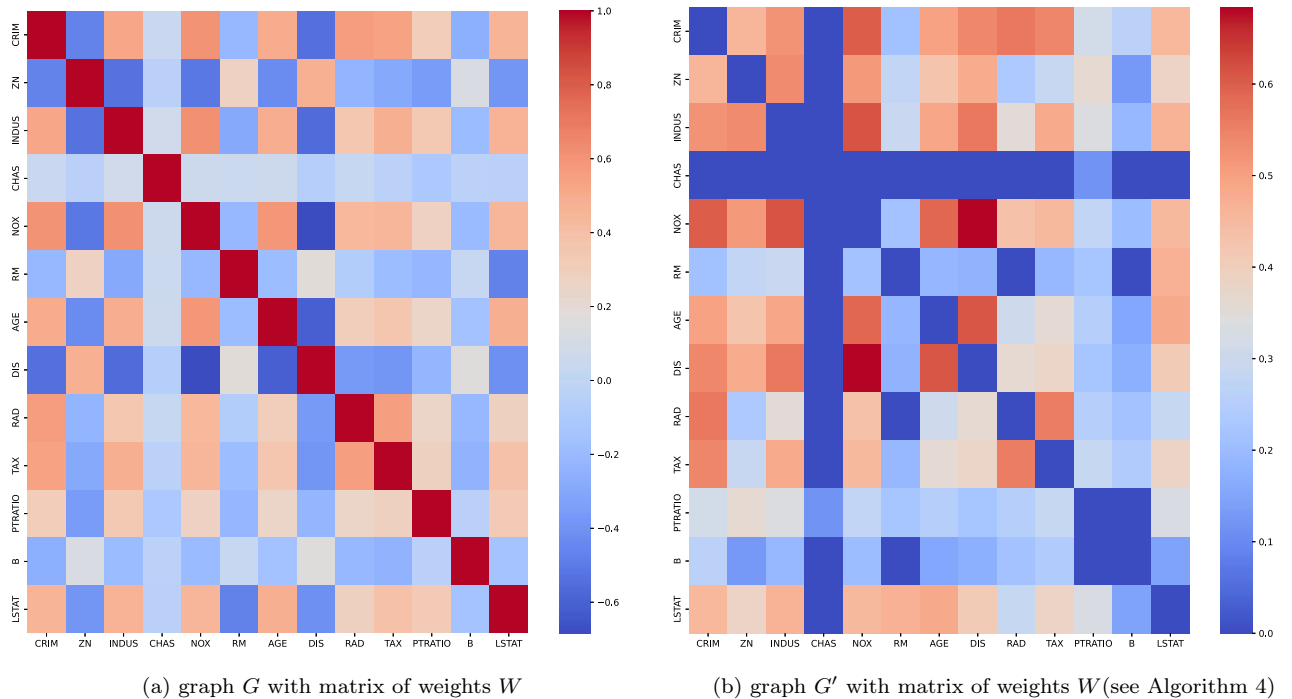
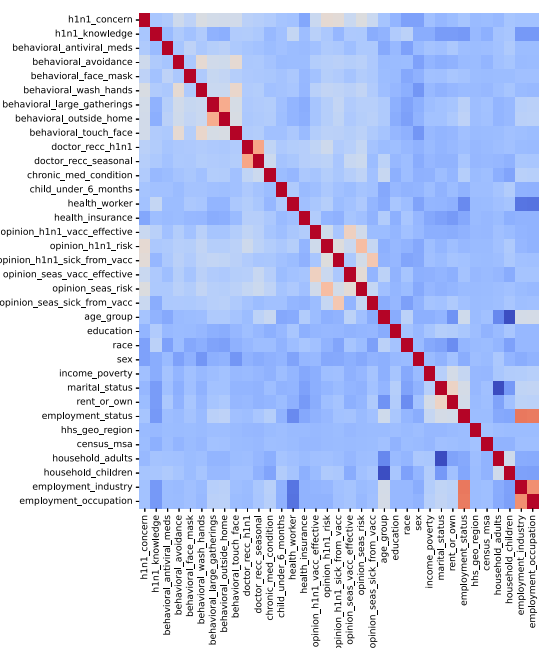
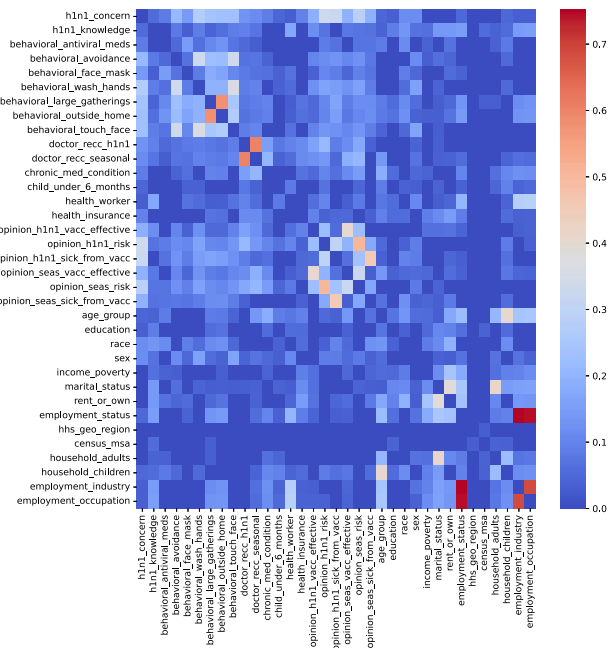


Figure 29: Heatmap of Pearson correlation coefficients for the “housing price” dataset

The original undirected graph G with the nodes representing features of the “housing price” (“H1N1”) dataset is given in Figure 31a (Figure 32a), while Figure 31b (Figure 32b) shows the undirected graph G' calculated by Algorithm 4. So,



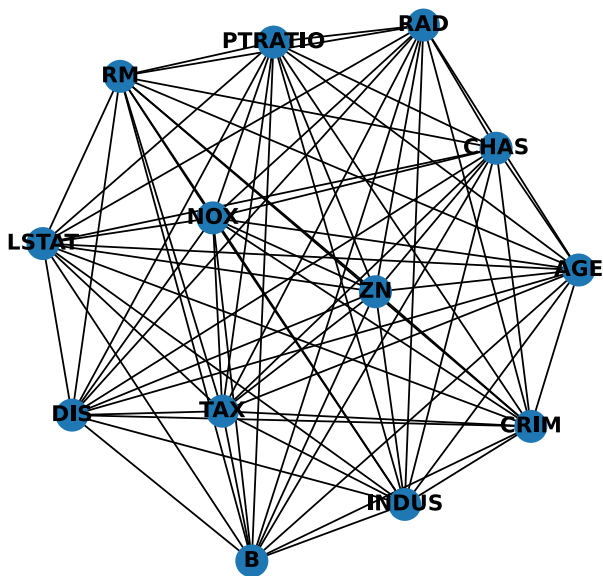
(a) graph G with matrix of weights W



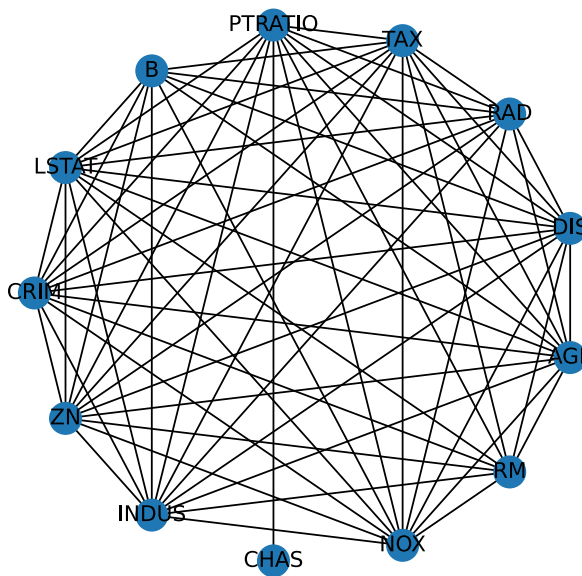
(b) graph G' with matrix of weights W (see Algorithm 4)

Figure 30: Heatmap of Pearson correlation coefficients for the “H1N1” dataset

we simplify graph structures and keep important feature pairs improving running efficiency and explainability of the ShapG method.



(a) Original graph G



(b) Reduced graph G' computed by Algorithm 4

Figure 31: Graph connecting features in “housing price” dataset

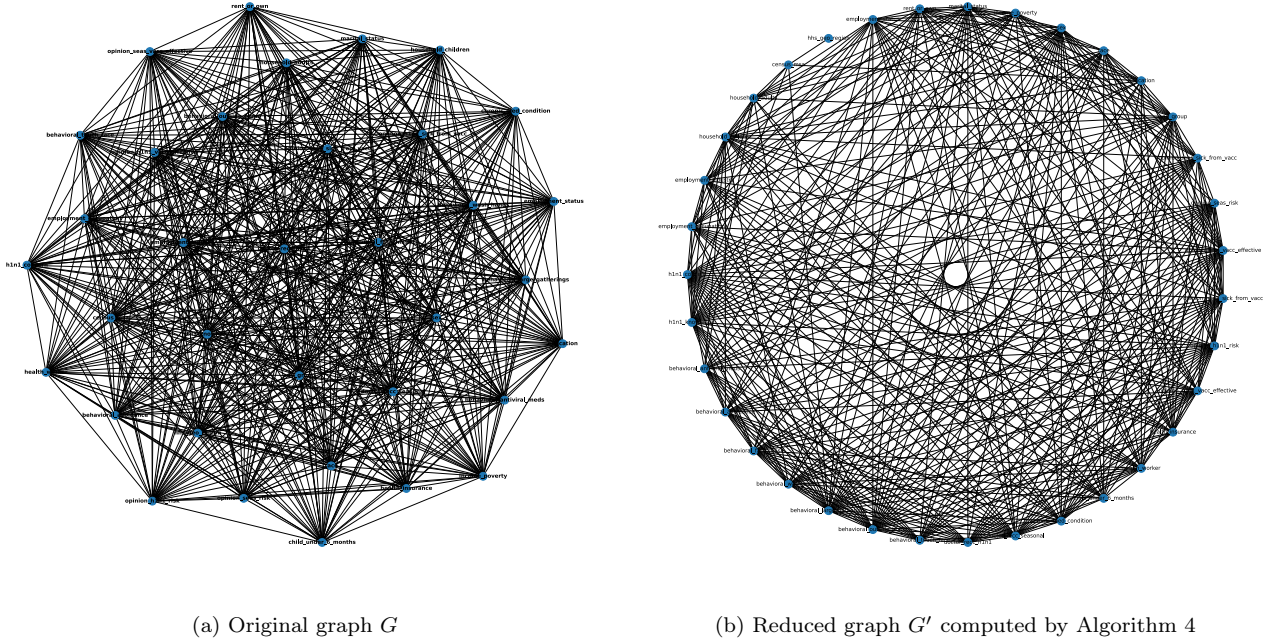


Figure 32: Graph connecting features in “H1N1” dataset

4.3.3 AI prediction models

To evaluate efficiency of our method ShapG, we apply several Explainable Artificial Intelligence (XAI) methods including our method to explain the importance of features within LightGBM and MLP (Multilayer Perceptron) models. Moreover, our ShapG algorithm can provide explanations for complex AI models that existing XAI methods cannot explain in a reasonable running time. We adopt ensemble learning and a two-by-two combination of tree models, neural network models, linear models, machine learning models to construct hybrid prediction models. We construct these hybrid models as combination of different types of simpler models to achieve better performance and explainability. There is a list of constructed models (including their combinations) and their descriptions:

- **LightGBM (LGB):** It is an efficient Gradient Boosting Decision Tree (GBDT) framework, which has fast training speed and high performance. It is widely used in practice [101].
- **Multilayer Perceptron (MLP):** It is a feed-forward artificial neural network consisting of fully connected neurons with nonlinear activation functions [102]. Through several tests, we adjust the parameters to obtain the best model prediction. In the regression model (for “housing price” dataset), a

hidden layer containing 300 neurons is used, the learning rate is set to 0.007, the activation function is chosen as ReLU (Rectified Linear Unit), and the optimization algorithm is Adam (Adaptive Moment Estimation). These parameters achieve the best accuracy of prediction for “housing price” dataset. In the classification model (for “H1N1” dataset), to obtain a high performance, the model has two hidden layers, the first one is with 100 neurons, and the second one is with 50 neurons. The model performs up to 3 iterations to complete training dataset, and the activation function is also chosen as ReLU.

- **Ensemble Learning (Stacking):** It is an approach to get better predictive performance by combining several single models [103]. Stacking is one of the ensemble learning methods, where first several different types of base models are trained using a training sample, and second, meta model is trained using the predictions of base models as input features in combination with real labels [104]. In this paper, we choose Random Forest and XGBoost as basic models, and LightGBM as the meta model.
- **Linear Regression – LightGBM (Linear - LGB):** It is a classic linear regression model and the more representative LightGBM model for regression prediction on the “housing price” dataset. Linear models effectively capture linear relationships between features [105], while tree models are able to handle non-linear relationships and high-dimensional features.
- **Logistic Regression – LightGBM (Logistic - LGB):** Logistic regression is a common linear model used for classification problems [106]. For “H1N1” dataset, we combined logistic regression and LightGBM for classification.
- **Linear Regression – Multilayer Perceptron (Linear - MLP):** Multilayer Perceptron (MLP) is an artificial neural network (ANN) consisting of multiple layers of interconnected neurons with ability to process various data types [107]. We combined linear regression model and neural network model – MLP to construct a regression prediction model.
- **Logistic Regression – Multilayer Perceptron (Logistic - MLP):** We combine Logistic Regression and MLP to make a classification model. Using logistic regression to initially categorize the input data, we then apply MLP to capture more complex patterns and nonlinear relationships. It is suitable for dealing with a variety of complex classification problems.

- **Linear Regression – K-Nearest Neighbors (Linear - KNN):** KNN is a common machine learning algorithm that is widely used to solve classification and regression problems [108]. The Linear-KNN model can be used to predict both linear and nonlinear relationships when dealing with regression problems, improving the flexibility and accuracy of the model.
- **Logistic Regression – K-Nearest Neighbors (Logistic - KNN):** We combine logistic regression with KNN algorithm, and this hybrid model is able to utilize both the linear relationship of logistic regression and nonparametric properties of KNN when dealing with classification problems.
- **Multilayer Perceptron – LightGBM (MLP - LGB):** A hybrid model of Multilayer Perceptron (MLP) neural networks and LightGBM can simultaneously utilize the nonlinear fitting ability of neural networks and efficient performance of gradient boosted decision trees.
- **K-Nearest Neighbors – LightGBM (KNN - LGB):** The hybrid model of KNN and LightGBM gives an advantage of the high performance and efficiency of LightGBM while exploiting nonparametric properties of KNN.
- **Multilayer Perceptron – K-Nearest Neighbors (MLP - KNN):** We combine a multilayer perceptron (MLP) neural network with KNN algorithm, which is capable of exploiting both the nonlinear fitting ability of the neural network and nonparametric properties of KNN to provide better classification or regression performance.

The goal of the hybrid models is to combine different types of models for better performance and explainability. Our proposed XAI algorithm explains these complex AI models and demonstrates its broad applicability. This means that it can be applied to a wide range of complex hybrid models and is not limited by specific model types. Therefore, it is model-agnostic.

We first calculate feature importance for LightGBM and MLP models for two datasets in Section 4.4.1, and then for more complex models presented in the above given list in Section 4.4.3.

4.3.4 Evaluation of XAI methods

A common way of evaluating XAI methods is by performing a perturbation analysis of the features, removing features in order of their importance from the largest to the smallest, and observing a decrease of accuracy or R^2 of the model [109].

When we consequently remove features based on their importance, if performance of a model decreases significantly, it indicates that the feature is very important. By evaluating the XAI method in this way, we can understand the contribution of each feature to the model’s predictive performance.

In order to prove an efficiency of our ShapG method, we can use this evaluation method to compare ShapG with existing popular XAI methods. First, we will apply our proposed XAI approach and other popular XAI methods to generate explanation results. Then, we will remove features gradually according to their importance and observe the changes in the model’s performance. We compare this process using the following XAI methods: Feature Importance, Permutation Feature Importance, LIME, SHAP (KernelSHAP and SamplingSHAP), and ShapG.

4.4 Results and analysis

4.4.1 Feature importance calculated by ShapG

“Housing price” dataset

Figure 33 shows the feature importance calculated by ShapG algorithm for the “housing price” dataset based on LightGBM (Fig. 33a) and MLP (Fig. 33b) models, respectively. For the “housing price” dataset with regression prediction based on LightGBM, the four most important features given by ShapG are “LSTAT (lower status of the population)”, “RM (average number of rooms per dwelling)”, “NOX (nitric oxides concentration)”, “PTRATIO (pupil-teacher ratio by town)”, while with

regression prediction based on MLP model, these features are “LSTAT (lower status of the population)”, “B (the proportion of blacks by town)”, “RM (average number of rooms per dwelling)”, “PTRATIO (pupil-teacher ratio by town)”. The three features are the same for both models, and the most important one is the unique for both models too.

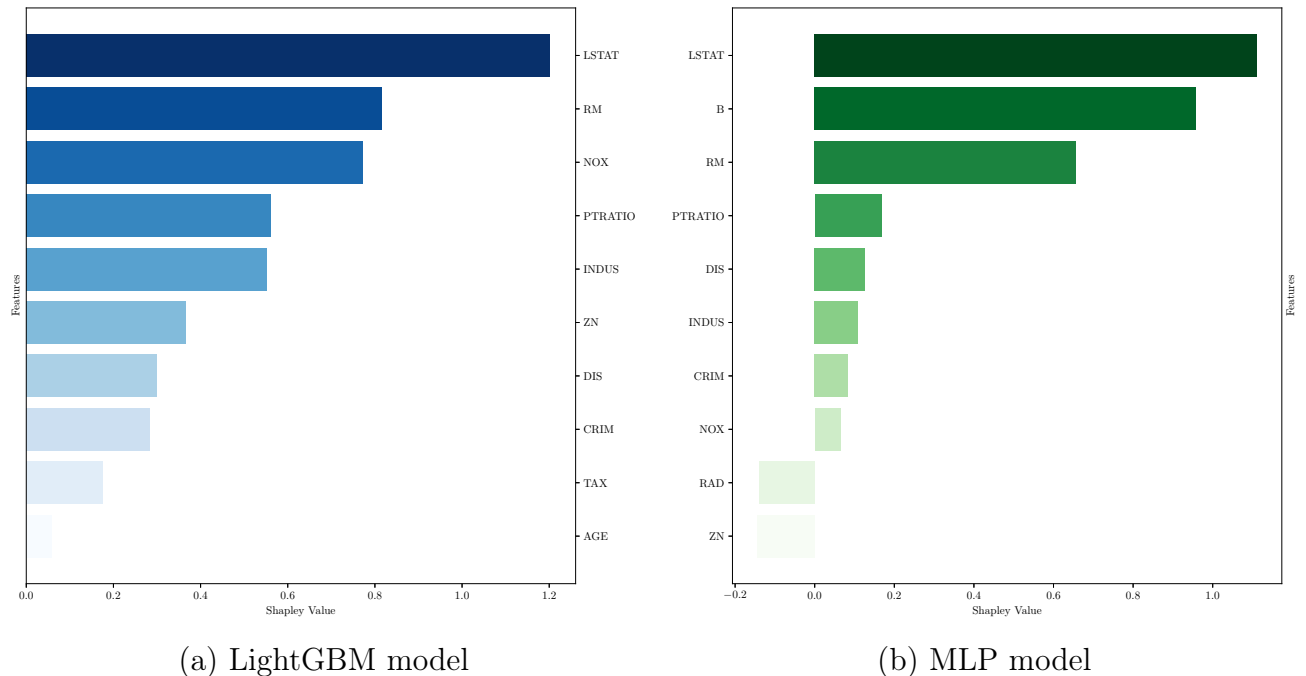


Figure 33: Feature importance in “housing price” dataset calculated with ShapG

“H1N1” dataset

Figures 34 shows the feature importance calculated by ShapG algorithm for the “H1N1” dataset based on LightGBM (Fig. 34a) and MLP (Fig. 34b) models, respectively. For the “H1N1” dataset with classification prediction, the feature’s approximated Shapley values given by ShapG algorithm represent the importance of each feature for people’s willingness to be vaccinated against H1N1. For classification based on LightGBM, the five most important features are “doctor recc h1n1 (H1N1 flu vaccine was recommended by doctor)”, “opinion h1n1 risk (Respondent’s opinion about risk of getting sick with H1N1 flu without vaccine)”, “health insurance”, “opinion h1n1 vacc effective (Respondent’s opinion about H1N1 vaccine effectiveness)”, “employment occupation (Type of occupation of respondent)”.

For classification based on MLP model, the five most important features are “doctor recc h1n1 (H1N1 flu vaccine was recommended by doctor)”, “opinion h1n1 risk (Respondent’s opinion about risk of getting sick with H1N1 flu without vac-

cine)”, “doctor recc seasonal (Seasonal flu vaccine was recommended by doctor)”, “health insurance”, “health worker”. Three most important features for both LightGBM and MLP models coincide.

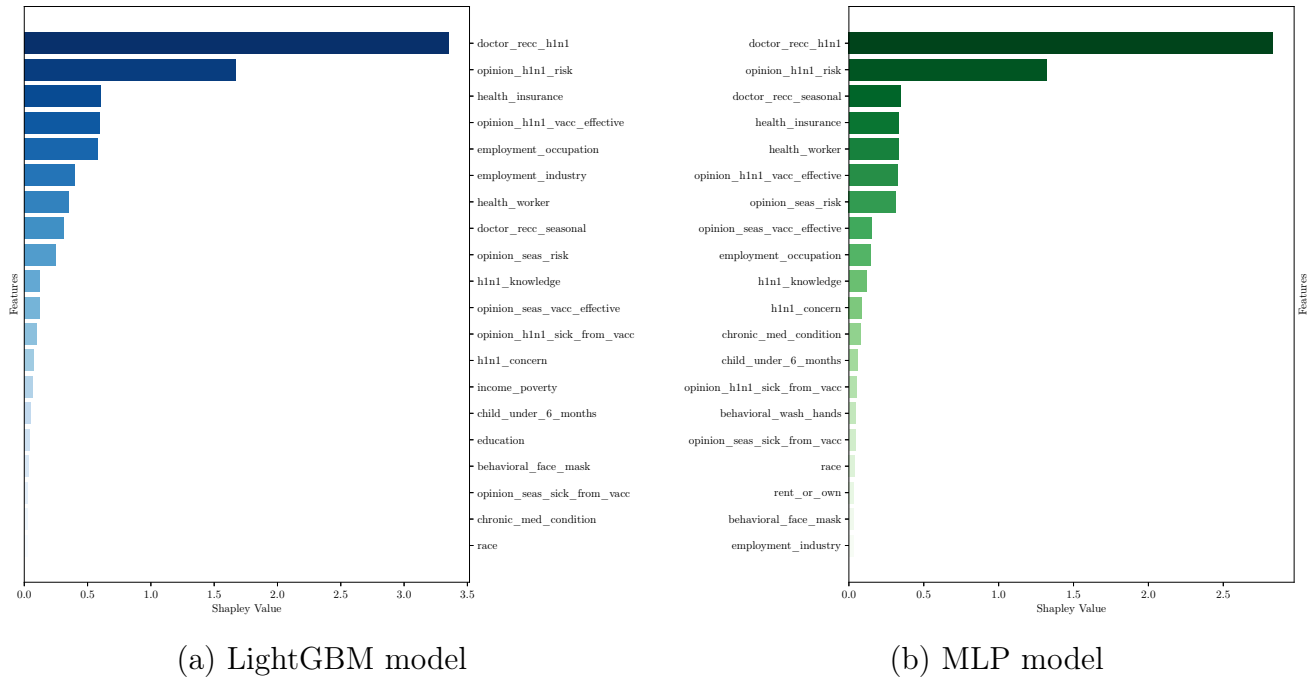


Figure 34: Feature importance in “H1N1” dataset calculated with ShapG

Based on the XAI approach to explain different AI models, it produces different explanation results since different model architectures may have differences in processing data, extracting features and predictions, this leads to differences in the result explanation. When both AI models (LightGBM and MLP) give the same most important features, we can make better decisions and understand the behavior of “black-box” models. As mentioned above, for the “housing price” dataset, we can observe that features “LSTAT”, “RM”, and “PTRATIO” are considered to be the most important features for both LightGBM and MLP models to predict house prices. For “H1N1” dataset, such three features are “doctor recc h1n1”, “opinion h1n1 risk (Respondent’s opinion about risk of getting sick with H1N1 flu without vaccine)”, and “health insurance”, these are considered to be the most important ones to predict if a person is willing to be vaccinated thereby developing relevant strategies to increase vaccination rates.

4.4.2 Evaluation of XAI methods

In order to prove efficiency of our proposed XAI method ShapG, we compare the results of its work with other existing XAI methods by (i) evaluating all methods based on perturbation of features, and (ii) measuring running time to obtain results.

We introduce the feature numbering for “housing price” and “H1N1” in Tables 25 and 26, respectively.

Table 25: Feature No. in “house price” dataset

No.	Feature	No.	Feature	No.	Feature
1	CRIM	6	RM	11	PTRATIO
2	ZN	7	AGE	12	B
3	INDUS	8	DIS	13	LSTAT
4	CHAS	9	RAD		
5	NOX	10	TAX		

Table 26: Feature No. in “H1N1” dataset

No.	Feature	No.	Feature
1	h1n1 concern	19	opinion seas vacc effective
2	h1n1 knowledge	20	opinion seas risk
3	behavioral antiviral meds	21	opinion seas sick from vacc
4	behavioral avoidance	22	age group
5	behavioral face mask	23	education
6	behavioral wash hands	24	race
7	behavioral large gatherings	25	sex
8	behavioral outside home	26	income poverty
9	behavioral touch face	27	marital status
10	doctor recc h1n1	28	rent or own
11	doctor recc seasonal	29	employment status
12	chronic med condition	30	hhs geo region
13	child under 6 months	31	census msa
14	health worker	32	household adults
15	health insurance	33	household children
16	opinion h1n1 vacc effective	34	employment industry
17	opinion h1n1 risk	35	employment occupation
18	opinion h1n1 sick from vacc		

Table 27 shows the results (ranking of feature importance) given by different XAI methods for “housing price” dataset based on LightGBM and MLP models.

We observe that SamplingSHAP and KernelSHAP give the same results for both LightGBM and MLP models, so they are combined into one column. As we can see in Table 27, for the LightGBM model for the “house price” dataset, all XAI methods have the same ranking of the first and second most important features, “LSTAT (lower status of the population)” and “RM (average number of rooms per dwelling)”. The XAI methods applied for the MLP model including SHAP, SamplingSHAP, KernelSHAP, and ShapG, give the same ranking on the first and second most important features “LSTAT” and “B (the proportion of blacks by town)”, while the ranking given by LIME is the opposite. In addition, the PFI method already disagrees with the results of other XAI methods in the ranking of the second feature. We use the XAI evaluation method to compare performance of these XAI methods, and comparison results are shown in Figures 35 and 36.

Table 27: Feature importance ranking for “housing price” dataset

LightGBM model

Rank \ XAI	SHAP	SamplingSHAP (KernelSHAP)	LIME	Feature Importance	ShapG
Top 1	13	13	13	13	13
Top 2	6	6	6	6	6
Top 3	8	7	7	1	5
Top 4	7	8	11	8	11
Top 5	5	5	10	7	3
Top 6	1	1	12	11	2
Top 7	11	11	4	12	8

MLP model

Rank \ XAI	SHAP	SamplingSHAP (KernelSHAP)	LIME	Permutation Feature Importance	ShapG
Top 1	13	13	12	13	13
Top 2	12	12	13	7	12
Top 3	7	7	11	12	6
Top 4	2	9	3	2	11
Top 5	9	2	1	9	8
Top 6	8	8	7	8	3
Top 7	6	6	6	6	1

Table 28 shows the results (ranking of feature importance) of different XAI methods for LightGBM and MLP models constructed for “H1N1” dataset. We should mention that we do not use KernelSHAP to get explanation results due to its very large running time: it requires more than 72 hours for LightGBM and more than 654 hours for MLP model. As we can see in Table 28, SamplingSHAP, LIME, and ShapG give the same ranking for the most important feature “doctor recc h1n1 (10)” for the LightGBM model for “H1N1” dataset. SHAP, SamplingSHAP, and ShapG rank the first four features in different orders, but the set of these features is the same: “doctor

recc h1n1 (10)”, “opinion h1n1 risk (17)”, “health insurance (15)”, and “opinion h1n1 vacc effective (16)”. However, starting from the fifth feature, the feature rankings significantly differ for different XAI methods. Meanwhile, explanation of the FI method is significantly different from the results of other XAI methods. For the MLP model, we can see that SHAP, SamplingSHAP, LIME, and Permutation Feature Importance methods rank feature “opinion h1n1 risk (17)” as the most important feature, while this feature is ranked as the second by ShapG. Due to differences in explanations, it becomes extremely important to evaluate and compare these XAI methods. The comparison results are shown in Figures 37 and 38.

Remark 1. *We should highlight that for LightGBM model we use Feature importance (FI) XAI method contrary to Perturbation Feature Importance (PFI) used for MLP model. The reason is as follows. FI is an explanation method built in tree models and it is widely used to explain tree-based models. However, for MLP models, which do not have tree structures to directly compute feature importance. Therefore, the impact of features in MLP models can be estimated using PFI more efficiently. Since the PFI method is independent of the specific model, it can be widely used in various types of models to replace FI method in calculating feature importance. Thus, in our experiments we use FI in the LightGBM model and PFI in the MLP model for both datasets.*

Figures 35 and 36 show the changes in R^2 after gradually dropping features in different XAI methods for LightGBM and MLP models, respectively, constructed for “housing price” dataset. From these data, we can intuitively compare the decrease and conclude that the accuracy in explanation of the results given by proposed XAI method ShapG is better than given by other existing methods.

Figures 37 and 38 show the changes in accuracy after gradually dropping features based on different XAI methods for LightGBM and MLP models, respectively, constructed for “H1N1” dataset. From Figure 37 we can observe that in explanation results for the LightGBM model, the accuracy of three methods: SHAP, SamplingSHAP, and ShapG, exhibit a similar trend of steady decline. Although the accuracy of ShapG remains unchanged when removing the five and six features and other situations does not show an increase, it is noteworthy that SHAP shows an increase in accuracy when removing the first six and eight features, while SamplingSHAP shows continuous increase in accuracy when removing the first six, seven, and eight features. Moreover, although ShapG’s explanation performance is comparable to most XAI methods, in terms of accuracy changes after the top ten features are re-

Table 28: Feature importance ranking for “H1N1” dataset

LightGBM model

Rank \ XAI	SHAP	SamplingSHAP	LIME	Feature Importance	ShapG
Top 1	15	10	10	30	10
Top 2	10	15	15	35	17
Top 3	16	17	16	34	15
Top 4	17	16	14	22	16
Top 5	20	20	27	16	35
Top 6	19	22	20	17	34
Top 7	11	30	17	31	14
Top 8	30	18	35	23	11
Top 9	35	35	18	20	20
Top 10	21	34	4	26	2

MLP model

Rank \ XAI	SHAP	SamplingSHAP	LIME	Permutation Feature Importance	ShapG
Top 1	17	17	17	17	10
Top 2	10	35	10	35	17
Top 3	20	21	28	10	11
Top 4	35	15	29	15	15
Top 5	34	10	32	11	14
Top 6	15	30	22	31	16
Top 7	16	31	11	18	20
Top 8	30	18	8	30	19
Top 9	27	22	30	29	35
Top 10	1	29	7	24	2

moved, ShapG performs the best. As we can see in Figure 38, ShapG significantly outperforms other XAI methods in the explanation results based on the MLP model.

In order to compare running time of XAI methods based on game-theoretical approach, we compare KernelSHAP, SamplingSHAP, and ShapG. We perform the following results. Tables 29 and 30 show running time of XAI methods for both LightGBM and MLP models on “housing price” and “H1N1” datasets, respectively. The comparison obviously shows that ShapG method is much faster among such game-theoretical methods as KernelSHAP and SamplingSHAP. With “housing price” dataset with 13 features, ShapG works more than 5 (4.8) times faster than KernelSHAP with LightGBM (MLP) model. We do not start KernelShap with “H1N1” dataset containing 35 features because an estimated time of its work is approximately 4219 (39284) min for LightGBM (MLP) model, which is more than 161 (685) times more than with our ShapG method.

We do not consider SHAP in comparison analysis of a running time for several reasons:

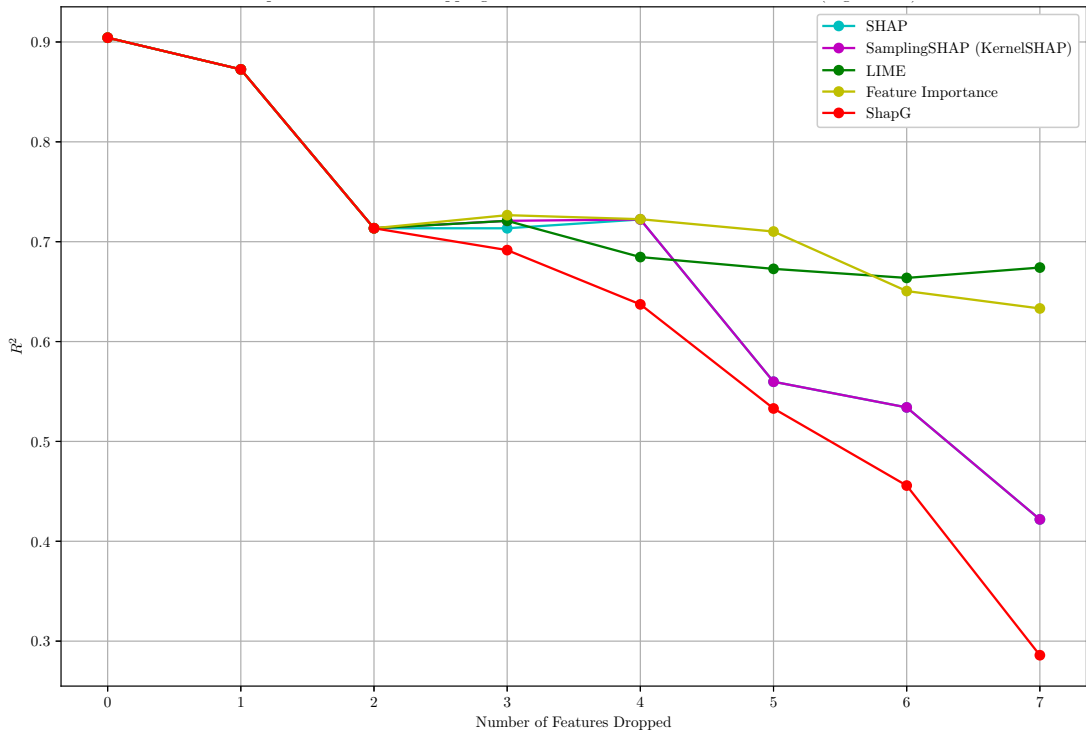


Figure 35: Comparison of R^2 after dropping features based on different XAI methods in “housing price” dataset (LightGBM)

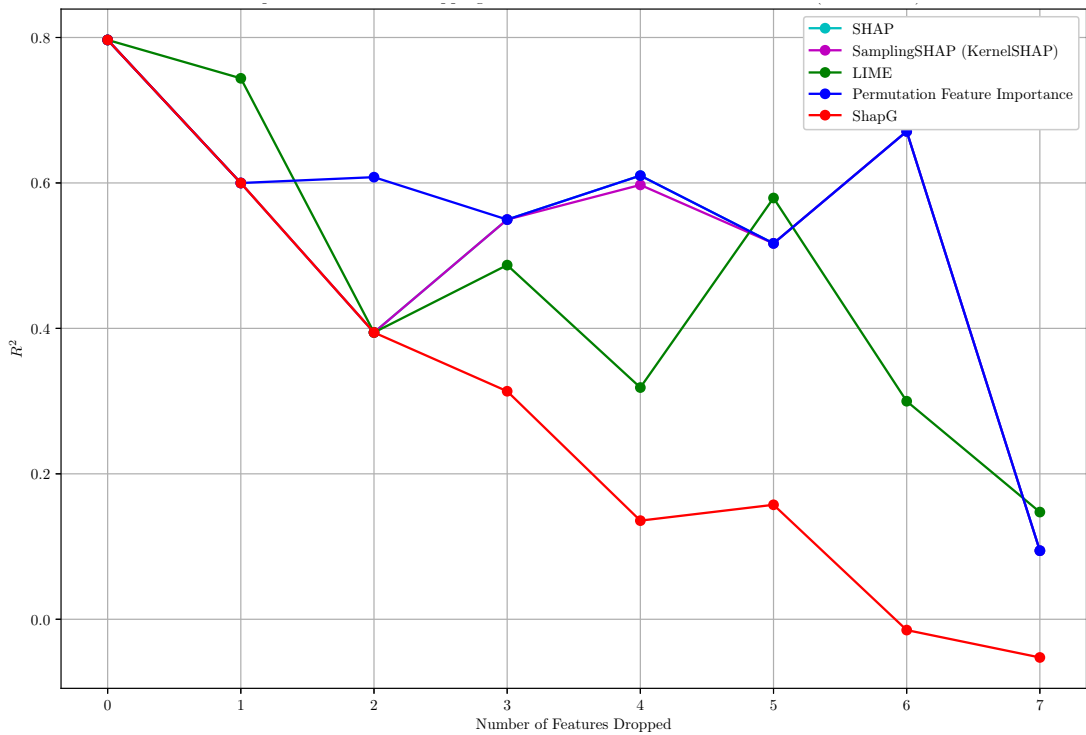


Figure 36: Comparison of R^2 after dropping features based on different XAI methods in “housing price dataset” (MLP model)

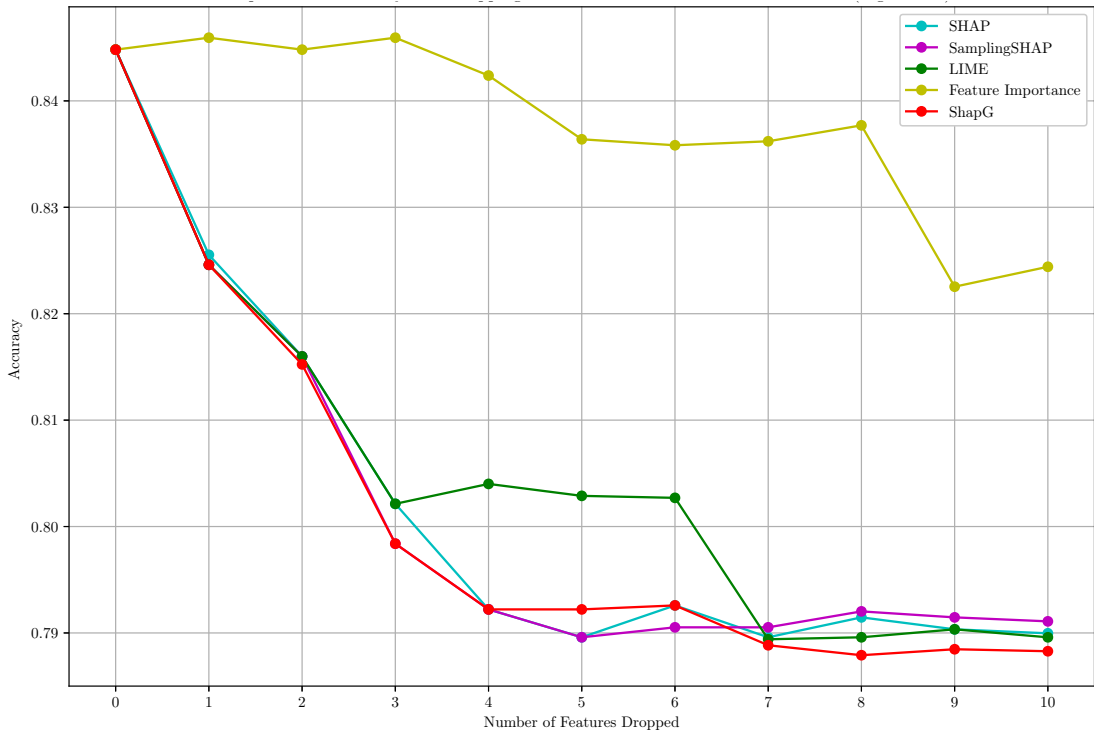


Figure 37: Comparison of accuracy after dropping features based on different XAI methods in “H1N1” dataset (LightGBM)

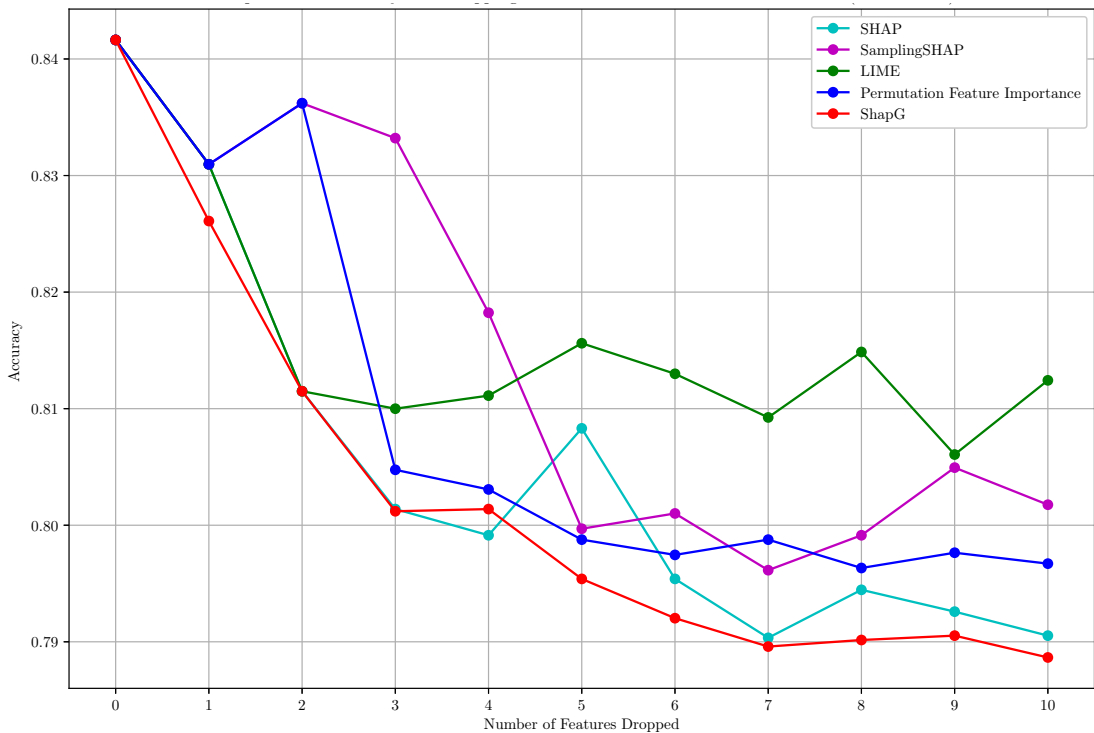


Figure 38: Comparison of accuracy after dropping features based on different XAI methods in “H1N1” dataset (MLP model)

Table 29: Running time (in sec) of XAI methods for “housing price” dataset

LightGBM model

XAI method	KernelSHAP	SamplingSHAP	ShapG
Running time	214.55 s	42.14 s	37.73 s

MLP model

XAI method	KernelSHAP	SamplingSHAP	ShapG
Running time	901.51 s	269.85 s	184.22 s

Table 30: Running time of XAI methods for “H1N1” dataset

LightGBM model

XAI method	KernelSHAP	SamplingSHAP	ShapG
Running time	~ 4219 min	374 min 87 s	26 min 12 s

MLP model

XAI method	KernelSHAP	SamplingSHAP	ShapG
Running time	~ 39284 min	364 min 50 s	57 min 22 s

- When explaining tree models (e.g., LightGBM), SHAP calls the TreeSHAP Explainer. TreeSHAP utilizes the properties of the tree model itself to quickly compute approximate SHAP values, which significantly improves computational efficiency. However, the limitation of TreeSHAP is that it can only be used to explain tree models. This means that we cannot apply it to complex models such as neural networks.
- SHAP can automatically select the most suitable explainer for different AI models. When we use SHAP to explain MLP model, SHAP calls PermutationExplainer to generate explanation results. It works by iterating through forward and reverse feature permutations, which change the features one by one, thus effectively evaluating the independent contribution of each feature to the final output. PermutationExplainer does not directly construct a subset of all possible features, and does not use the Shapley value formula. Instead, it uses a method of permuting features to approximate the SHAP value.

By comparing evaluation results and running time, our XAI method ShapG provides a significant advantage in efficiency and speed. This further confirms the feasibility and usefulness of ShapG algorithm. Moreover, in the following section we show that it can be used for prediction models with complex architectures when

other XAI methods based on game-theoretical approach, like KernelSHAP and SamplingSHAP do not work.²⁰ This makes ShapG method a high-performance tool for XAI.

4.4.3 Explanation of complex models

The proposed XAI method ShapG can be used not only for a single model like LightGBM or MLP as we show in the previous section, but it also provides global explanations for more complex models. These complex models include single models with complex architectures, ensemble learning models, hybrid models, etc. The models we use in this section are described in Section 4.3.3. ShapG method can be applied to a wide range of models and can provide explanations of their decision-making processes.

Tables 31 and 32 represent feature importance ranking given by ShapG for hybrid models constructed by combining different types of AI models in two-by-two combinations for “housing price” and “H1N1” datasets, respectively. We can observe very minor differences in the ranking of feature importance for different models.

Table 31: Feature importance ranking by ShapG for complex AI models for “housing price” dataset

Rank \ AI Model	LGB	Stacking	Linear-LGB	Linear-MLP	Linear-KNN	MLP-LGB	KNN-LGB	MLP-KNN
Top 1	13	13	13	13	13	13	13	13
Top 2	6	5	6	6	6	6	6	6
Top 3	5	6	5	2	11	3	5	3
Top 4	11	3	11	3	3	5	3	11
Top 5	3	11	3	11	5	2	11	2
Top 6	2	2	2	8	2	11	2	5
Top 7	8	10	8	1	10	8	8	1

For the “housing price” dataset, from Table 31 we can clearly observe that feature “LSTAT – lower status of the population (13)” is the most important feature in predicting housing prices, and it is the same as is identified by simple LightGBM and MLP models. In addition, we can also notice that features “RM – average number of rooms per dwelling(6)”, “INDUS – proportion of non-retail business acres

²⁰They could give the result but running time is so large that it makes them impossible to apply in practice, especially, for datasets with many features.

per town (3)”, and “NOX – nitric oxides concentration (5)” appear more frequently in the top important features among all complex models.

For the “H1N1” dataset, from Table 32 we can clearly observe that “doctor recd h1n1 (10)” and “opinion h1n1 risk (17)” are the two most important features, which indicate that they have a significant effect on whether respondent received H1N1 flu vaccine. Similarly, it can be noticed that “health insurance (15)” and “opinion h1n1 vacc effective (16)” appear more frequently among the most important features. This means that these features are also important factors that affect people’s intention to receive vaccination.

Table 32: Feature importance ranking by ShapG for complex AI models for “H1N1” dataset

Rank \ AI Model	LGB	Stacking	Logistic-LGB	Logistic-MLP	Logistic-KNN	MLP-LGB	KNN-LGB	MLP-KNN
Top 1	10	10	10	10	10	10	10	10
Top 2	17	17	17	17	17	17	17	17
Top 3	15	15	16	15	20	15	16	15
Top 4	16	16	11	14	16	11	15	14
Top 5	35	35	15	11	15	14	35	11
Top 6	34	14	14	20	11	35	11	24
Top 7	14	34	20	16	14	16	34	13
Top 8	11	11	19	5	24	20	14	20
Top 9	20	20	13	13	13	34	20	16
Top 10	2	19	1	1	29	24	2	35

Table 33: Running time by ShapG for complex AI models for “housing price” dataset

AI Model	LGB	Stacking	Linear-LGB	Linear-MLP	Linear-KNN	MLP-LGB	KNN-LGB	MLP-KNN
Running time	37.73 s	457.37 s	51.51 s	364.32 s	11.03 s	536.28 s	51.47 s	377.75 s

Table 34: Running time by ShapG for complex AI models for “H1N1” dataset

AI Model	LGB	Stacking	Logistic-LGB	Logistic-MLP	Logistic-KNN	MLP-LGB	KNN-LGB	MLP-KNN
Running Time	26 min 12 s	249 min 35s	42 min 31 s	61 min 17 s	115 min 02 s	105 min 03 s	157 min 16 s	156 min 44 s

Tables 33 and 34 show the running time required by the ShapG algorithm to explain complex AI models for “housing price” and “H1N1” datasets, respectively. Depending on the size of the dataset and model complexity, ShapG requires different running time. As expected, running time for “H1N1” dataset containing 35 features is much larger than the time required for the same model constructed for “housing price” dataset with 13 features. Although running time required by ShapG for complex AI models is much larger than for simple models (LightGBM or MLP), the ShapG method has many significant advantages in comparison with other XAI methods. The ShapG method has not only been applicable to any model from

theoretical point of view, but its performance is also verified in many experiments with complex AI models. Although SHAP, SamplingSHAP, and KernelSHAP are assumed to be applicable to explain any model, during our experiments we found that when applying these methods to more complex neural network models or hybrid models, the code often did not run successfully and did not give valid explanation results. These methods lack good compatibility with complex AI models and require more in-depth adjustments to the dataset or AI models. Therefore, for researchers who are not specialized in the field of AI, these methods cannot easily provide explainable results. When explaining complex AI models, ShapG is still able to provide reliable explanation results.

4.5 Conclusion to Chapter 4

In this chapter, we proposed a new explainable artificial intelligence method called ShapG, which is based on the Shapley value for graph games. The main idea of this algorithm is from the proposed centrality measures in Chapter 3. ShapG is a model-agnostic global explanation method, which calculates feature importance by constructing an undirected graph of features, where nodes in the graph represent features, and samples based on graph. It starts with an empty graph and consequently adds the edges, which are pairs of features with the strongest correlation. The algorithm stops when all features are connected to ensure that the feature graph contains important structural information. In the process of calculating the Shapley value, we only need to consider the coalitions between each node and its reachable nodes, not all possible coalitions. This optimization improves the efficiency of the algorithm.

We have compared ShapG with several popular XAI methods, e.g., Feature Importance (FI), Permutation Feature Importance (PFI), LIME, SHAP, SamplingSHAP, and KernelSHAP. Our ShapG exhibits excellent explanation results, which are significantly better than other XAI methods for two datasets. In addition, compared to SamplingSHAP and KernelSHAP methods also based on cooperative game theory, ShapG saves significant computational resources in running time. These results provide validation of reliability and wide applicability of our method.

ShapG can be considered as a useful XAI method that can be applied not only to simple AI models, but also to provide global explanations for complex models. It can reliably explain decision-making process of complex models, thus helping users to better understand these models.

Conclusions

This thesis is devoted to the analyse of the influence of the network structure and the behavior of agents on the opinion dynamics in social networks. Through this research, opinion dynamics of several specific network structures are described by statistical-physical (macroscopic) models or simulated by microscopic models under the assumption that the agents can interact in the internal layer. A series of classical network structures is examined in terms of the opinion dynamics within macroscopic-GCVM and microscopic-GCVM with different configurations (Chapter 1 and 2). Chapter 2 examines how network properties and individual's behavior affect the opinion dynamics. A two-layer generated network based on a real social network, the Zachery's karate club network, is used to simulate opinion dynamics by microscopic-GCVM, and the impact of centralities of authority nodes on opinion dynamics is discussed in Chapter 3. The approach of simplifying the two-layer network with opinion dynamics in this network to one-layer weighted network and the algorithms of approximating the game-theoretic centrality measures are discovered in Chapter 3. In Chapter 4, a global explanation method for machine learning models based on game-theoretic centrality measures proposed in Chapter 3 is developed.

The main results of the work are the follows:

1. We formulate a series of opinion dynamics models for different network structures (i.e. complete, cycle, star, two-star-even, two-star-odd, two-clique-even, and two-clique-odd for internal layer, and complete, and cycle for external layer), and simulated the models by the statistical-physics method with different parameters which are related to the individual's behavior. The main observations are: (i) If individuals in the social network are not good at expressing their opinions publicly, internal interaction does not have a great influence on the winning rate and consensus time; (ii) a simple external layer network structure such as cyclic external structure significantly prolongs the consensus time, even creates troubles for reaching a consensus. (See [24].)
2. We use the proposed microscopic-GCVM to simulate the opinion dynamics and observe the impact of each type of behavior and the combination of behaviors on opinion dynamics, respectively. The simulation results further verify the correctness of our macroscopic model proposed in Chapter 1. Three different ways to extend a cyclic external structure to a complete one are examined, and

we verify a hypothesis about an impact of the way of extension of a circle to a complete graph impacts consensus time and winning rate. The main results can be summarized as follows: (i) there is a strong linear relationship between pairwise average shortest path and consensus time; (ii) the way of extension of a circle to a complete graph has a significant impact on consensus time and winning rate; (iii) cyclic external structure always increases consensus time; (iv) cyclic external structure has a positive impact on a winning rate; (v) each parameter has a different impact on consensus time but almost no impact on winning rate, for instance, expressing your real opinion to a certain extent is not effective to reach consensus within the whole system. But beyond this threshold, along with an increase of desire to express your opinion, for the system it is easier to reach consensus, and when people are more willing to accept their own external opinion, it will accelerate consensus of the whole system; (vi) the combination of parameters has a significant impact on consensus time. (See [25].)

3. A simplification approach for a two-layer network with the predefined opinion dynamics to one-layer weighted network is proposed, this approach is suitable for stochastic processes in two-layer networks with similar dynamics. The algorithms to approximate the game-theoretic centrality measures are discovered. The testing results show that the simplification method and the proposed algorithms are efficient. We also found a strong and significant negative correlation between centralities of authoritative nodes and consensus time, i.e. the main person in a social network plays a crucial role for reaching consensus, which is expected. (See [51].)
4. We propose a new explainable artificial intelligence algorithm assuming features in the machine learning models as nodes on an undirected graph constructed by a special way. A series of experiments verified the interpretability, applicability, and runtime speed of the algorithm. Our algorithm is superior to other algorithms (including PFI, FI, LIME and SHAP) in these three aspects. (See [52])

We conclude that all the tasks formulated in this thesis are achieved, and the objectives are fully accomplished.

List of acronyms and symbols

BVM basic voter model.

CVM concealed voter model.

FI feature importance.

GCVM general concealed voter model.

KNN K-nearest neighbors.

KPI key performance indicator.

LIME local interpretable model-agnostic explanations.

MLP multilayer perceptron.

PCC Pearson correlation coefficient.

PFI permutation feature importance.

SHAP Shapley additive explanations.

ShapG explanations based on Shapley value for graphs.

XAI explainable artificial intelligence.

List of Figures

1	A complete network connecting 10 individuals in BVM	24
2	A two-layer network structure in CVM with 10 individuals	25
3	Representation of GCVM with 10 individuals: symmetric case	29
4	Representation of an internal star-coupled network	32
5	Representation of internal two star-coupled network	34
6	Representation of internal two-clique network	37
7	Representation of external cycle network	40
8	Representation of external cycle and internal two star-coupled network	40
9	Representation of external cycle and internal two-clique network . . .	41
10	Consensus time for 20 runs, comparison between BVM, CVM and GCVM with external complete network structure	43
11	Consensus time for 20 runs, comparison between all models	43
12	Observed average consensus time for all models	44
13	Winning rate of red opinion for all models	45
14	Observed consensus time (a) and winning rate (b) for models with different two-layer network structures	53
15	Winning rate (a) and consensus time (b) for the models with different extension ways	56
16	Consensus time vs d	57
17	Pearson correlation coefficients	58
18	Frequency mass function and empirical cumulative distribution func- tion (ECDF) for different KPIs	59
19	Winning rate for different copying rates	64
20	Winning rate for different externalization and internalization rates . .	65
21	Consensus time for different copying rates	66
22	Consensus time for different externalization and internalization rates	67
23	Empirical cumulative distribution function of consensus time with respect to parameters π_{c_e} , π_{c_i} , π_e , and π_i	70
24	One-layer Zachary's karate club network	75

25	Two-layer networks used in CVM and GCVM: (a) CVM: two-layer network with external Zachary’s karate club and empty internal layer, (b) GCVM: two-layer network with external Zachary’s karate club and star internal layer.	76
26	Internal average shortest path and density for different network structures	97
27	Different centralities for different structures	98
28	Winning rate and consensus time for different structures	98
29	Heatmap of Pearson correlation coefficients for the “housing price” dataset	112
30	Heatmap of Pearson correlation coefficients for the “H1N1” dataset .	113
31	Graph connecting features in “housing price” dataset	113
32	Graph connecting features in “H1N1” dataset	114
33	Feature importance in “housing price” dataset calculated with ShapG	118
34	Feature importance in “H1N1” dataset calculated with ShapG	119
35	Comparison of R^2 after dropping features based on different XAI methods in “housing price” dataset (LightGBM)	124
36	Comparison of R^2 after dropping features based on different XAI methods in “housing price dataset” (MLP model)	124
37	Comparison of accuracy after dropping features based on different XAI methods in “H1N1” dataset (LightGBM)	125
38	Comparison of accuracy after dropping features based on different XAI methods in “H1N1” dataset (MLP model)	125

List of Tables

1	Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in CVM	27
2	Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM system with symmetric case	30
3	Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with internal star-coupled network	33
4	Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with two star-coupled internal networks (odd case)	35
5	Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with two star-coupled internal networks (even case)	36
6	Transitions from the state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with internal two clique-coupled networks (odd case)	38
7	Transitions from state $(\rho_{r_e}, \rho_{r_i}, \rho_r)$ in GCVM with internal two clique-coupled networks (even case)	39
8	Results of Kolmogorov–Smirnov tests	60
9	Results of normality tests	61
10	Results of variance equity tests	62
11	Results of mean equity tests	63
12	Observed minimum and maximum consensus time	68
13	Observed minimum and maximum winning rate	68
14	Descriptive statistics of clusters within winning rate (complete-complete-50)	69
15	Consensus time cluster sizes for each model	71
16	Results on the Shapley value when ξ scaling factor is applied	89
17	Results on the Myerson value without ξ scaling	91
18	Results on the Myerson value when ξ scaling is applied	91
19	Results on the Myerson value without ξ scaling for networks with 10 nodes	92
20	Results on the Myerson value when ξ scaling is applied for networks with 10 nodes	93
21	Exact Shapley value vs an approximated Shapley value for “graph-1.0”	94
22	Exact Myerson value vs an approximated Myerson value for “graph-0.2”	94

23	Accuracy defined by (32) for the proposed in this paper and classical centrality measures	95
24	Correlation coefficients	100
25	Feature No. in “house price” dataset	120
26	Feature No. in “H1N1” dataset	120
27	Feature importance ranking for “housing price” dataset	121
28	Feature importance ranking for “H1N1” dataset	123
29	Running time (in sec) of XAI methods for “housing price” dataset . .	126
30	Running time of XAI methods for “H1N1” dataset	126
31	Feature importance ranking by ShapG for complex AI models for “housing price” dataset	127
32	Feature importance ranking by ShapG for complex AI models for “H1N1” dataset	128
33	Running time by ShapG for complex AI models for “housing price” dataset	128
34	Running time by ShapG for complex AI models for “H1N1” dataset .	128

References

- [1] Wiener N. Cybernetics // Bulletin of the American Academy of Arts and Sciences. — 1950. — Vol. 3, no. 7. — P. 2–4.
- [2] Wiener N. The human use of human beings: cybernetics and society. — Houghton Mifflin, 1950.
- [3] McKeehan L. W. A contribution to the theory of ferromagnetism // Physical Review. — 1925. — Vol. 26, no. 2. — P. 274.
- [4] Holley R. A., Liggett T. M. Ergodic theorems for weakly interacting infinite systems and the voter model // The annals of probability. — 1975. — P. 643–663.
- [5] Noorazar H. Recent advances in opinion propagation dynamics: A 2020 survey // The European Physical Journal Plus. — 2020. — Vol. 135. — P. 1–20.
- [6] Sznajd-Weron K., Sznajd J. Opinion evolution in closed community // International Journal of Modern Physics C. — 2000. — Vol. 11, no. 06. — P. 1157–1165.
- [7] DeGroot M. H. Reaching a consensus // Journal of the American Statistical Association. — 1974. — Vol. 69, no. 345. — P. 118–121.
- [8] Friedkin N. E., Johnsen E. C. Social influence and opinions // Journal of Mathematical Sociology. — 1990. — Vol. 15, no. 3-4. — P. 193–206.
- [9] Hegselmann R., Ulrich K. Opinion dynamics and bounded confidence: models, analysis and simulation // Journal of Artificial Societies and Social Simulation (JASSS). — 2002. — Vol. 5, no. 3.
- [10] Mixing beliefs among interacting agents / Deffuant G., Neau D., Amblard F., and Weisbuch G. // Advances in Complex Systems. — 2000. — Vol. 3, no. 01n04. — P. 87–98.
- [11] Novel multidimensional models of opinion dynamics in social networks / Parsegov S. E., Proskurnikov A. V., Tempo R., and Friedkin N. E. // IEEE Transactions on Automatic Control. — 2016. — Vol. 62, no. 5. — P. 2270–2285.
- [12] Rogov M. A., Sedakov A. A. Coordinated Influence on the Opinions of Social Network Members // Autom Remote Control. — 2020. — Vol. 81. — P. 528–547.

- [13] Mazalov V. V., Parilina E. M. The Euler-Equation Approach in Average-Oriented Opinion Dynamics // Mathematics. — 2020. — Vol. 8, no. 3. — Access mode: <https://www.mdpi.com/2227-7390/8/3/355>.
- [14] Kareeva Y. S., Sedakov A. A., Zhen M. Influence in social networks with stubborn agents: From competition to bargaining // Applied Mathematics and Computation. — 2023. — Vol. 444. — P. 127790. — Access mode: <https://www.sciencedirect.com/science/article/pii/S009630032200858X>.
- [15] Opinion dynamics in finance and business: a literature review and research opportunities / Zha Q., Kou G., Zhang H., Liang H., Chen X., Li C.-C., and Dong Y. // Financial Innovation. — 2020. — Vol. 6. — P. 1–22.
- [16] Krasnoshchekov P. S. The simplest mathematical model of behaviour. Psychology of conformism // Matematicheskoe Modelirovanie [In Russian]. — 1998. — Vol. 10, no. 7. — P. 76–92.
- [17] Kozitsin I. V., Belolipetskii A. A. Opinion convergence in the Krasnoshchekov model // The Journal of Mathematical Sociology. — 2019. — Vol. 43, no. 2. — P. 104–121.
- [18] Bounded confidence opinion dynamics: A survey / Bernardo C., Altafini C., Proskurnikov A. V., and Vasca F. // Automatica. — 2024. — Vol. 159. — P. 111302. — Access mode: <https://www.sciencedirect.com/science/article/pii/S0005109823004661>.
- [19] Éminence Grise Coalitions: On the Shaping of Public Opinion / Bolouki S., Malhamé R. P., Siami M., and Motee N. // IEEE Transactions on Control of Network Systems. — 2017. — Vol. 4, no. 2. — P. 133–145.
- [20] A survey on the fusion process in opinion dynamics / Dong Y., Zhan M., Kou G., Ding Z., and Liang H. // Information Fusion. — 2018. — Vol. 43. — P. 57–65.
- [21] Gastner M. T., Oborny B., Gulyás M. Consensus time in a voter model with concealed and publicly expressed opinions // Journal of Statistical Mechanics: Theory and Experiment. — 2018. — Vol. 2018, no. 6. — P. 063401.
- [22] The impact of hypocrisy on opinion formation: A dynamic model / Gastner M. T., Takács K., Gulyás M., Szvetelszky Z., and Oborny B. // PloS one. — 2019. — Vol. 14, no. 6. — P. e0218729.

- [23] Donati S., Zappalà S., González-Romá V. The influence of friendship and communication network density on individual innovative behaviours: a multilevel study // *European Journal of Work and Organizational Psychology*. — 2016. — Vol. 25, no. 4. — P. 583–596.
- [24] Zhao C., Parilina E. M. Opinion Dynamics in Two-Layer Networks with Hypocrisy // *Journal of the Operations Research Society of China*. — 2024. — Mar. — Vol. 12, no. 1. — P. 109–132. — Access mode: <https://doi.org/10.1007/s40305-023-00503-2>.
- [25] Zhao C., Parilina E. M. Analysis of consensus time and winning rate in two-layer networks with hypocrisy of different structures // *Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes*. — 2024. — Vol. 20, no. 2. — P. 170–192.
- [26] Zhao C., Parilina E. M. Network Structure Properties and Opinion Dynamics in Two-Layer Networks with Hypocrisy // *Mathematical Optimization Theory and Operations Research*. — Cham : Springer Nature Switzerland. — 2024. — P. 300–314.
- [27] Zhao C., Parilina E. M. Consensus time and winning rate based on simulations in two-layer networks with hypocrisy // *2023 7th Scientific School Dynamics of Complex Networks and their Applications (DCNA)*. — 2023. — P. 68–71.
- [28] Mazalov V. V., Chirkova J. V. *Networking games: network forming games and games on networks*. — Academic Press, 2019. — P. 117–118.
- [29] Freeman L. C. A set of measures of centrality based on betweenness // *Sociometry*. — 1977. — P. 35–41.
- [30] Bavelas A. Communication patterns in task-oriented groups // *The journal of the acoustical society of America*. — 1950. — Vol. 22, no. 6. — P. 725–730.
- [31] Freeman L. C. Centrality in social networks conceptual clarification // *Social Networks*. — 1978. — Vol. 1, no. 3. — P. 215–239. — Access mode: <https://www.sciencedirect.com/science/article/pii/0378873378900217>.
- [32] Sabidussi G. The centrality index of a graph // *Psychometrika*. — 1966. — Vol. 31, no. 4. — P. 581–603.

- [33] Powell J., Hopkins M. 9 - Library networks—coauthorship, citation, and usage graphs // A Librarian's Guide to Graphs, Data and the Semantic Web. — Chandos Publishing, 2015. — Chandos Information Professional Series. — P. 75–81. — Access mode: <https://www.sciencedirect.com/science/article/pii/B9781843347538000099>.
- [34] Random walk centrality in interconnected multilayer networks / Solé-Ribalta A., De Domenico M., Gómez S., and Arenas A. // Physica D: Nonlinear Phenomena. — 2016. — Vol. 323. — P. 73–79.
- [35] Newman M. A measure of betweenness centrality based on random walks // Social networks. — 2005. — Vol. 27, no. 1. — P. 39–54.
- [36] Game-theoretic network centrality: A review / Tarkowski M. K., Michalak T. P., Rahwan T., and Wooldridge M. // arXiv preprint arXiv:1801.00218. — 2017.
- [37] Shapley L. S. A value for n-person games // Contribution to the Theory of Games. — 1953. — Vol. 2.
- [38] Myerson R. B. Graphs and cooperation in games // Mathematics of operations research. — 1977. — Vol. 2, no. 3. — P. 225–229.
- [39] Suri N. R., Narahari Y. Determining the top-k nodes in social networks using the shapley value // Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 3. — 2008. — P. 1509–1512.
- [40] Game-theoretic centrality measures for weighted graphs / Mazalov V. V., Avrachenkov K. E., Trukhina L. I., and Tsynguev B. T. // Fundamenta Informaticae. — 2016. — Vol. 145, no. 3. — P. 341–358.
- [41] Mazalov V. V., Khitraya V. A. A modified Myerson value for determining the centrality of graph vertices // Automation and Remote Control. — 2021. — Vol. 82. — P. 145–159.
- [42] Khitraya V. A., Mazalov V. V. Game-Theoretic Centrality of Directed Graph Vertices // Automation And Remote Control. — 2024. — Vol. 85, no. 2.
- [43] Boldi P., Vigna S. Axioms for centrality // Internet Mathematics. — 2014. — Vol. 10, no. 3-4. — P. 222–262.

- [44] Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI / Arrieta A. B., Díaz-Rodríguez N., Del Ser J., Bennetot A., Tabik S., Barbado A., García S., Gil-López S., Molina D., Benjamins R., et al. // *Information fusion*. — 2020. — Vol. 58. — P. 82–115.
- [45] Plumb G., Molitor D., Talwalkar A. S. Model agnostic supervised local explanations // *Advances in neural information processing systems*. — 2018. — Vol. 31.
- [46] Global explanations of neural networks: Mapping the landscape of predictions / Ibrahim M., Louie M., Modarres C., and Paisley J. // *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. — 2019. — P. 279–287.
- [47] Ribeiro M. T., Singh S., Guestrin C. " Why should i trust you?" Explaining the predictions of any classifier // *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. — 2016. — P. 1135–1144.
- [48] Lundberg S. M., Lee S.-I. A unified approach to interpreting model predictions // *Advances in neural information processing systems*. — 2017. — Vol. 30.
- [49] From local explanations to global understanding with explainable AI for trees / Lundberg S. M., Erion G., Chen H., DeGrave A., Prutkin J. M., Nair B., Katz R., Himmelfarb J., Bansal N., and Lee S.-I. // *Nature machine intelligence*. — 2020. — Vol. 2, no. 1. — P. 56–67.
- [50] Zhao C. shapG. — PyPI: <https://pypi.org/project/shapG/>; GitHub: <https://github.com/vectorsss/shapG>. — 2024. — July. — (Accessed on 11/16/2024).
- [51] Zhao C., Parilina E. M. Centrality measures and opinion dynamics in two-layer networks with replica nodes // *arXiv preprint arXiv:2406.18780v2*. — 2024. — 2406.18780v2.
- [52] Zhao C., Liu J., Parilina E. M. ShapG: new feature importance method based on the Shapley value // *arXiv preprint arXiv:2407.00506*. — 2024. — 2407.00506.
- [53] Certificate of state registration of the computer program No. 2023661532 Russian Federation : Program for modeling the dynamics of binary opinion spread

- in two-layer networks Zhao C. (CN) ; Federal State Budgetary Educational Institution of Higher Education “Saint Petersburg State University” (SPbSU) ; Zhao Chi. — no. 2023660681 ; req. 24.05.2023 ; publ. 01.06.2023.
- [54] Festinger L. A theory of cognitive dissonance. — Stanford : Stanford University Press, 1957.
- [55] Synchronization and graph topology / Belykh I., Hasler M., Lauret M., and Nijmeijer H. // International Journal of Bifurcation and Chaos. — 2005. — Vol. 15, no. 11. — P. 3423–3433.
- [56] Bianconi G. Multilayer networks: structure and function. — Oxford university press, 2018.
- [57] SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python / Virtanen P., Gommers R., Oliphant T. E., Haberland M., Reddy T., Cournapeau D., Burovski E., Peterson P., Weckesser W., Bright J., van der Walt S. J., Brett M., Wilson J., Millman K. J., Mayorov N., Nelson A. R. J., Jones E., Kern R., Larson E., Carey C. J., Polat İ., Feng Y., Moore E. W., VanderPlas J., Laxalde D., Perktold J., Cimrman R., Henriksen I., Quintero E. A., Harris C. R., Archibald A. M., Ribeiro A. H., Pedregosa F., van Mulbregt P., and SciPy 1.0 Contributors // Nature Methods. — 2020. — Vol. 17. — P. 261–272.
- [58] `scipy.stats.pearsonr` — SciPy v1.12.0 Manual // <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html>. — (Accessed on 02/11/2024).
- [59] Student. Probable Error Of A Correlation Coefficient // Biometrika. — 1908. — Sep. — Vol. 6, no. 2-3. — P. 302–310. — https://academic.oup.com/biomet/article-pdf/6/2-3/302/49729619/biomet_6_2-3_302.pdf.
- [60] Simard R., L’Ecuyer P. Computing the two-sided Kolmogorov-Smirnov distribution // Journal of Statistical Software. — 2011. — Vol. 39. — P. 1–18.
- [61] `scipy.stats.kstest` — SciPy v1.12.0 Manual // <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kstest.html>. — (Accessed on 02/11/2024).

- [62] Shapiro S. S., Wilk M. B. An analysis of variance test for normality (complete samples) // *Biometrika*. — 1965. — Vol. 52, no. 3/4. — P. 591–611.
- [63] `scipy.stats.normaltest` — SciPy v1.12.0 Manual // <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.normaltest.html>. — (Accessed on 02/11/2024).
- [64] Levene H. *Contributions to probability and statistics: essays in honor of Harold Hotelling*. — Stanford University Press, 1960. — P. 278–292.
- [65] Bartlett M. S. Properties of sufficiency and statistical tests // *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*. — 1937. — Vol. 160, no. 901. — P. 268–282.
- [66] Fligner M. A., Killeen T. J. Distribution-free two-sample tests for scale // *Journal of the American Statistical Association*. — 1976. — Vol. 71, no. 353. — P. 210–213.
- [67] `scipy.stats.levene` — SciPy v1.12.0 Manual // <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.levene.html>. — (Accessed on 02/11/2024).
- [68] Kruskal W. H., Wallis W. A. Use of ranks in one-criterion variance analysis // *Journal of the American statistical Association*. — 1952. — Vol. 47, no. 260. — P. 583–621.
- [69] Heiman G. W. *Understanding research methods and statistics: An integrated introduction for psychology*. — Houghton, Mifflin and Company, 2001.
- [70] Alexander R. A., Govern D. M. A new and simpler approximation for ANOVA under variance heterogeneity // *Journal of Educational Statistics*. — 1994. — Vol. 19, no. 2. — P. 91–101.
- [71] `scipy.stats.kruskal` — SciPy v1.12.0 Manual // <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kruskal.html>. — (Accessed on 02/11/2024).
- [72] Fast Exact k-Means, k-Medians and Bregman Divergence Clustering in 1D. — 2018. — Apr. — arXiv:1701.07204 [cs]. online; accessed: <http://arxiv.org/abs/1701.07204> (online; accessed: 2023-10-24).

- [73] Scikit-learn: Machine Learning in Python / Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., and Duchesnay E. // *Journal of Machine Learning Research*. — 2011. — Vol. 12. — P. 2825–2830.
- [74] Rousseeuw P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis // *Journal of computational and applied mathematics*. — 1987. — Vol. 20. — P. 53–65.
- [75] Zachary W. W. An information flow model for conflict and fission in small groups // *Journal of anthropological research*. — 1977. — Vol. 33, no. 4. — P. 452–473.
- [76] Dense graph - Wikipedia // https://en.wikipedia.org/wiki/Dense_graph. — (Accessed on 02/20/2024).
- [77] Everett M. G., Borgatti S. P. The centrality of groups and classes // *The Journal of mathematical sociology*. — 1999. — Vol. 23, no. 3. — P. 181–201.
- [78] Everett M. G., Borgatti S. P. Extending centrality // *Models and methods in social network analysis*. — 2005. — Vol. 35, no. 1. — P. 57–76.
- [79] Measuring and maximizing group closeness centrality over disk-resident graphs / Zhao J., Lui J. C., Towsley D., and Guan X. // *Proceedings of the 23rd International Conference on World Wide Web*. — 2014. — P. 689–694.
- [80] Chung F. R. *Spectral graph theory*. — American Mathematical Soc., 1997. — Vol. 92.
- [81] Newman M. *Networks: An Introduction*. — Oxford University Press, 2010. — Mar. — ISBN: 9780199206650. — Access mode: <https://doi.org/10.1093/acprof:oso/9780199206650.001.0001>.
- [82] Freeman L. C., Borgatti S. P., White D. R. Centrality in valued graphs: A measure of betweenness based on network flow // *Social networks*. — 1991. — Vol. 13, no. 2. — P. 141–154.
- [83] Stephenson K., Zelen M. Rethinking centrality: Methods and examples // *Social networks*. — 1989. — Vol. 11, no. 1. — P. 1–37.

- [84] Lovász L. Random walks on graphs: A survey // *Combinatorics, Paul Erdos is Eighty.* — 1993. — Vol. 2, no. 1. — P. 1–46. — Access mode: http://scholar.google.de/scholar.bib?q=info:11cRghStI1oJ:scholar.google.com/&output=citation&hl=de&as_sdt=0,5&ct=citation&cd=3.
- [85] Mean first-passage time for random walks on undirected networks / Zhang Z., Julaiti A., Hou B., Zhang H., and Chen G. // *The European Physical Journal B.* — 2011. — Vol. 84. — P. 691–697.
- [86] Kemeny J. G., Snell J. L. et al. *Finite markov chains.* — van Nostrand Princeton, NJ, 1969. — Vol. 26.
- [87] Pólya G. Eine Wahrscheinlichkeitsaufgabe in der Kundenwerbung // *Zammzeitschrift Fur Angewandte Mathematik Und Mechanik.* — 1930. — Vol. 10. — P. 96–97. — Access mode: <https://api.semanticscholar.org/CorpusID:123056679>.
- [88] Jackson M. O. et al. *Social and economic networks.* — Princeton university press Princeton, 2008. — Vol. 3.
- [89] Mazalov V. V., Trukhina L. I. Generating functions and the Myerson vector in communication networks // *Discrete Mathematics and Applications.* — 2014. — Vol. 24, no. 5. — P. 295–303.
- [90] Milgram S. The small world problem // *Psychology today.* — 1967. — Vol. 2, no. 1. — P. 60–67.
- [91] Kendall M. G. The treatment of ties in ranking problems // *Biometrika.* — 1945. — Vol. 33, no. 3. — P. 239–251.
- [92] Spearman C. The proof and measurement of association between two things // *The American journal of psychology.* — 1987. — Vol. 100, no. 3/4. — P. 441–471.
- [93] Louppe G. Understanding random forests: From theory to practice // *arXiv preprint arXiv:1407.7502.* — 2014.
- [94] Permutation importance: a corrected feature importance measure / Altmann A., Toloşi L., Sander O., and Lengauer T. // *Bioinformatics.* — 2010. — Vol. 26, no. 10. — P. 1340–1347.

- [95] Mishra S., Sturm B. L., Dixon S. Local interpretable model-agnostic explanations for music content analysis. // *Ismir*. — 2017. — Vol. 53. — P. 537–543.
- [96] Štrumbelj E., Kononenko I. An efficient explanation of individual classifications using game theory // *The Journal of Machine Learning Research*. — 2010. — Vol. 11. — P. 1–18.
- [97] Castro J., Gómez D., Tejada J. Polynomial calculation of the Shapley value based on sampling // *Computers & Operations Research*. — 2009. — Vol. 36, no. 5. — P. 1726–1730.
- [98] Shapley L. S. Notes on the n-person game—ii: The value of an n-person game. — Rand Corporation, 1951.
- [99] Naumova N. I. Shapley Value and its Extensions. — VVM Publishing House, 2017. — P. 60.
- [100] Štrumbelj E., Kononenko I. Explaining prediction models and individual predictions with feature contributions // *Knowledge and information systems*. — 2014. — Vol. 41. — P. 647–665.
- [101] Lightgbm: A highly efficient gradient boosting decision tree / Ke G., Meng Q., Finley T., Wang T., Chen W., Ma W., Ye Q., and Liu T.-Y. // *Advances in neural information processing systems*. — 2017. — Vol. 30.
- [102] Samatin Njikam A. N., Zhao H. A novel activation function for multilayer feed-forward neural networks // *Applied Intelligence*. — 2016. — Vol. 45. — P. 75–82.
- [103] Polikar R. Ensemble learning // *Ensemble machine learning: Methods and applications*. — 2012. — P. 1–34.
- [104] Stacking ensemble learning for short-term electricity consumption forecasting / Divina F., Gilson A., Gómez-Vela F., García Torres M., and Torres J. F. // *Energies*. — 2018. — Vol. 11, no. 4. — P. 949.
- [105] Weisberg S. Applied linear regression. — John Wiley & Sons, 2005. — Vol. 528.
- [106] LaValley M. P. Logistic regression // *Circulation*. — 2008. — Vol. 117, no. 18. — P. 2395–2399.

- [107] Taud H., Mas J.-F. Multilayer perceptron (MLP) // Geomatic approaches for modeling land change scenarios. — 2018. — P. 451–455.
- [108] KNN model-based approach in classification / Guo G., Wang H., Bell D., Bi Y., and Greer K. // On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings / Springer. — 2003. — P. 986–996.
- [109] Towards a rigorous evaluation of XAI methods on time series / Schlegel U., Arnout H., El-Assady M., Oelke D., and Keim D. A. // 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW) / Ieee. — 2019. — P. 4197–4201.

Appendix

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО
о государственной регистрации программы для ЭВМ
№ 2023661532

**«Программа для моделирования динамики
распространения бинарных мнений в двухслойных
сетях» (TL-Net OD)**

Правообладатель: *федеральное государственное бюджетное
образовательное учреждение высшего образования
"Санкт-Петербургский государственный университет"
(СПбГУ) (RU)*

Автор(ы): *Чжао Чи (CN)*



Заявка № **2023660681**
Дата поступления **24 мая 2023 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ **01 июня 2023 г.**

*Руководитель Федеральной службы
по интеллектуальной собственности*

Ю.С. Зубов