



HOKKAIDO UNIVERSITY

Title	A Study on Finding Core Communities in Social Networks
Author(s)	幸若, 完壮
Degree Grantor	北海道大学
Degree Name	博士(情報科学)
Dissertation Number	甲第11971号
Issue Date	2015-09-25
DOI	https://doi.org/10.14943/doctoral.k11971
Doc URL	https://hdl.handle.net/2115/59956
Type	doctoral thesis
File Information	Sadamori_Koujaku.pdf



A Study on Finding Core Communities in Social Networks



Sadamori Koujaku

Department of Computer Science
Hokkaido University

This dissertation is submitted for the degree of
Doctor of Computer Science

August 2015

I would like to dedicate this thesis to my loving parents ...

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Sadamori Koujaku

August 2015

Acknowledgements

My school experience at Hokkaido University has been exciting, thanks for, in large part, the people that I have met through sailing and research activities. First and foremost, I wish to express my sincere gratitude to Professor Mineichi Kudo, Laboratory for Pattern Recognition and Machine Learning, for letting me work in his laboratory, and for providing me a lot of brick walls and numerous support to break. In my three years here I have learned more than I had expected. When I first joined the lab, I didn't know much about planning, surveying, doing experiments, writing and presentations. Thank you for all of the opportunities you have given me and for supporting me. I am truly grateful.

I am grateful to my supervisors, Professor Hideyuki Imai, Professor Masanori Sugimoto, Associate Professor Atsuyoshi Nakamura and especially Associate Professor Ichigaku Takigawa from whom I learned a lot. Thank you for always answering my questions, and giving me a lot of advices. They taught me what a researcher has to be careful and to prepare through their hard work.

I also would like to express my deep appreciation to my previous supervisors, Professor Hajime Igarashi, Associate Professor So Noguchi and Associate Professor Kota Watanabe for opening my eyes to the world of computer science. They encouraged me to do what I would like to do by myself and, thanks for their support, I could find the fan of science. I owe them for where I am today.

To everyone else in the lab that I did not work with directly. Thank you for fixing computer programs, cooking lunch, and going for drinking with me when I was depressed. And of course I cannot forget about people enjoying sailing with me who taught me how to respond unreasonable and irrational requests. In my eight years, I could meet many attractive, thoughtful and respectable people having a strong will in their mind. Thank you for enjoying sailing with me and also teaching me a lot of precious things. Last but not the least, I would like to thank my family for all their love and support. Thank my mother, father and sister for all the sacrifices they have made. And also thank my grandfather for letting me do whatever I would like to do.

Abstract

In our society people are integrated into communities to share ideas, information and to achieve a common goal. Within a community, a small group of individuals holds great popularities and leaderships, which we call a core community. A core community takes general tasks, coordinates the interactions among members, and holds the important source of the social norm, values and subcultures. Therefore, identifying core communities is of great importance to understand our social systems. There has been concert effort to find core communities embedded in social networks. However, one limitation is that they are too coarse to find multiple cores in a network. We, therefore, proposed a novel community model called dense core. It can capture almost perfect subgraphs and also well-separated subgraphs. On the basis of this unique nature, we have shown its effectiveness in identifying multiple core/periphery structures. The spontaneity is supported by the fact that the dense core model is relevant to various community models in different traditions such as pseudo cliques and graph partitioning. We applied the dense core model for visualizing a graph to find multiple core communities. In addition, we applied our model for the change analysis of dynamic social networks and illustrated its effectiveness.

Table of contents

List of figures	xiii
List of tables	xvii
1 Introduction	1
2 Related Work	5
2.1 Construction of Social Networks	5
2.2 Community structures	6
2.3 Core/Periphery Structures	8
3 Dense core model	11
3.1 Preliminaries	11
3.2 Model	12
3.3 Property	13
3.4 Algorithm	18
3.5 Experiment	19
3.5.1 Synthetic graph	19
3.5.2 Analysis of Real-life network	22
3.6 Discussion	30
3.7 Summary	31
4 Dense core model with sub-peripheries	33
4.1 Model	33
4.2 Property	35
4.3 Algorithm	37
4.3.1 Exact Algorithm	38
4.3.2 Heuristic Algorithm	38
4.4 Experiment	40

4.5	Discussion	43
4.6	Summary	46
5	Application to Graph Visualization	47
5.1	Layout	47
5.2	Algorithm	49
5.3	Related work	51
5.4	Experiment	52
5.5	Summary	58
6	Application to change detection task	59
6.1	Background	59
6.2	Preliminaries	60
6.3	Change detection of Communities	61
6.3.1	Community Models	61
6.3.2	Framework of change analysis	62
6.4	Experiment	64
6.4.1	Analysis of Synthetic graphs	64
6.4.2	Analysis of E-mail interactions	65
6.4.3	Analysis of Phone-call interactions	68
6.5	Discussion	71
6.6	Summary	71
7	Conclusion	73
	References	75
	Appendix A Dinkelbach's Algorithm	81
	Appendix B Spectral clustering	83

List of figures

2.1	Comparison of core/periphery models. (a) The original graph consists of two sub-cliques. The Extaction (b), relaxed core/periphery (d) sets natural intra-community partition between core and periphery, but failed inter-community partition. The k -truss (c) and Modularity (e), on the other hands, sets natural inter-community partition, but, failed intra-community partitions. Among them, only ρ -dense core (f) sets inter/intra-community partitions naturally and uncovers two core/periphery communities.	10
3.1	Communities found with (a) Modularity ($\rho = 1$) and (b) Proposed quality functions ($\rho = 0.5$). A community is the nodes in the same colored region. Modularity divides cliques into different elements, while dense core does not.	13
3.2	Graph of two cores (black circles) and peripheries (white circles).	20
3.3	Communtiiies found with (b) Modularity, (c) Discrete Model, (d) Truss and (e) Dense core. Nodes in the same community is surrounded by the same colored region. The parameters of the models are shown at the bottum of each figure.	21
3.4	Comparison of the proportion of correct partitions.	22
3.5	Graph of Zachary's karate club. Node # 1 and # 34 are the two leading individuals in this club.	23
3.6	Cohesive subgraphs found by Modularity, Truss, Discrete model, and Dense core.	23
3.7	Graph of political books on the United States presidential election in 2004. The color of each node represents the political alignment: liberals (blue), conservatives (red) and centrists (black).	24
3.8	Cohesive subgraphs found by (a) Modularity, Truss, Discrete model, and Dense core. All models except Discrete divided the graph into one conservative community, one liberal community and centrist communities. Truss and Dense core also found very cohesive subgraphs in these communities.	25

3.9	Detailed structures of conservative, and liberal communities. Dense core found more finer cores compared to Truss.	26
3.10	Graph of collaboration among network scientists.	27
3.11	Cohesive subgraphs found by (a) Modularity, Truss, Discrete model, and Dense core. Modularity found coarse communities, and Truss and Dense core found more finer communities.	28
3.12	Detailed structures around (A) in Fig. 3.8 (c). Truss and dense core found cliques in (b) and (c), whereas failed to find star structured subgraph in (d) and (e).	29
4.1	Example of dendrogram of graph	34
4.2	Dendrograms of nodes in synthetic graph.	41
4.3	Dendrograms of nodes in karate-club network.	42
4.4	Dendrograms of nodes in political book network.	44
4.5	Dendrograms of nodes in collaboration networks.	45
5.1	Illustration of how to draw proposed graph diagram.	50
5.2	Centralities of nodes in synthetic graph scored by eigenvector centrality and dense core model	52
5.3	Illustration of karate club network	53
5.4	Illustration political book network	55
5.5	Illustration of collaboration network	56
5.6	Illustration of blog network	57
6.1	Communities found by Modularity, Extraction and ρ -dense core ($\rho = 0.5$). The value of ε is the strength of noise. The color of nodes expresses the membership of the nodes, while a gray node does not belongs to any community.	61
6.2	Change detection framework with double sliding windows	63
6.3	Base graphs of six scenarios of changes	64
6.4	Frequency of e-mails (a), change scores (VI) of ρ -dense core (b) and Extraction (c) in Enron data. Detected change points are red bars.	66
6.5	Graphs before/after change point (A) where the width of an edge represents the value of its weight. EMP, MGR, VP, P and DIR stand for Employee, Managers Vice President, President and Director. ρ -dense core extracts the change of communities surrounded by dotted rectangles. For example, the community in the bottom rectangle seems to have been collapsed after this change.	67

6.6	Frequency of phone-calls (a), change scores (VI) of ρ -dense core (b) and Extraction (c) in MIT student data. The detected change points are red bars.	69
6.7	Graphs before/after change point (A) detected by ρ -dense core. We can see that many small communities have changed.	70
6.8	Graphs before/after change point (D) detected by Extraction. Two small communities have been combined into a large community	70

List of tables

2.1	Summary of studies reporting core/periphery structures	8
3.1	Three cohesive measures in one clique and six graphs. A graph is more cohesive with a larger density, a larger degree, a shorter diameter, or a higher index of ρ dense core.	15
5.1	Ranking of scientists according to coreness measure	54
6.1	Summary of graph partitioning criteria	62
6.2	AUCs of community change detection tasks	65
6.3	List of major events in Enron Inc.	68
6.4	List of major events in School.	68

Chapter 1

Introduction

Throughout history social science has been dealing with inherently complex social systems made up of people influencing each other. In the last decade the social system expands remarkably by the development of telecommunication technologies such as telephone, e-mails, and social networking services. Understanding how people influence each other is of great importance to understand our social systems and social phenomena. *Social Network Analysis* (SNA) concerns the structures of the relationships among people represented by a graph called *social network*. In this network, individuals are expressed as nodes and their relationships are represented as edges. People tend to belong *communities* to share information and ideas, or to achieve their goals. Within a community, an individual undertakes roles to serve his/her community, and affects to other members. Therefore, understanding how individuals form a community is a substantial implication for improving our understandings of social systems. In this paper, we concentrate on the task of community discovery from social networks and aim to identify the leading individuals.

The study of social network has a history going back several centuries, but has recently been studied intensively, partly as a result of the development of mathematics and computers. Current social network analysis are based on a number of interdisciplinary studies including psychology, sociology, anthropology, physics and computer science, but is rooted in the study of "gestalt" in psychology. It is an organized system consisting of distinct "parts", yet is more than or different from the sum of its parts. Social psychology in this research tradition has stressed the importance of the influence of group organizations. To understand these relationships, Moreno invented a new diagram called *sociogram* [48] in which an individual is represented as a circle and the relationships between two individuals is represented as a line between two circles. Many of the leading gestalt theorists conducted a considerable amount of research on the problems of sociology and community dynamics. During 1930s, they explored community structures and their functions by using laboratory methods. At

the same time, sociologist and anthropologist have investigated the social relations of people across various fields such as urban/countryside, tribe, employees and employers. These investigations lead to the discovery that a community basically has three kinds of structures: *block structure*, *star structure*, and *core/periphery structure*. The block structured community is often seen in a non-hierarchical group whose members are familiar with each other. On the other hands, the star structured community is made up of one leading individual and many followers. The core/periphery structure is the mixture of the block and the star structures. It consists of a leading group called core and many followers called periphery.

In 1950s and 1960s, many studies in different traditions eventually brought together and contemporary social network analysis was born. Throughout this process, social network analysis has evolved to a more rigorous science with graph theory and statistics. There had been a concerted effort to develop mathematical models characterizing the notion of block and star structures. Luce introduced a *clique* [47] as a perfect model of a block structure in which every members are fully connected with the other members. The clique was soon found to be too strict to fit in the real-life community, and therefore, a number of *pseudo clique* have been developed by relaxing the clique model. Followed by this, Hoffman introduced a seminal model of the star structure called *n-star* [36]. The *n-star* consists of a node at the center and many peripheral nodes connecting only to the central node. Likewise the clique, the *n-star* is too strict, so that its relaxed variants have been introduced, and brought *centrality measures* [68] that assess how close a node is to the center of the whole graph.

Apart from the exploration of community models, the computational complexity is another major concern in the analysis of social networks. Since 1970, computers have been making significant contributions in the analysis of very complex social networks. A number of improvements have been made in the reduction of computational complexities [10, 74, 79]. In 1990s, the computational cost becomes one of the most serious problems since the social relation expanded dramatically with telecommunication technologies. One major breakthrough was made in 2000s, in which physicists began to analyse social networks. Ignoring, or unaware of the earlier work in social science, they brought the concept of physics to social network analysis such as *spin-glass*, *attraction/detracton*, *energy* and *field* [27, 34]. *Graph partitioning* is the one of the clustering techniques that physicists have developed for circuit design [31, 59, 82]. The graph cut methods are computationally less expensive than the traditional community models like pseudo cliques, and therefore, often used for the analysis of large networks [16, 35, 78, 82].

The investigation of the massive amount of social relations leads to another discovery that the real-life communities often have core/periphery structures [7, 21, 22, 55]. Followed by these investigations, Borgatti and Everett first introduced a formal model of core/periphery

structures [8]. Since the model is too strict to fit in real networks, as the same with the clique model, recent studies concentrate on relaxing this model [9, 44, 64, 66]. They focus on relatively small networks consisting of more or less 10 to 100 nodes and assumed that the network contains only one core community. However, there might be more than two core communities even in a small network. Indeed, recent studies have shown that distinct multiple core communities are observed in different parameter configurations [64]. In order to discover them, we need to set two partitions; one is the partition between communities (*inter-community partition*), and other is the partition between core-periphery (*intra-community partition*). Typically, these two partitions are set one by one so as to satisfy a criterion of partitions [86, 89]. For example, one approach is to divide a graph into communities, and then divide the communities into core and periphery [62]. Another approach is to find cores, and then find the corresponding peripheries [86, 89]. One problem is that the existing methods set these two partitions with the same criterion, ignoring the important differences in inter/intra-community partitions.

In this paper, we concentrate on developing another core/periphery community model taking this difference into account. In particular, we introduce a novel graph partitioning model termed as *dense core* to set inter/intra-community partitions such that

- Communities are well separated (inter-community partition),
- Core has clique-like structure (intra-community partition),
- These two partitions do not intersect.

We also introduce an algorithm to find optimal partitions for the relatively small graph ($\mathcal{O}(n^3)$ where n is the number of nodes) and a heuristic algorithm for the analysis of more larger graphs.

Dense core model is relevant to graph partitioning and pseudo cliques, which are two major traditions of community models. Chapter 2 describes the relevance and recent work on the analysis of core/periphery structures. Chapter 3 presents the dense core model, its properties, and illustrated the effectiveness in discovering multiple cores in networks. As suggested by the recent work, our experimental results suggest that the real-life community contains sub-peripheries lying between a core and its periphery. We therefore extend the proposed model to capture more realistic community structure in Chapter 4. We extend the proposed model by adding another partition other than inter-intra community partitions. It is shown in this chapter that the dense core model can capture multiple core/periphery structures even though their structural patterns such as density are different. Chapter 5 and 6 are devoted for illustrating the usefulness in practical applications. In Chapter 5, we introduce

a novel coreness. It is a centrality measure that can capture multiple cores in the network. In Chapter 6 is devoted for the analysis of dynamic social networks and the conclusion follows in Chapter 7.

Chapter 2

Related Work

2.1 Construction of Social Networks

Researchers have collected social data mostly from survey, recorded documents, interviewing and observations. Traditionally, survey data have been a major source of relational data [51]. By collecting a large number of standardized questionnaires systematically, researchers can correct a huge amount of desired social data. However, it costs too much to conduct survey many times. Documents such as newspaper, yearbooks and directories are rich sources of social relations [72]. Full list of members and their affiliations are obtained from yearbooks for industries, sectors, or political groupings. Other documents such as newspaper also include relational data on individuals. Unlike survey data, it is less expensive and not so overwhelming. In the last decade, these documents are available in electronic form which can be parsed automatically with computers. Interviewing is a way of collecting data in which researchers can control over the form of the data. Subjects are asked about their social relations such as the tie strength and positive or negative feeling to others. This is especially useful in the analysis of a specific individual or a community. Instead of asking directly to subjects, observational methods focus on their behavior such as the number of contacts and durations. The behaviors can also be collected through World Wide Web such as Talk pages in Wikipedia and questions and answer site like Stack Overflow.

Unlike physical data of natural science, these social data are ambiguous so that the construction of social networks involves a process of human interpretation. For example, in the study of friendship, researchers may ask respondents to identify their friends by asking question as “Please name your friends and score the degree of friendship”. However, the feeling of being friend would not be the same in each respondent. It is, at present, impossible to eliminate all ambiguity so that researchers need to carefully examine social data so that it is reliable enough for the analysis. For further discussion, see [73].

A social network consists of a set of nodes representing individuals and a set of edges representing relationship among them. An important question that often arises in social network analysis is to formalize the notion of relationship. Typically, three kinds of relationship have been considered: *tie strength*, *friendship/enmity* and *orientation*. The tie strength in social networks is a function of duration, emotional intensity and intimacy [50]. The strength is expressed as a weight of the edge. The friendship/enmity between two individuals is expressed as the sign of the weight, i.e., positive edges represent friendship and negative edges represent enmities between the corresponding individuals [12]. A social relationship is often unidirectional rather than bidirectional. For example, an individual may like another individual who does not like the individual. Although the weight, sign and orientation of edges represent detailed social relation among people, many studies focuses on an unweighted, unsigned, and undirected graph termed as *simple graph*. Although simple graph looses detail social relation, the structure itself holds sufficiently interesting information in many applications. In this paper, we also consider only a simple graph and assume that the existences of edges are carefully examined before proceeding the analysis, e.g., threshold the tie strength, omitting the negative edges and orientations.

2.2 Community structures

A community has a unique social norm, value, and subculture and hold the most important sources of a person's identify and sense of belonging [73]. It was appreciated that there were several forms of communities having distinct structures, which are classified into two classes: *Block structure* and *Star structure*. The *block structure* [47] is seen in the group of individuals without any hierarchical relation. Each member is familiar with each other and has equal powers. On the other hand, the *star structure* [33] is seen in the group with hierarchy. Within a star structure, one strong leader holds great popularity among his followers, and has strong power. Followed by the early studies, sociologists have attempted to formalize these structures with graph theory. As an ideal model of block structure, *clique* is introduced [47]. This is a perfect graph whose members are fully connected with the rest of members. However, this model cannot capture real-life block structures due to its state of perfections. Therefore, a number of *pseudo cliques* have been developed by relaxing the clique model. Typically, three kinds of cohesiveness are considered in the relaxation: *distance*, *degree* and *density*: (a) *distance* is the number of edges of the shortest path between two nodes; (b) *degree* is the number of edges a node has; (c) *density* is the ratio of the number of edges to the maximum possible number of edges. Relaxing the cohesiveness in terms of distance, *n-clique* [46] is developed, which is a maximal subgraph such that the distance of two nodes

is less than or equal to n (the path may step out the subgraph). A clique is a 1-clique since any two nodes are connected by an edge. If one edge is removed from a 1-clique, then it becomes a 2-clique. A k -plex [75] and a k -core [74] are obtained by relaxing the degree. A k -plex [75] is a maximal subgraph in which a node is nonadjacent to at most k nodes in the subgraph. As a complement model of the k -plex, a k -core [74] is a maximal subgraph in which each node is adjacent to at least k nodes in the subgraph. For example, a 4-node clique subgraph is a 3-core if no node of the clique is linked to three or more nodes outside the clique. By relaxing the density, ρ -dense [1] is developed. A connected subgraph is a ρ -dense if the density is greater than ρ . Hence, a 1-dense (100%-dense) subgraph is a clique. One problem of these pseudo cliques is that they are too relaxed from cliques so that they find subgraphs with excessively coarse structures. For example, being an n -clique ($n > 1$) does not mean that the subgraph is connected, because the definition allows that two separated components of the subgraph are connected through a node at outside of the subgraphs. In order to compromise between overly restrictive cliques and overly relaxed pseudo cliques, many variants have been proposed. An n -clan [54] and an n -club [54] were proposed as intermediate concepts between an n -clique and a clique. They adopt another measure of distance, *diameter*, i.e., maximum distance over all pairs of nodes. An n -club is a maximal subgraph with diameter n . An n -clan is an n -clique [46] with diameter n , so that every n -clan is an n -club. The n -club and n -clan are always connected if $n \geq 1$. As a stronger variant of k -core, k -truss [13] (also known as k -dense [69], or *Dense Neighborhood graph* [85]) is introduced. A k -truss is a maximal subgraph in which any pair of nodes have at least $k - 2$ common neighbors (nodes linked to the node by an edge). For example, a 4-node clique subgraph is a 4-truss subgraph if it is maximal in the set of subgraphs sharing the same property. A remarkable nature of a k -truss is that the enumeration of all k -truss subgraphs needs only a polynomial time in the number of nodes. These pseudo cliques are very useful for finding communities with block structures.

The seminal model n -star of star structure was proposed by Hoffman in 1965. This is a subgraph consists of one node at center and n peripheral nodes connecting only to the central node. Likewise clique, this definition is too strict to find more complex star structured communities. The n -star model is then extended to *centrality measures* to assess the closeness of nodes to the center of a graph. There are three kinds of centrality depending on the scale of structures to be considered: local, global and community-level centrality. A node is locally central if it has large number of edges with others, while is globally central if it positions at the center of the whole network [26, 30, 68]. As a middle of these scales, a node is central at the community-level if it has many edges with its community members and positions at the

Table 2.1 Summary of studies reporting core/periphery structures

Category	Network	Core/Periphery	Study
Social Network	Conversation	Elite/New comer	[8]
	Volunteer organizations	Leaders/Non-leaders	[52]
	Co-citation among journals	Popular/Specific journal	[8]
	Citation on micro-blogs	Opinion Leader/Followers	[25, 39]
	Migrants flow among cities	Capital/Countryside	[44]
	Scientific citation	Seminal/New study	[22, 55]
Ecological Networks	Food web among species	General/Special species	[7]
	Plant/Insect Network	Essential/non-essential species	[67]
Biological Networks	Protein-Protein Interaction	General/organ-specific function	[17]
	Relation among Metabolites	Specialized/variable metabolisms	[49]
	Neural network	Backbone/Task related neurons	[28]
Engineered Networks	Autonomous system	Hub/-	[90]
	Transportation	Urban/countryside	[60]
	Power grid	Supplier/Consumer	[52]
	World Wide Web	Popular/-	[90]

center of the community [8, 86]. Among them, the community-level centrality measures are useful for identifying powerful individual within a community.

2.3 Core/Periphery Structures

Block structure is characterized by the density of nodes, while the star structure is characterized by the position of nodes. In this sense, the star and block structure are important complementary structures. Recently, the investigations of the massive amount of social relations lead to another discovery that there is a mixed structure of these two structure called core/periphery structures *core/periphery structures* [7, 21, 22, 55]. The core/periphery structure consists of two classes of nodes, one class is a *core*, and the other class is a *periphery*. This structure is seen in the group with a leading subgroup such as broad directors and opinion leaders. The core/periphery structure was firstly reported [19] in 1941, and since then have been found in various networks (see Table 2.1).

According to these studies, the core is found to be a leading subgroup that undertakes some general tasks and roles, and peripheral nodes are those who follow the core group and do some specific tasks. An idealized model of core/periphery structure was firstly introduced by Borgatti & Everett [8]. This model consists of a single partition between cores and peripheries such that core is a clique and peripheries connect to all the core nodes, but no connections with themselves. This perfect structure is merely seen in real-life networks due its state of perfections. One practical way, as Borgatti & Everett suggested, is to seek the most probable partitions of core and periphery that maximizes a likelihood of core/peripheries structures. Following this study, several likelihood functions have been introduced to capture

more realistic core/periphery communities [9, 64, 66]. One limitation of these models is that they only find single core and periphery. Real-life networks may contain more than two core/periphery structures, especially for a large network. Indeed, recent study reported that multiple distinct core/periphery structures are observed [64].

In order to find the multiple core/periphery structures, we need to find two refinement partitions: *inter-community partition* and *intra-community partition*. *Inter-community partition* divides a graph into communities, and *intra-community partition* divide each community into core and peripheries. Traditionally, the problem of partitioning a graph, *graph partitioning problem* [82], has been studied for circuit design in which the number of communities are given by users. In social networks, however, the number and the size of communities are not clear in advance. For this issue, Newman introduced *modularity* [56]. Modularity determines the size and the number of communities according to the complexity of the graph. The modularity and its variants [6, 84, 88] tend to divide a graph into well-balanced communities in terms of the number of edges and nodes. Recently, *Extraction* [89] and *Graph scan* [83] are developed to find the most densest community in the graph. Unlike the ordinal partitioning methods, it puts its priority in finding the densest subgraphs in the graph like cores, rather than balanced subgraphs. As an extension of graph partitioning problem, finding of these refinement partitions have been studied as a *hierarchical graph partitioning problem* [27, 40, 73]. A refinement partitions are found by recursively dividing a graph into elements according to a graph partitioning model until all elements are singleton nodes. As we have seen, there are various methods and models for partitioning a graph [43, 56, 62, 65]. However, in a core/periphery structure, they often produce too coarse cores, or divide a core into fragmented sub-communities. This is because they do not take the important difference between two partitions. In particular, the inter-community partitions are between weakly connected communities, while the intra-community partitions are between cores and peripheries which may be densely connected. As an example, let us consider the graph shown in Fig. 6.1 where two core/periphery communities are adjacent to each other. Coarse cores are found with discrete model [8], extraction[89] and truss [13].

We, therefore, introduce a novel model called *dense core* for discovering multiple cores in social networks. As we shall see, it can discover two kinds of communities: well-separated communities and (almost) perfect subgraphs. On the basis of this unique nature, we have shown its effectiveness in identifying multiple core/periphery structures.

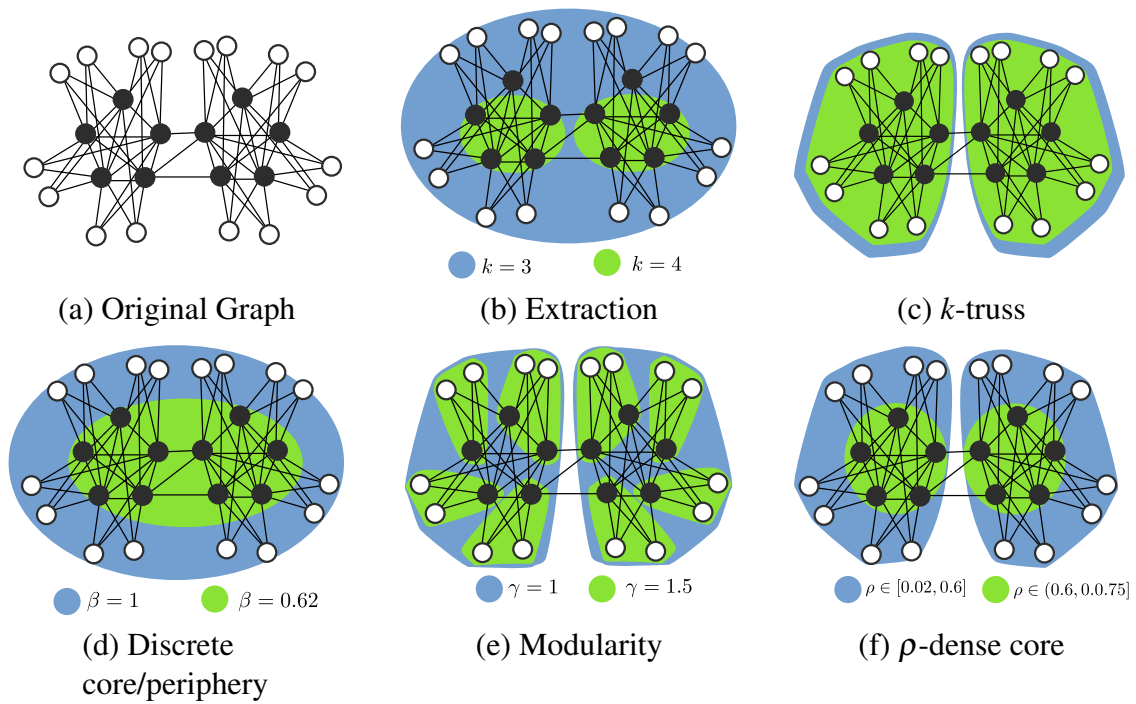


Fig. 2.1 Comparison of core/periphery models. (a) The original graph consists of two sub-cliques. The Extaction (b), relaxed core/periphery (d) sets natural intra-community partition between core and periphery, but failed inter-community partition. The k -truss (c) and Modularity (e), on the other hands, sets natural inter-community partition, but, failed intra-community partitions. Among them, only ρ -dense core (f) sets inter/intra-community partitions naturally and uncovers two core/periphery communities.

Chapter 3

Dense core model

3.1 Preliminaries

A social network can be modeled as a graph $G = (V, E)$ with a set of nodes $V = \{1, 2, \dots, n\}$ and a set of edges among V . We assume that there is no self-loop edge. It is convenient to represent a graph as an *adjacency matrix* $\mathbf{A}$ defined as

$$\mathbf{A} = (a_{ij}), \quad a_{ij} := \begin{cases} 1 & (\{i, j\} \in E) \\ 0 & \text{otherwise} \end{cases}. \quad (3.1)$$

It should be noted the adjacency matrix is symmetric since $\{i, j\} = \{j, i\}$.

We denote the *within-subset weight* by $W(U_i)$ and *between-subset weight* by $B(U_i, U_j)$:

$$W(U) := \sum_{\substack{r, s \in U \\ r < s}} a_{rs}, \quad \text{and} \quad B(U_i, U_j) := \sum_{r \in U_i, s \in U_j} a_{rs}. \quad (3.2)$$

We also denote the density of edges inside U by $\eta(U)$ defined as

$$\eta_w(U) := \frac{W(U)}{|U|(|U| - 1)/2}, \quad \forall |U| > 1, \quad (3.3)$$

and $\eta(U) = 0$ when $|U| \leq 1$.

A partition $\mathcal{P} = \{U_1, U_2, \dots, U_K\}$ over V is defined as a family of subsets of V such that $U_i \cap U_j = \emptyset$ for all $i \neq j$ and $\cup_{U \in \mathcal{P}} U = V$. For two partitions $\mathcal{P}_1$ and $\mathcal{P}_2$, we denote $\mathcal{P}_1 \preceq \mathcal{P}_2$ if $\mathcal{P}_1$ is the refinement of the partition $\mathcal{P}_2$.

3.2 Model

In a core/periphery structure, a graph constitutes of communities, and each community contains highly cohesive subgraphs. This hierarchical structure can be expressed with two partitions $\mathcal{P}_0$ and $\mathcal{P}_1$ such that $\mathcal{P}_0$ divides the graph into communities (inter partition), and $\mathcal{P}_1$ divides each community into a core and a periphery (intra partition). In addition to that, these two partitions do not interest, that is $\mathcal{P}_1$ is the refinement of the partition $\mathcal{P}_0$ ($\mathcal{P}_1 \prec \mathcal{P}_0$). The inter partitions and the intra partitions have different characteristics; inter partition cuts sparse connections between communities, while the intra partition cuts dense connections between cores and peripheries. Therefore, our criteria for these partitions are as follow:

- Each community is loosely connected with each other
- Core has a block structure and periphery does not.

The problem of finding partitions $\mathcal{P}_0$ and $\mathcal{P}_1$ are formulated as a *hierarchical graph partitioning problem* [27, 40, 73].

$$\max_{\mathcal{P}_0} \sum_{U \in \mathcal{P}_0} q(U; \rho_0) \text{ and } \max_{\mathcal{P}_1} \sum_{U \in \mathcal{P}_1} q(U; \rho_1), \text{ such that } \mathcal{P}_2 \prec \mathcal{P}_1, \quad (3.4)$$

where $q : 2^V \rightarrow \mathbb{R}$ is a quality function, and ρ_0 and ρ_1 are parameter of q . The task then we face is to design the quality function q so as to meet our criteria.

A number of quality functions have been developed to discover communities and its hierarchy. *Modularity* [57] is one of the well-known criteria in social network analysis. The quality function is defined by

$$q_{\text{mod}}(U; \rho) := \sum_{\substack{i, j \in U \\ i < j}} \left[a_{ij} - \rho \frac{d_i d_j}{2W(V)} \right], \quad (3.5)$$

where d_i is the degree of node i defined by $d_i = \sum_{j \in V} a_{ij}$ and ρ ($0 \leq \rho \leq 2W(V)$) is *resolution parameter*. The first term counts the number of edges inside U , giving a higher quality for a larger and denser subgraph. On the other hands, the second term gets larger when two nodes have large degree, giving repulsion force between hubs. The resolution parameter controls the balance between these two terms and larger ρ produces finer communities. It should be noted that modularity tends to divide hubs into different communities due to the repulsion force. Therefore, cores which have many edges tend to be divided. In order to glue hubs in the same community, we remove their repulsion forces and introduce our quality function

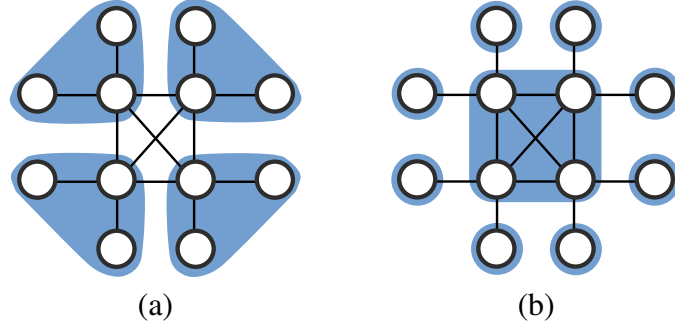


Fig. 3.1 Communities found with (a) Modularity ($\rho = 1$) and (b) Proposed quality functions ($\rho = 0.5$). A community is the nodes in the same colored region. Modularity divides cliques into different elements, while dense core does not.

[42] as

$$\begin{aligned}
 q(U; \rho) &:= \sum_{\substack{i, j \in U \\ i < j}} [a_{ij} - \rho] \\
 &= W(U) - \frac{\rho}{2} |U|(|U| - 1),
 \end{aligned} \tag{3.6}$$

where ρ is resolution parameter ($0 \leq \rho \leq 1$). Unlike the modularity, our quality function does not give large penalties to the connections between hubs. The first term is the same with the modularity, but the second term depends only on the size of subgraph, rather than degree of nodes. The quality function gets higher if the subset is dense and large. Hereinafter we call (3.6) as the *dense core criterion*. Fig. 3.1 shows the communities found with modularity and proposed criterion, respectively. The hub nodes are divided in modularity, but are in the same community in the dense core criterion.

3.3 Property

The proposed quality function has many desired properties for the criteria of inter/intra-community partitions. Here we show two important properties for identifying core/periphery structures [42].

Theorem 1 (Communities found with dense core criterion). *The optimal partition $\mathcal{P} = \{U_1, U_2, \dots, U_K\}$ of the dense core criterion has following two properties.*

1. Any two elements U_i, U_j ($i \neq j$) are connected with density at most ρ . That is

$$\max_{\substack{U_i, U_j \in \mathcal{P} \\ i \neq j}} \frac{B(U_i, U_j)}{|U_i||U_j|} \leq \rho \quad (3.7)$$

2. For each element $U \in \mathcal{P}$ with size $|U| \geq 2$, any subset Z of U connects to the rest $U \setminus Z$ with density at least ρ . That is,

$$\min_{Z \subset U} \frac{B(Z, U \setminus Z)}{|Z||U \setminus Z|} \geq \rho, \forall U \in \mathcal{P} \quad (3.8)$$

Proof. It follows from the optimality of $\mathcal{P}$ that the merge of any two communities necessarily decreases the partition quality. That is,

$$\begin{aligned} q(U_i \cup U_j) &\leq q(U_i) + q(U_j), \forall U_i, U_j \in \mathcal{P}, i \neq j \\ \iff B(U_i, U_j) - \rho|U_i||U_j| &\leq 0, \forall U_i, U_j \in \mathcal{P}, i \neq j \\ \iff \max_{\substack{U_i, U_j \in \mathcal{P} \\ i \neq j}} \frac{B(U_i, U_j)}{|U_i||U_j|} &\leq \rho \end{aligned}$$

Likewise, any dichotomies of any element $U \in \mathcal{P}$ do not increase the value of q . Therefore, we have

$$\begin{aligned} q(U) &\geq q(Z) + q(U \setminus Z), \forall Z \subset U \in \mathcal{P} \\ \iff B(Z, U \setminus Z) - \rho|Z||U \setminus Z| &\geq 0, \forall Z \subset U \in \mathcal{P} \\ \iff \min_{Z \subset U} \frac{B(Z, U \setminus Z)}{|Z||U \setminus Z|} &\geq \rho, \forall U \in \mathcal{P} \end{aligned}$$

□

It follows from Theorem 1.1 that the dense core criterion produces well-separated communities with a small value of ρ , which meets our criterion of inter-community partitions. Theorem 1.2 suggests that a community should be dense everywhere. It will be later shown that the community tends to be a clique-like subgraph in which every node connects almost equally, which meets our criterion of intra-community partition.

Let us see what Theorem 1.2 implies in more detail. We note that the left hand side of (3.8) is the *ratio cut objective* [32]. It tries to divide a subset U into “balanced” subgraphs in terms of the number of nodes. We call the minimum value of the objective as *minimum cut*

Table 3.1 Three cohesive measures in one clique and six graphs. A graph is more cohesive with a larger density, a larger degree, a shorter diameter, or a higher index of ρ dense core.

Cohesive Measure	(a)	(b)	(c)	(d)	(e)	(f)	(g)
Diameter	1	2	3	2	2	3	7
Minimum Degree	7	3	2	1	1	3	1
Density	1	12/28	10/28	13/28	7/28	13/28	7/28
Minimum cut	1	1/3	1/4	1/7	1/7	1/16	1/16

and denote

$$\psi(U) := \min_{Z \subset U} \frac{B(Z, U \setminus Z)}{|Z||U \setminus Z|}.$$

Now, we define the concept of ρ -dense core as follow [42].

Definition (ρ -dense core). A subset of nodes U ($U \subseteq V$, $|U| \geq 2$) is said to be ρ -dense core if and only if the minimum cut of U is greater than or equals to ρ , that is $\psi(U) \geq \rho$.

The ρ -dense core is a homogeneous variant of ρ -dense concept. The ρ -dense requires for U to be “dense” in the total number of edges, while the ρ -dense core requires for U to be dense everywhere. As will be shown, a ρ -dense core is necessarily a ρ -dense, but the reverse does not hold. Note also that a subgraph of a ρ -dense core is not necessary a ρ -dense core. For example, suppose a graph of five nodes forming a ring (each node have two edges). Although this graph is a $1/3$ -dense core, any 4-node subgraph of it is only a $1/4$ -dense core. This is different from the case of cliques. We hereafter use the word “ ρ -dense core” even for a subgraph whose node set is ρ -dense core.

Some examples are shown in Table 3.1. We note that the minimum cut is the maximum value of ρ for which the graph is a ρ -dense core. In every measure, clique (a) achieves the best score. The graph (b) achieves high scores in all indices. The minimum degree measure is greatly affected by the local structure but is not affected by the global structure. On the other hand, the density and diameter measures are not affected by the local structures but are affected by the global structures. The minimum cut gives a more detailed evaluation than the others. It evaluates them in the order of (a) > (b) > (c) > (d) = (e) > (f) = (g). The ρ -dense core is akin to ρ -dense but is more sensitive to the degree of homogeneity (Compare the values in (b) and (d) for example.). Therefore, if we put priority on exploration of cliques or a subgraph of mutually connected nodes, the index of the dense core is the most promising candidate because of its higher resolution. These examples suggest that the minimum cut can be seen as a “cliqueness” measure. Now, in terms of the words “degree”, “density” and

“distance”, we describe what ρ -dense core is and how it is related to the other pseudo cliques [42].

Property 1 (Degree, Density and Diameter of ρ -dense core). *Let $U \subseteq V$ be a ρ -dense core. Then the following holds.*

1. *The minimum degree of U is greater than or equal to $\rho(|U| - 1)$.*
2. *The density of U is greater than or equal to ρ .*
3. *The diameter $D_{\max}$ of U is bounded by*

$$D_{\max} \leq \begin{cases} \lceil 1/\rho \rceil & (\rho \geq 0.5) \\ 3 \lfloor 1/\rho \rfloor - 1 & (\rho < 0.5) \end{cases} \quad (3.9)$$

4. *U has vertex expansion ρ .*

Proof.

1. By the definition of ρ -dense core, the density between a node $v \in U$ to the others $U \setminus \{v\}$ is greater than or equal to ρ , that is, $B(\{v\}, U \setminus \{v\}) \geq \rho(|U| - 1)$.
2. Since $W(U)$ is a half of the sum of degrees of all nodes and the degrees are all larger than $\rho(|U| - 1)$ from Property 1.1, we have $\eta(U) = W(U)/(|U|(|U| - 1)/2) \geq \rho$.
3. This last property is a direct consequence of Theorem 5 in [11] from which we can first know the diameter of a k -core. The diameter D of a k -core with n nodes is bounded by

$$D \leq \begin{cases} \left\lceil \frac{n}{k+1} \right\rceil & (n \leq 2k+1) \\ 3 \left\lfloor \frac{n}{k+1} \right\rfloor - 1 & (\text{otherwise}) \end{cases}$$

In terms of the word “ k -core”, a ρ -dense core of n nodes is a $\rho(n-1)$ -core from Property 1.1. Therefore, substituting k with $\rho(n-1)$, we have

$$D \leq \left\lceil \frac{n}{\rho(n-1)+1} \right\rceil = \left\lceil \frac{n}{\rho n + 1 - \rho} \right\rceil \leq \left\lceil \frac{n}{\rho n} \right\rceil = \left\lceil \frac{1}{\rho} \right\rceil \quad (\rho \geq 0.5),$$

where we have made use of the relation $1 - \rho \geq 0$. Similarly, we have the other bound.

4. Let us denote Z as the subgraph of U , and ∂Z as the boundary of nodes in Z having edges with the nodes in $U \setminus Z$. The vertex expansion of U is defined by

$$\min_{Z \subset U, 1 \leq |Z| \leq \frac{|U|}{2}} \frac{|\partial Z|}{|Z|},$$

which is the portion of gate keeper nodes in Z that intermediate Z and $U \setminus Z$. The maximum possible number of edges between $U \setminus Z$ and ∂Z is $|\partial Z||U \setminus Z|$ so that we have

$$|\partial Z||U \setminus Z| \geq B(Z, U \setminus Z) \iff |\partial Z| \geq \frac{B(Z, U \setminus Z)}{|U \setminus Z|}.$$

By definition of ρ -dense core, we have $B(Z, U \setminus Z) \geq \rho|Z||U \setminus Z|$, which leads to $|\partial Z| \geq \rho|Z|$. Therefore, we have

$$\min_{Z \subset U, 1 < |Z| < \frac{|U|}{2}} \frac{|\partial Z|}{|Z|} \geq \min_{Z \subset U, 1 < |Z| < \frac{|U|}{2}} \frac{|Z|}{|Z|} \rho = \rho.$$

□

Note that all values of degree, density and distance of a ρ -dense core are bounded only by the value of ρ . As the value of ρ increases, the subgraph has a shorter diameter, a better expansion, larger minimum degree and larger density simultaneously. And therefore, by increasing the value of ρ , dense core is close to a clique. The proof is omitted because it is straightforward.

Property 2 (Relevance to other pseudo cliques). *A graph U of ρ -dense core ($U \subseteq V$) has the following relations with other pseudo cliques:*

1. *A 1-dense core is equivalent to a clique.*
2. *U is a (not maximal) k -core with $k = \rho(|U| - 1)$.*
3. *U is a ρ -dense.*
4. *U is an (not maximal) n -clan of $n = \lceil 1/\rho \rceil$ if $\rho \geq 0.5$, otherwise $n = (3\lceil 1/\rho \rceil - 1)$.*
5. *U is ρ -vertex expander graph.*

In summary, dense core criterion is shown to produce well separated communities with a small value of ρ (Theorem 1.1), and also produces subgraphs that are dense everywhere (Theorem 1.2). We also show that the found community has short diameter, large degree and high density and thus has a block structure (Property 1). These properties meets the criteria of inter intra-community partitions.

3.4 Algorithm

We have shown that the optimal partitions agrees with the criteria of inter/inter-community partitions. The most naive way to find the best partitions is to examine all possible partitions over V . Such an exhaustive search is, however, practically infeasible even for a modest number of nodes. This is because the number of possible partitions grows exponentially with respect to the number of nodes. In this section, we introduce an alternative way to find the best partition under the assumption that the graph is sparse [42].

Suppose that we have a family of candidate elements $\mathcal{C} = \{C_1, C_2, \dots, C_L\}$ of an optimal partition $\mathcal{P} = \{U_1, U_2, \dots, U_K\}$. That is, $\mathcal{P} \subseteq \mathcal{C}$. Then, the graph partitioning problem is cast into *set packing problem* [5].

$$\max_{\mathbf{x}=[x_1, x_2, \dots, x_L]} \sum_{i=1}^L q(C_i) \cdot x_i, \quad (3.10)$$

subject to

$$\forall x_i \in \{0, 1\} \quad (3.11)$$

$$\mathbf{R}\mathbf{x} = \mathbf{1}, \quad (3.12)$$

where x_i is the binary indicator and

$$R_{ji} = \begin{cases} 1 & (j\text{th node belong to } C_i) \\ 0 & (\text{otherwise}) \end{cases}. \quad (3.13)$$

We note that x_i expresses whether C_i is included to the optimal partition $\mathcal{P}$ or not and (3.13) is the constraint to produce partition over V . By this reformulation, efficient state-of-the-art LP solvers are available to the graph partitioning problem. The task then we face is to enumerate the candidate element $\mathcal{C}$. In order to solve (3.10) in a short time, the candidate set $\mathcal{C}$ should be as small as possible. At the same time, the enumeration of $\mathcal{C}$ should be completed in a short time as well. An appropriate candidate meeting these criteria is ρ -dense. It follows from Theorem 1.2 that the optimal partition consists only of ρ -dense core or singleton nodes. From Property 2, ρ -dense core is necessarily ρ -dense, that is the set of ρ -dense includes the set of ρ -dense core. In addition, there is an efficient algorithm [81] finding all ρ -dense subgraphs. We, therefore, adopt the set of ρ -dense subgraphs as the candidate set of ρ -dense cores. The enumeration algorithm [81] of ρ -dense starts from the set of singleton nodes and repeat adding another node as long as the density is higher than ρ . It runs in time order $\mathcal{O}(2^l)$ if the maximum node size of ρ -dense subgraphs is l . For a sparse

Algorithm 1 DENSE CORE PARTITIONING**Input:** Graph $G = (V, E)$, Resolution parameter $\rho \geq 0$ **Output:** Partition $\mathcal{U}^*$ consisting of ρ -dense cores and some singleton sets

- 1: Call SEARCHDENSESUBGRAPH(G, ρ) to obtain $\mathcal{C}$
- 2: Construct matrix $\mathbf{R}$ showing $\mathcal{C}$

$$R_{ji} = \begin{cases} 1 & (j\text{th node belong to } C_i) \\ 0 & (\text{otherwise}) \end{cases}$$

- 3: Solve the following binary integer programming over $\mathbf{x} = (x_1, x_2, \dots, x_{|\mathcal{C}|})$ ($x_i \in \{0, 1\}$):

$$\max_{\mathbf{x}} \sum_{i=1}^{|\mathcal{C}|} q(C_i) \cdot x_i, \quad \text{subject to} \quad \mathbf{R}\mathbf{x} = \mathbf{1}.$$

- 4: Construct $\mathcal{U}^*$ by the indices of the maximizer $\mathbf{x}^*$

Subroutine SEARCHDENSESUBGRAPH [81]**Input:** Graph $G = (V, E)$, Density $\rho \geq 0$ **Output:** All ρ -dense subgraphs

- 5: Initialize $\mathcal{U} = \{i \mid i \in V\}$
- 6: **repeat**
- 7: **for** each $U \in \mathcal{U}$ and $v \in V \setminus U$ **do**
- 8: **for** each neighbor $v \in V \setminus U$ of U **do**
- 9: **if** $\eta(U \cup v) \geq \rho$ and $\{U \cup v\}$ is unique in $\mathcal{U}$ **then** Add $\{U \cup v\}$ to $\mathcal{U}$
- 10: **end if**
- 11: **end for**
- 12: **end for**
- 13: **until** no changes in $\mathcal{U}$

graph, the l is small. Therefore, the algorithm would terminate within a reasonable time. The pseudo code of the algorithm for finding the optimal partition is described in Algorithm 1. In practice, we made use of *CPLEX* [15] to solve (3.10), a state-of-the-art solver for integer programming, at Step 3 in Algorithm 1.

3.5 Experiment

3.5.1 Synthetic graph

In order to examine if the core/periphery communities are extracted or not, we analyzed two graphs forming two core/periphery structures as shown in Fig. 3.2. Each graph includes two five node cliques (core) and many peripheral nodes. We compared the dense core model

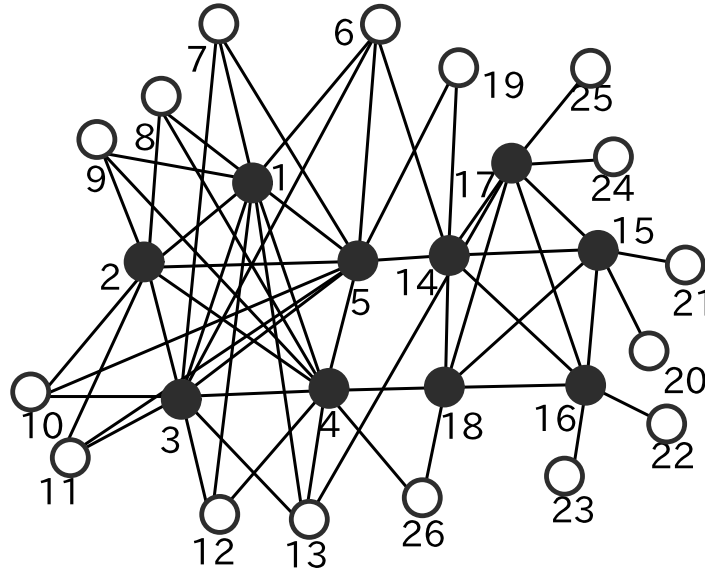


Fig. 3.2 Graph of two cores (black circles) and peripheries (white circles).

with *modularity* [57], *truss* [14] and *discrete model* [8]. The parameters of these models are set as follows. For truss, we examined all possible parameters and showed its result. For modularity and dense core, we examined various parameters and selected the best results in which inter/intra-community partitions are set most appropriately. discrete model does not have any user-defined parameter. To optimize the criterion of modularity, we adopt a well-known heuristic [57].

The result is shown in Fig. 3.3. We observed that modularity divided the graph into communities, but failed to divide core and peripheries. Contrarily, discrete model and truss divided the graph into cores and peripheries, but failed to divide into communities. Dense core divided the graph into communities first, and further divided each community into cores and peripheries.

We then rewired the edges of the graph randomly and investigated whether the cores are extracted or not. The edges are rewired according to the following random process generating core/periphery structures [8]:

1. A pair of core nodes is connected with probability $1 - p$.
2. A core and a peripheral node are connected with probability p .
3. No peripheral node connect to other peripheral nodes.

We note that the cores get sparser and peripheries connects denser to the cores at the same time as the value of p increases, that is the detection of core communities are hard. We generated 100 graphs and counted the number of partitions dividing core communities and

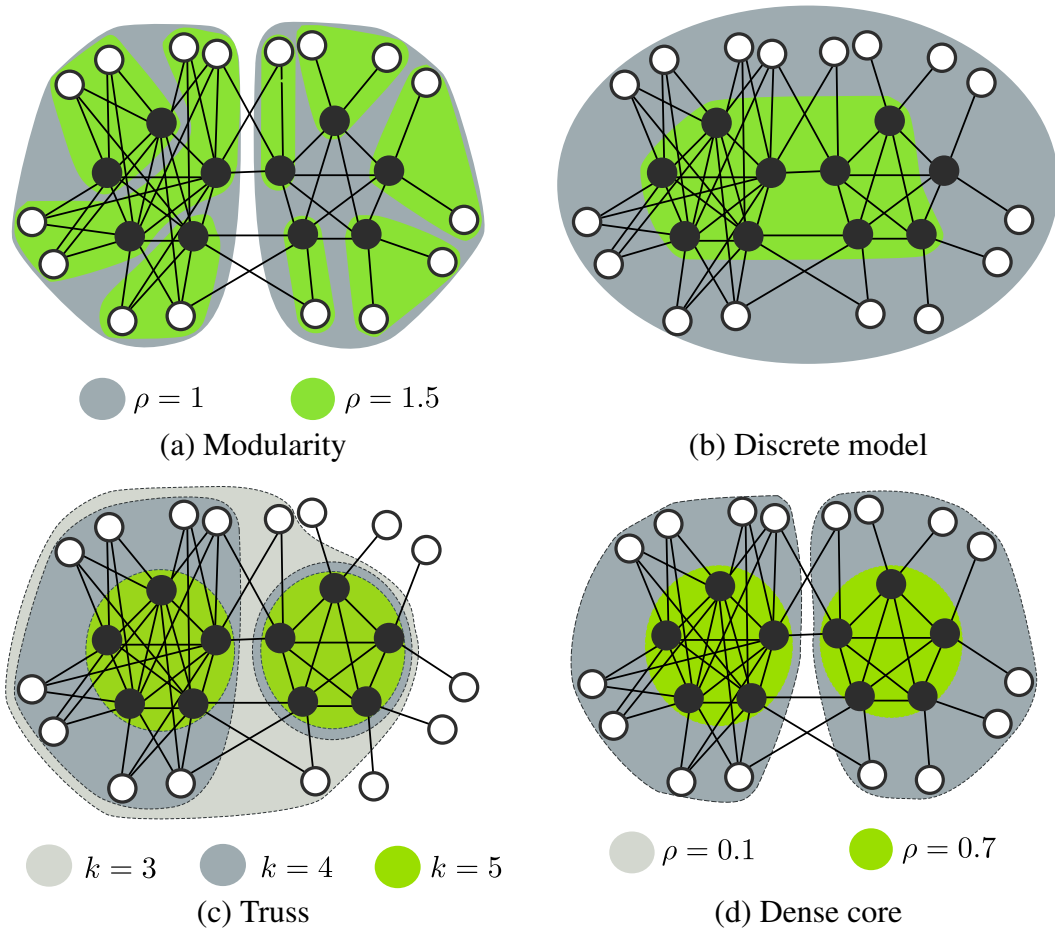


Fig. 3.3 Communities found with (b) Modularity, (c) Discrete Model, (d) Truss and (e) Dense core. Nodes in the same community is surrounded by the same colored region. The parameters of the models are shown at the bottom of each figure.

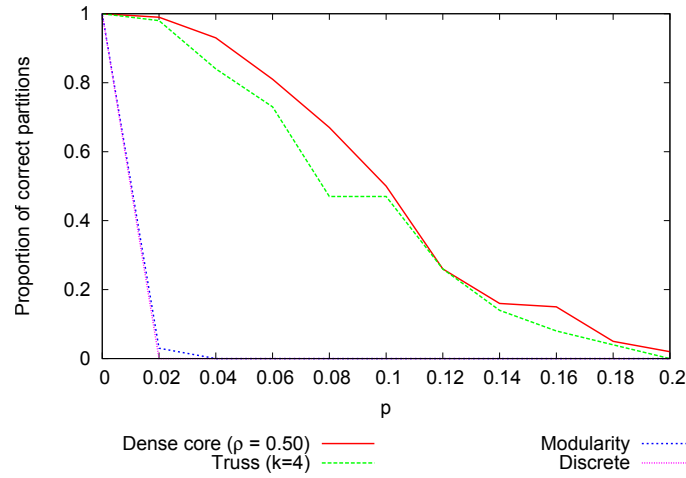


Fig. 3.4 Comparison of the proportion of correct partitions.

peripheries correctly. We set the parameters of the models such that the number of correct partitions were the highest. The results are shown in Fig. 3.4 in which the horizontal axis is the value of p and the vertical axis represents the proportion of the correct partitions among 100 partitions. We observed that the proportions in all models started from one and decreased as the value of p increased. Dense core model found the core communities more often than the other models in all the values of p .

3.5.2 Analysis of Real-life network

Next, we compared the community models on three real-life networks: *karate club*, *political book network*, and *science collaboration network*.

Fig. 3.5 shows *karate-club network*, in which a node is an individual and two individuals are connected if they are familiar with each other. The node #34 and #1 conflicted in the price of karate lesson and other members followed either. We investigated all possible parameters of truss, and several parameters of dense core, and modularity. We selected the parameter which yielded results observed frequently. The results are shown in Fig. 3.6. All models except discrete model divided the node #34 and #1 into different communities. Dense core model and truss found similar communities, but dense core model found finer and more cohesive communities.

Next, we analyzed the network of political books consisting of 105 nodes representing books on the United States presidential election in 2004 [57]. Two nodes are connected when the corresponding books are frequently purchased by the same customer. Each book has its own political agenda of liberal, conservative or centrist, which was judged on the basis

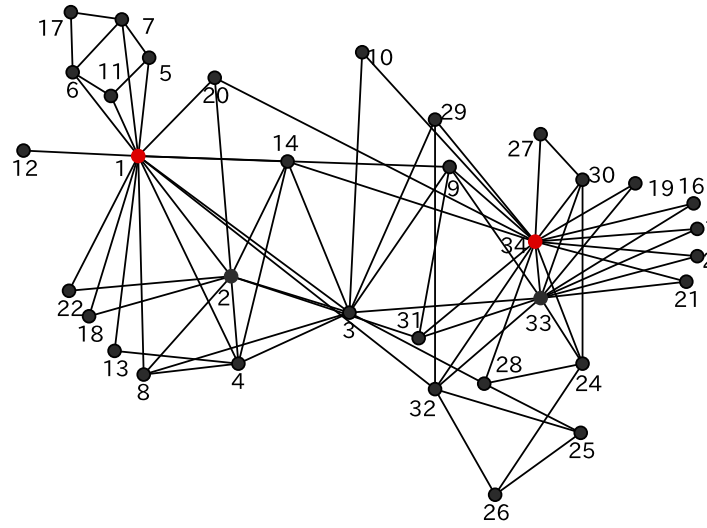


Fig. 3.5 Graph of Zachary's karate club. Node # 1 and # 34 are the two leading individuals in this club.

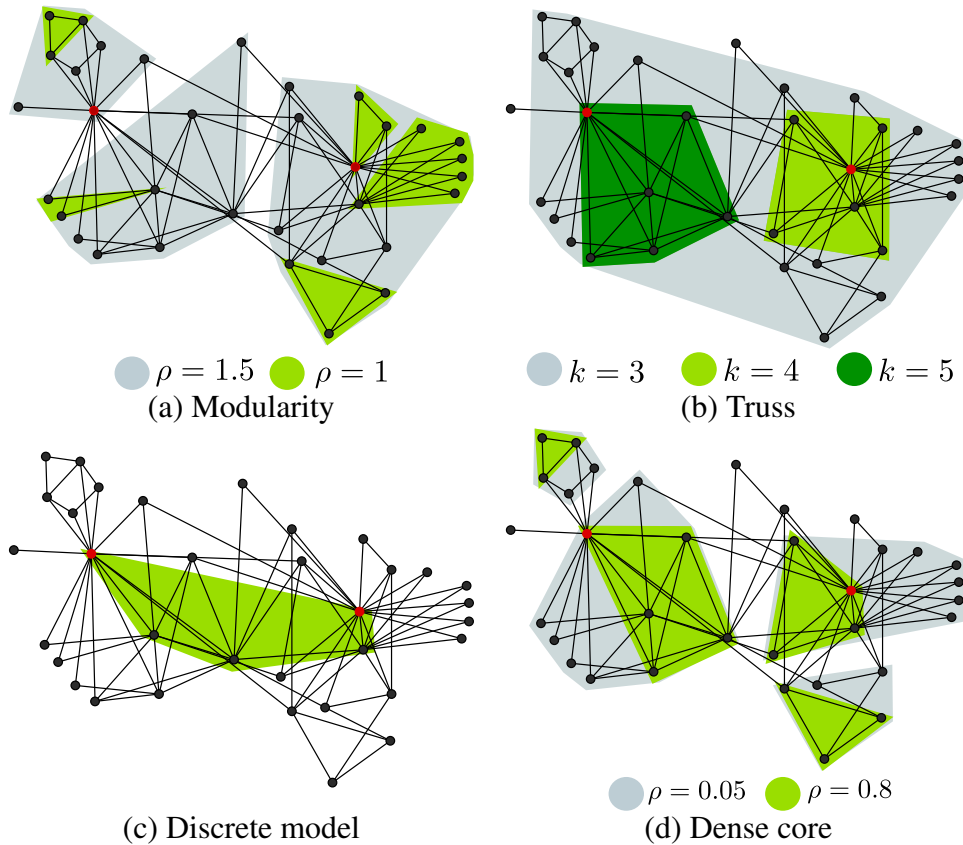


Fig. 3.6 Cohesive subgraphs found by Modularity, Truss, Discrete model, and Dense core.

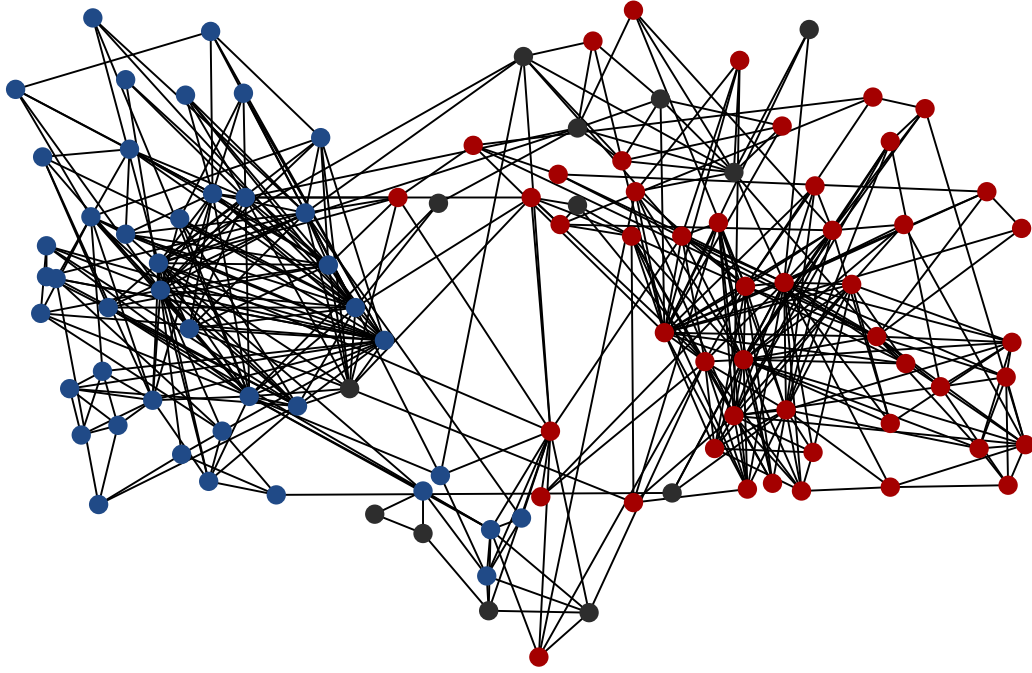


Fig. 3.7 Graph of political books on the United States presidential election in 2004. The color of each node represents the political alignment: liberals (blue), conservatives (red) and centrists (black).

of descriptions and reviews [57]. The graph is shown in Fig. 3.7 where the colors of nodes represent conservative (red), centrist (black) and liberal (blue). The results are shown in Fig. 3.8. All models except discrete model divided the graph into several communities. We observed that two of them consist of either liberals or conservatives, and the rest consists of both. Modularity and truss extracted four relatively large groups of liberal, conservatives and centrists, while the dense core found many small cores within the communities.

Fig. 3.9 shows the detailed structures of the liberal and conservative communities extracted with dense core and truss. We observed that the truss found large core communities surrounded by many peripheral nodes. Among the peripheries, we found sub-peripheries that connects strongly with the core and also the other peripheries. On the other hand, dense core model found two finer core communities than truss, but did not find two different peripheries. As we shall see, we will extend the dense core model to capture the sub-peripheries.

Next, we analyzed a network of collaborations among 379 scientists studying network theory and experiments [57]. In this network, a node represents a scientist and two nodes are connected if they coauthored. We decreased the value of ρ from 0.9 to 0.3 by step of 0.1 and found the optimal partition by ρ -dense cores using Algorithm 1. The enumeration of a

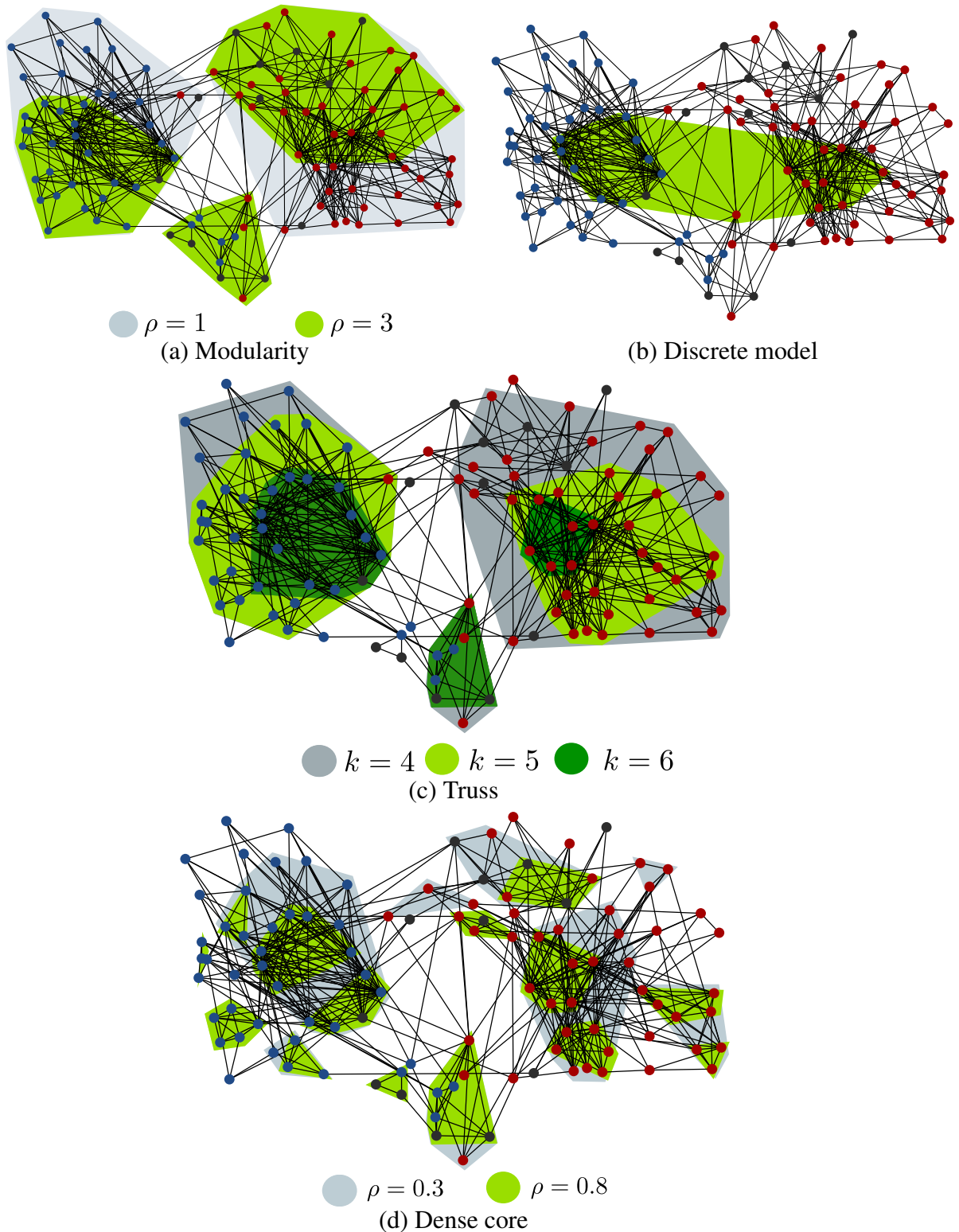


Fig. 3.8 Cohesive subgraphs found by (a) Modularity, Truss, Discrete model, and Dense core. All models except Discrete divided the graph into one conservative community, one liberal community and centrist communities. Truss and Dense core also found very cohesive subgraphs in these communities.

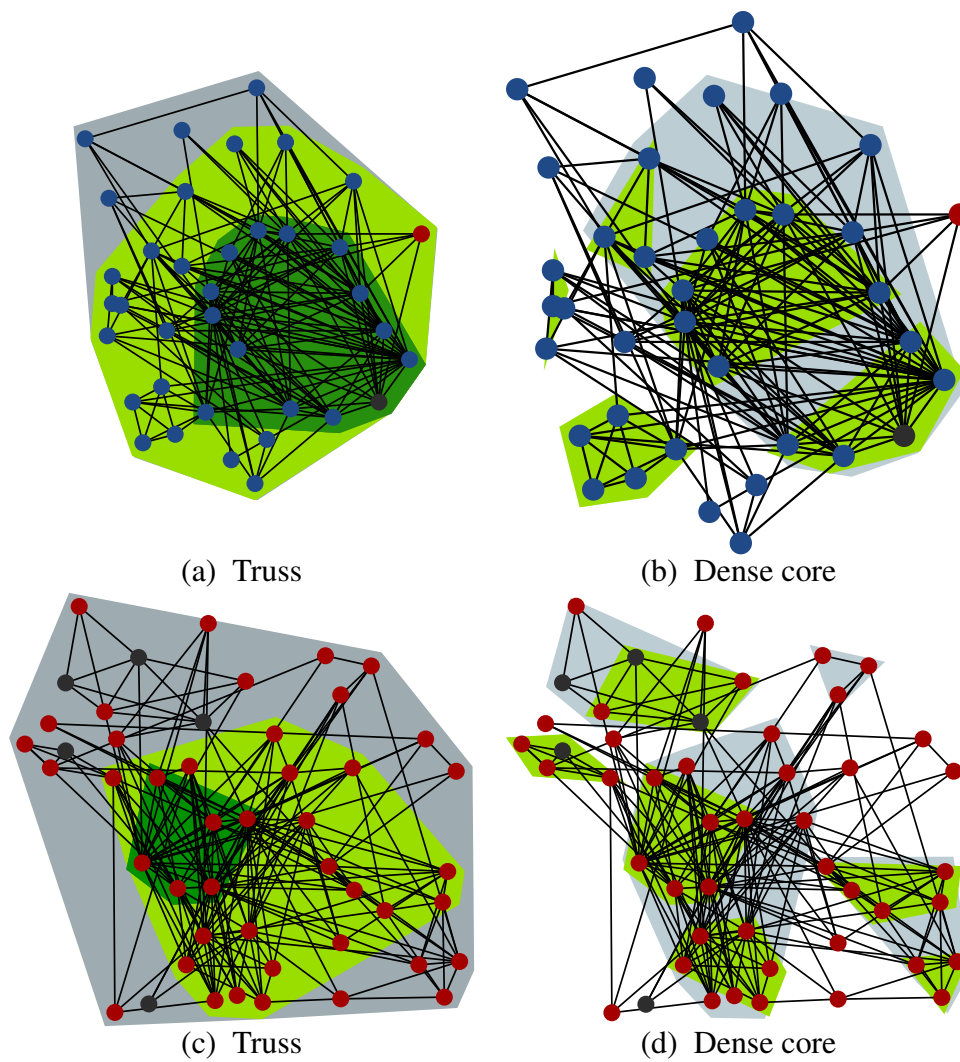


Fig. 3.9 Detailed structures of conservative, and liberal communities. Dense core found more finer cores compared to Truss.

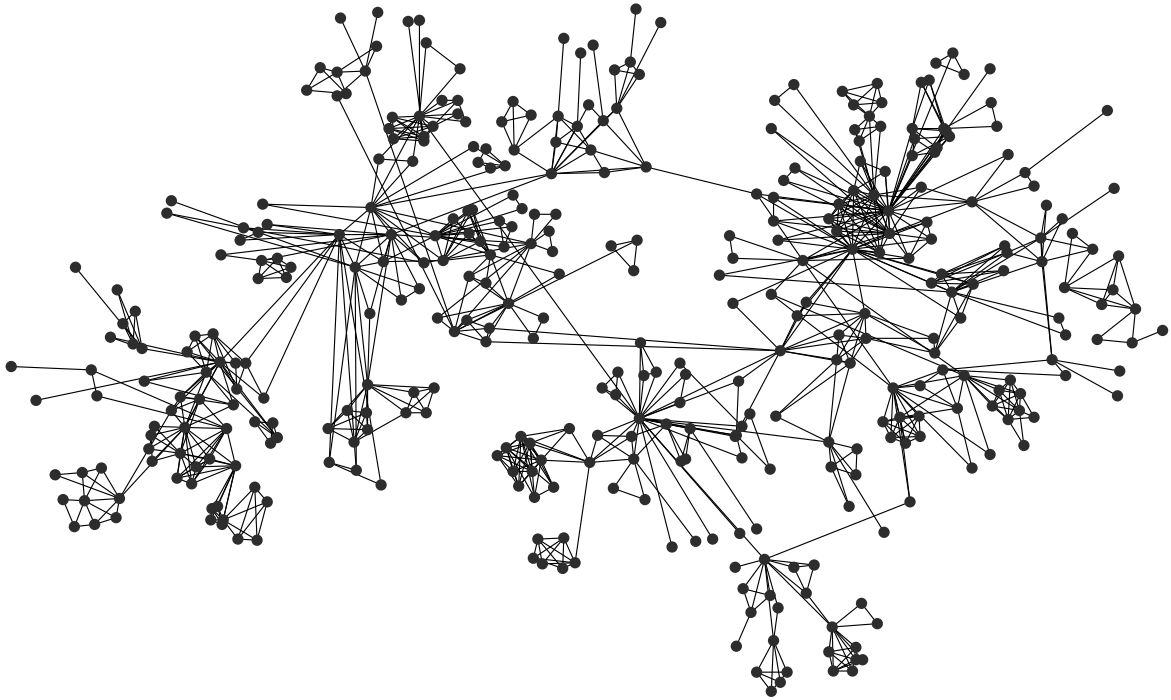


Fig. 3.10 Graph of collaboration among network scientists.

ρ -dense core with ρ of less than 0.3 could not be completed due to memory limitation. We also noted that the result of discrete model is omitted due to the limited space of the paper.

Fig. 3.11 shows subgraphs found by modularity and dense core. We observed that the modularity found larger communities, while truss and dense core found finer and similar communities. Fig. 3.12 shows the subgraphs around two communities (A) and (B). In the subgraph around (A), we noticed that there are many large cliques and clique-like subgraphs. Truss and dense core found these subgraph including one large clique-like subgraph (*). It is interesting to note that the large subgraph consists of leading scientists in biology such as Dr. Barabash, Dr. Jeong, and Dr. Oltvai. In the subgraph around (B), we noticed that, unlike (A), this subgraph has a star structure consisting of a hub node (red-colored node) corresponding a leading physicist, Dr. Newman. Truss and dense core failed to find communities with star structure, but modularity succeeded. These results imply that the truss and dense core is useful for capturing block structured cores, and modularity is advantageous in capturing star structured cores.

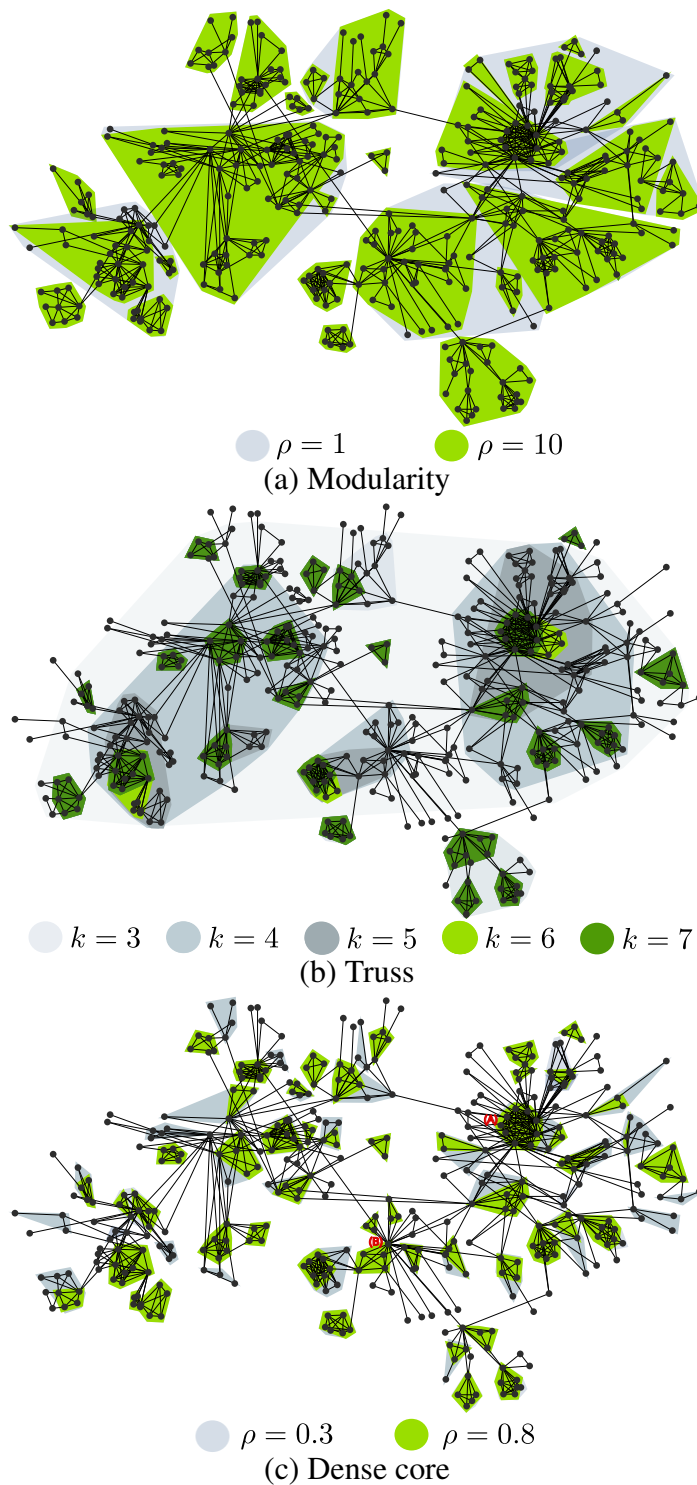


Fig. 3.11 Cohesive subgraphs found by (a) Modularity, Truss, Discrete model, and Dense core. Modularity found coarse communities, and Truss and Dense core found more finer communities.

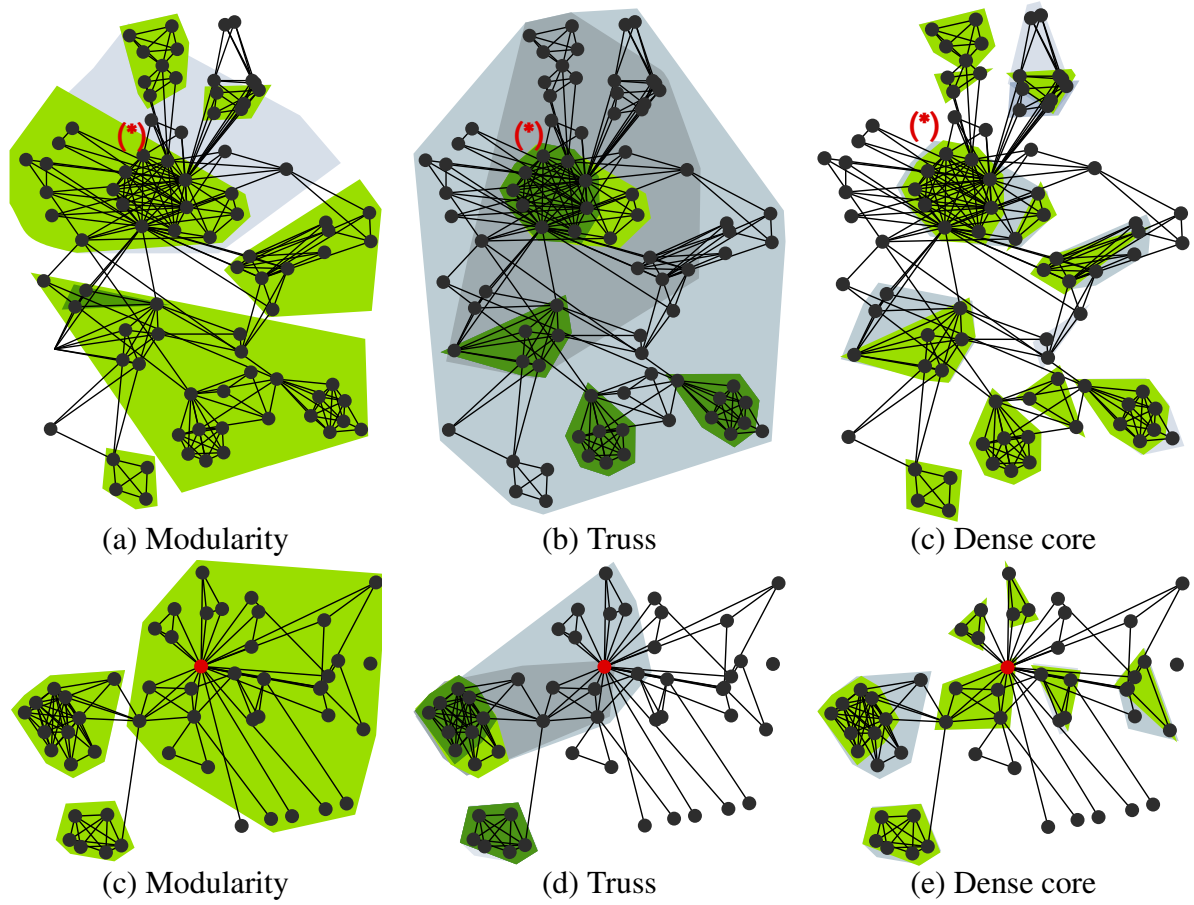


Fig. 3.12 Detailed structures around (A) in Fig. 3.8 (c). Truss and dense core found cliques in (b) and (c), whereas failed to find star structured subgraph in (d) and (e).

3.6 Discussion

The dense core model can find two core/periphery structures embedded in synthetic and karate club networks. However, for the networks of political books and scientists, we found that such a core-periphery dichotomy was hard. In particular, there were sub-peripheries lying between the core and the periphery. It is appreciated that some real-life communities cannot be well separated into core part and peripheral part. Rather they include sub-peripheries having dense connections with cores, and connect to other sub-peripheries to some extent. Indeed, sub-peripheries are found with truss. It is possible to extract more deep and detailed hierarchical relations with more than two partitions. However, there is a trade-off between computational costs and the detail of hierarchy. As many partitions we have, more detail hierarchy we have, but more time and memories we need.

We also noted that dense core model failed to capture star-structured community in collaboration network. This is due to the fact that the star structure has very small minimum cut, and therefore, is divided in a large value of ρ . On the other hand, the modularity could find star-structured community. Therefore, if we are interested in star-structured core, modularity would be more appropriate.

To find the best partitions over V , we adopted two step approach: enumerating all candidate subgraphs, and finding their best alignment. The time complexity for enumerating all candidate subgraphs (ρ -dense subgraphs) is $\mathcal{O}(n^L 2^{L-1})$, where n is the number of nodes and L is the maximum number of nodes of ρ -dense subgraphs. We also need to solve an integer programming problem to find the best alignment of these subgraphs, which is also known as an NP-hard problem. Therefore, the proposed algorithm is not scalable with respect to the number of nodes and edges. In our experiment, the algorithm could be carried out only for $\rho \in [0.3, 1]$ in the network of political books and the network of scientists. This is because the number of ρ -dense subgraph becomes too large with a smaller value of ρ . In other words, if a relatively large value of ρ is sufficient for analysis, or, equivalently, if only relatively small cohesive subgraphs exist, the algorithm scales. It might be possible to reduce the computation cost by screening candidate subgraphs to be examined. For example, $\rho(|U| - 1)$ -core is a necessary condition of ρ -dense core and its examination can be done very quickly. Another way is to employ a standard heuristic to optimize the criterion of ρ -dense core such as *recursive bisection* [57], *taboo search* [89]. Such a heuristic may be useful to reduce the computation cost with sub-optimal solutions.

3.7 Summary

In this chapter, we present dense core model for finding core/periphery structures. We have shown that dense core criterion produces well-separated subgraphs with a small value of ρ , and also produces cohesive subgraphs with block structures. By this unique nature, we are able to find communities as well as its cores with the dense core model. We show that the dense core model was advantageous in exploiting subgraphs with block structures, but not for finding star structures. In our experiments, we found that communities may not have clear core-periphery dichotomy in practice. Rather communities often contained sub-peripheries having dense connections with cores, and connects to other sub-peripheries to some extent. Therefore, further developments are needed in order to find them.

Chapter 4

Dense core model with sub-peripheries

In Chapter 3, we have concentrated on a simple core/periphery structure consisting of cores and peripheries. As shown in our experiment, real-life communities do not always have clear dichotomy between core and periphery. Rather they include sub-peripheries having dense connections with cores, and connects to other sub-peripheries to some extent. In this chapter, we consider more general core/periphery structures with sub-peripheries.

4.1 Model

We have considered a simple model consisting only of core and periphery, which are expressed as two partitions $\mathcal{P}_0, \mathcal{P}_1$ such that $\mathcal{P}_1 \preceq \mathcal{P}_0$. Hereinafter, we call this model as *two partition model*. In order to capture sub-peripheral nodes, additional partitions are needed between them. We, therefore, consider a variant consisting of a series of partitions $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_S$ such that

$$\max_{\mathcal{P}_s} \sum_{U \in \mathcal{P}_s} q(U; \rho_s), \quad (4.1)$$

where

$$\mathcal{P}_0 = \{V\}, \mathcal{P}_s \preceq \mathcal{P}_{s-1}, \forall s = 1, 2, \dots, S, \quad (4.2)$$

and $q(\cdot)$ is the quality function of ρ -dense core, and ρ_s is the resolution parameter of $\mathcal{P}_s$. We then define the *dendrogram* as the tree in which an element U in $\mathcal{P}_s$ has children $U_1, U_2, \dots, U_K$ in $\mathcal{P}_{s+1}$ such that $U = \cup_{i=1}^K U_i$. For example, Fig. 4.1 shows a graph (d) and its dendrograms (a)-(c). The height of the dendrogram is the resolution of the partitions. The dendrogram (a) is derived by two partition model, and the others (b) and (c) are derived by the model with three or 20 partitions. We observed that the three partition model (b) indicates that the tie between core and periphery is different in each community, which is not indicated

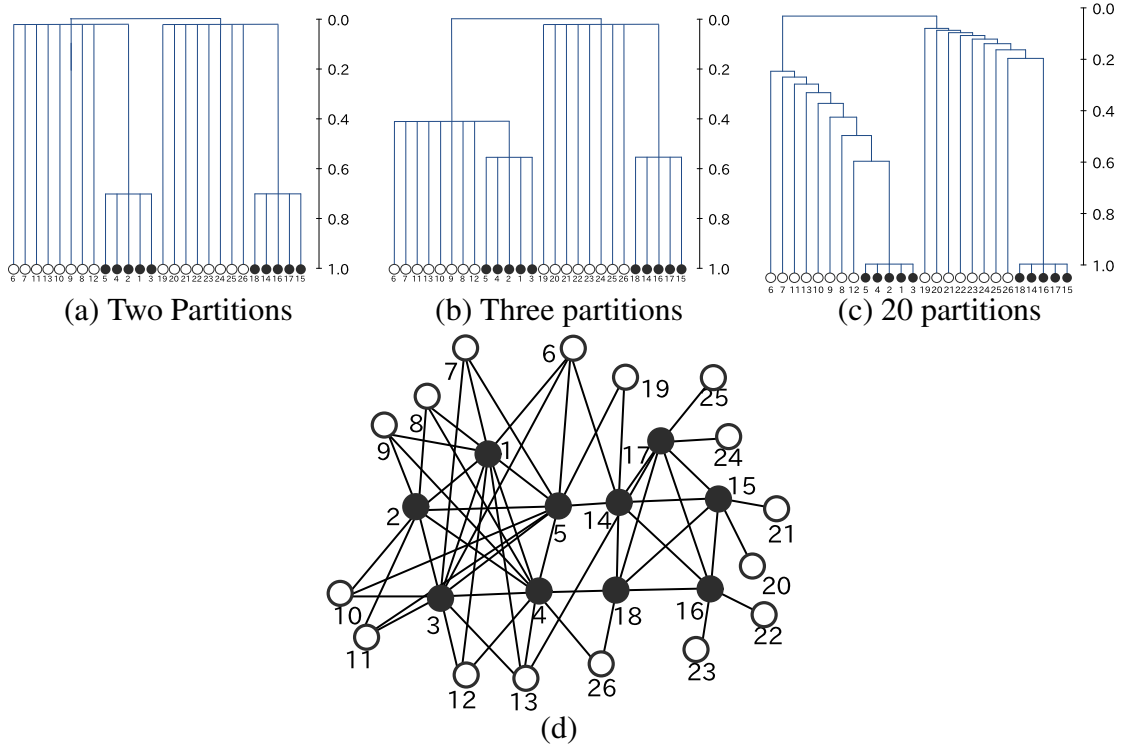


Fig. 4.1 Example of dendrogram of graph

from two partition model (a). By increasing the number of partitions, we can know more detail structures of the graph (c).

In this chapter, we only consider the most detailed model consisting of a series of refinement partitions $(\mathcal{P}_\rho)_{\rho=0}^1$ where $\mathcal{P}_\rho$ is the partition with resolution ρ . In order to find them, we need to address following two difficulties. First, we note that the series is findable by finding optimal partitions over all possible resolution, but this is not practical since ρ is a continuous parameter. However, by using the property of dense core we only need to solve (4.1) finite times to construct the dendrogram, which will be shown in the next section. Second, the construction of the dendrogram involves the extractions of many optimal partitions. The extraction is an NP-hard problem and turns out to be infeasible for small resolution parameters (see Section 3.6). In order to tackle this difficulty, we propose in Section 4.3 one exact and one heuristic algorithm.

4.2 Property

Consider a series of partitions $(\mathcal{P}_\rho)_{\rho=0}^1$ with resolution parameters $\tau_1, \tau_2, \dots, \tau_S$ ($\tau_1 < \tau_2 < \dots < \tau_S$) such that

$$\mathcal{P}_\rho = \begin{cases} \overline{\mathcal{P}}_0, \forall \rho \in [0, \tau_1) \\ \overline{\mathcal{P}}_s, \forall \rho \in [\tau_{s-1}, \tau_s) \quad , \\ \overline{\mathcal{P}}_S, \forall \rho \in [\tau_S, 1] \end{cases}$$

where $\overline{\mathcal{P}}_s$ is a partition over V . It should be noted that the length of the series $\overline{\mathcal{P}}_1, \overline{\mathcal{P}}_2, \dots, \overline{\mathcal{P}}_S$ is at most $n - 1$ since a graph is divided into singleton nodes with $n - 1$ partitions, and cannot be divided further. We call the resolution parameter τ_s at which the partition changes as a *transition point*.

The transition points are not clear in advance, but it is possible to find them with the property of dense core. We begin by showing the property of root and leaf partition $\mathcal{P}_0, \mathcal{P}_1$.

Property 3 (Optimal Partition at $\rho = 0$ and $\rho = 1$).

1. $\mathcal{P}_0 = \{V\}$.
2. $\mathcal{P}_1$ consists of cliques and singleton sets.

Proof.

1. Let us denote a subset of V by U . At $\rho = 0$, $q(U; 0) = W(U)$. Since $W(U) \geq W(Z) + W(U \setminus Z)$, $\forall Z \subseteq U$, $q(U; 0)$ is larger than the sum of qualities $q(Z; 0) + q(U \setminus Z; 0)$, that is $\mathcal{P}_0 = \{V\}$ achieves the maximum quality.
2. At $\rho = 1$, $q(U; 1) = W(U) - |U|(|U| - 1)/2$ is non-positive since $W(U) \leq |U|(|U| - 1)/2$ for any $U \subseteq V$. $q(U; 1)$ is maximum 0 if U is either a clique or a singleton node. Therefore, the optimal partition $\mathcal{P}_1$ consists of singleton sets and cliques.

□

Starting from $\{V\}$ ($\rho = 0$), a partition $\mathcal{P}_\rho$ changes to a finer partition $\mathcal{P}'$ at a transition point, and eventually all elements become cliques or singleton nodes ($\rho = 1$). The following theorem shows a way to derive the transition point.

Theorem 2 (Transition point of $\mathcal{P}_\rho$). *The transition point τ_s of $\mathcal{P}_s$ is given by*

$$\tau_s = \min_{U \in \mathcal{P}_s} \psi(U) \tag{4.3}$$

Proof. From (4.2), $\mathcal{P}_{s+1}$ can be derived by partitioning elements of $\mathcal{P}_s$. It follows from Lemma 1 that $\mathcal{P}_s$ is optimal if no dichotomies of any element $U \in \mathcal{P}_\rho$ increases the value of q , that is

$$\begin{aligned} q(U) &\geq q(Z) + q(U \setminus Z), \forall Z \subset U \in \mathcal{P}_s \\ &\iff \frac{B(Z, U \setminus Z)}{|Z||U \setminus Z|} \geq \rho, \forall Z \subset U \in \mathcal{P}_\rho \\ &\iff \min_{U \in \mathcal{P}_\rho} \min_{Z \subset U} \frac{B(Z, U \setminus Z)}{|Z||U \setminus Z|} \geq \rho. \end{aligned}$$

By definition of the transition point, the left hand side of the inequality is τ_s . $\square$

Lemma 1 (Optimality of U). *If no dichotomies of U increase the value of q , then no multiple partitions of U increases the value of q .*

Proof. The condition on which no dichotomies $\{Z, U \setminus Z\}$ ($Z \subset U$) of U increases the value of q is given by

$$\begin{aligned} q(U) &\geq q(Z) + q(U \setminus Z), \forall Z \subset U \\ &\iff B(Z, U \setminus Z) - \rho|Z||U \setminus Z| \geq 0, \forall Z \subset U. \end{aligned} \quad (4.4)$$

Let $\{Z_1, Z_2, \dots, Z_K\}$ be a partition of U with size K . We have

$$\begin{aligned} q(U) - \sum_{i=1}^K q(Z_i) &= \frac{1}{2} \sum_{i=1}^K \sum_{j=1, j \neq i}^K (B(Z_i, Z_j) - \rho|Z_i||Z_j|) \\ &= \frac{1}{2} \sum_{i=1}^K (B(Z_i, U \setminus Z_i) - \rho|Z_i||U \setminus Z_i|). \end{aligned}$$

It follows from (4.4) that $B(Z_i, U \setminus Z_i) - \rho|Z_i||U \setminus Z_i| \geq 0$, for all Z_i . Therefore, we have $q(U) \geq \sum_{i=1}^K q(Z_i)$ for any partitions. $\square$

This theorem directly induces the transition point of an optimal partition $\mathcal{P}_\rho$ by examining dichotomies of each element $U \in \mathcal{P}_\rho$, not all possible partitions.

Now, we will show that it is possible to construct $\mathcal{P}_{s+1}$ from $\mathcal{P}_s$ with a ratio cut partitioning.

Theorem 3 (Rerelation between $\mathcal{P}_s$ and $\mathcal{P}_{s+1}$).

$$\mathcal{P}_{s+1} = \mathcal{P}_s \setminus \{U\} \cup \{Z, U \setminus Z\},$$

Algorithm 2 DIVISIVE DENSE CORE PARTITIONING

Input: Graph $G = (V, E)$
Output: Refinement sequence $(\mathcal{P}_\rho^*)_{\rho=0}^1$

- 1: $\mathcal{P}_0 = \{V\}$
- 2: $(\mathcal{P})_{\rho=0}^1 = \text{BISECT}(V, 0)$

Subroutine BISECT

Input: Node set V , resolution parameter τ
Output: Refinement sequence $(\mathcal{P}_\rho)_{\rho=\tau}^1$ over V

- 3: **if** V includes only one node or is a clique **then**
- 4: $\mathcal{P}_\rho = \{V\}, \forall \tau \leq \rho \leq 1$
- 5: **Return**
- 6: **end if**
- 7: $Z = \text{RATIOCUT}(V)$
- 8: $\tau' = \frac{B(Z, V \setminus Z)}{|Z||V \setminus Z|}$
- 9: $\mathcal{P}_\rho = \{V\}, \text{ for all } \tau \leq \rho < \tau'$
- 10: $(\mathcal{P}')_{\rho=\tau'}^1 = \text{BISECT}(Z, \tau')$
- 11: $(\mathcal{P}'')_{\rho=\tau'}^1 = \text{BISECT}(V \setminus Z, \tau')$
- 12: Merge $(\mathcal{P}')_{\rho=\tau'}^1$ and $(\mathcal{P}'')_{\rho=\tau'}^1$ to $(\mathcal{P})_{\rho=\tau}^{\tau'}$

where U is the element giving the transition point τ_s in (4.3) and $\{Z, U \setminus Z\}$ is the corresponding dichotomy. That is $B(Z, U \setminus Z)/|Z||U \setminus Z| = \tau_s$.

Proof. It follows from (4.4) that $q(U) = q(Z) + q(U \setminus Z)$ holds at the transition point τ_s . That is, the dichotomy of U does not decrease the quality of partition. Therefore, the finer optimal partition $\mathcal{P}_{s+1}$ at τ_s can be constructed by replating $\{U\}$ with $\{Z, U \setminus Z\}$. $\square$

This theorem makes it possible to construct $\mathcal{P}_{s+1}$ from $\mathcal{P}_s$. It should be noted that the elements $Z, U \setminus Z$ dichotomized at resolution τ_s may be further dichotomized at τ_s , and so forth. In such a case, we denote the most finest partition at τ_s by $\mathcal{P}_{s+1}$.

4.3 Algorithm

On the basis of Theorem 2 and 3, the series of partitions $(\mathcal{P}_\rho)_{\rho=0}^1$ can be found with a recursive ratio cut bipartition. The pseudo code of this algorithm is shown in Algorithm 2. This algorithm involves a ratio cut algorithm requiring huge computational cost to find minimum cut. The simplest way to find the best dichotomy is to examine all possible 2^{n-1} partitions. Such an exhaustive search is, however, practically infeasible even for a modest number of nodes. Therefore, various efficient exact algorithms as well as heuristics have been developed to derive the solutions. Here we look at one exact algorithm and one heuristics.

4.3.1 Exact Algorithm

The ratio cut problem is a fractional programming problem [71], which is formulated as

$$\min_{\mathbf{x} \in S} \frac{f(\mathbf{x})}{g(\mathbf{x})},$$

where $f, g, h, j = 1, \dots, m$ are real-valued function defined on a set $S = \{x \in \mathbb{R} : h_j(\mathbf{x}) \leq 0, j = 1, \dots, m\}$ such that $g(\mathbf{x}) > 0$ on S . For a graph $G = (V, E)$, the ratio cut problem is expressed as

$$\begin{aligned} S &= \{x_i \in \{0, 1\}\} \\ f(\mathbf{x}) &= \mathbf{x}'\mathbf{L}\mathbf{x} \\ g(\mathbf{x}) &= \mathbf{x}'(n\mathbf{I} - \mathbf{1}\mathbf{1}')\mathbf{x}, \end{aligned}$$

where $\mathbf{L}$ is the laplacian matrix of the graph G defined by

$$\mathbf{L} = (l_{ij}), l_{ij} = \begin{cases} B(i, V \setminus i) & (i = j) \\ -1 & (\{i, j\} \in E) \\ 0 & (\{i, j\} \notin E) \end{cases}, \quad (4.5)$$

and $\mathbf{x}$ is the indicator vector expressing $x_i = 1$ if $i \in Z$, otherwise $i \in V \setminus Z$. By using efficient solvers, an optimal dichotomy may be found within a reasonable time for a large graph. We use a simple algorithm, *Dinkelbach's Algorithm* [20], to solve the fractional programming problem. The computational complexity is, in the worst case, $\mathcal{O}(2^n)$, but practically less than this if the graph is sparse. The pseudo code is shown in Algorithm 3. For more details, see Appendix A.

4.3.2 Heuristic Algorithm

Solutions to the ratio cut problem are generally derived using heuristics such as *Kernighan-Lin algorithm* [40], *Fiduccia-Mattheyses algorithm* [24] and *Spectral partitioning* [82]. *Kernighan-in algorithm* and *Fiduccia-Mattheyses algorithms* are well-known local algorithm for graph partitioning problems. Starting from an initial partitioning of the node set, they iteratively swap a pair nodes to improve the quality of the partitions. *Spectral Partitioning* is a global algorithm that focuses on the properties of the entire graph. It derives a sub-optimal solution from the spectrum of the laplacian matrix of the graph. Typically, the local algorithms run faster than the global algorithms, but produce poorer solutions. In order to maintain the quality of the partitions, we adopt Spectral partitioning as a heuristic for the

Algorithm 3 EXACT RATIOCUT ALGORITHM (Dinklebach's Algorithm)

- Input:** Graph $G = (V, E)$
Output: Subset U of nodes
- 1: Divide V randomly into U and $V \setminus U$
 - 2: $\delta_0 = B(U, V \setminus U) / |U||V \setminus U|$
 - 3: $k = 0$
 - 4: **repeat**
 - 5: $k = k + 1$
 - 6: Solve Linear Programming Problem

$$\min_{\mathbf{x} \in \{0,1\}^n} \mathbf{x}' \mathbf{L} \mathbf{x} - \delta_k \mathbf{x}' (n\mathbf{I} - \mathbf{1}\mathbf{1}') \mathbf{x},$$

where $\mathbf{x}$ is the indicator vector of U such that

$$\mathbf{x} = (x_i), x_i = \begin{cases} 1 & (i \in U) \\ 0 & (\text{otherwise}). \end{cases}$$

- 7: Construct U from $\mathbf{x}$
- 8: $\delta_k = B(U, V \setminus U) / |U||V \setminus U|$
- 9: **until** $\delta_k < \delta_{k-1}$

ratio cut problem. We note that the computational complexity of the spectral partitioning is $\mathcal{O}(|E| + |V|)$, therefore it is still scalable with respect to the size of graph. The pseudo code is shown in Algorithm 4. For more details, see Appendix B.

The proposed algorithm calls the ratio cut algorithm at most $|V| - 1$ times. Therefore, the complexity of the heuristic algorithm is $\mathcal{O}(|V|^2 + |V||E|)$ and $\mathcal{O}(|V|^2)$ for a sparse graph ($|E| \ll |V|^2$). In our simulation, the exact algorithm terminates within 30 minutes for a

Algorithm 4 HEURISTIC RATIOCUT ALGORITHM (Spectral Partitioning)

- Input:** Graph $G = (V, E)$
Output: Subset U of nodes
- 1: Construct laplacian matrix $\mathbf{L}$
 - 2: Compute the eigenvector $\mathbf{x}$ of the second smallest eigenvalue
 - 3: **for** $i = 1, 2, 3, \dots, n$ **do**
 - 4: Divide V into U and $V \setminus U$ such that $j \in U$ if $x_j \leq x_i$, otherwise $j \in V \setminus U$
 - 5: Compute $\delta_i = B(U, V \setminus U) / |U||V \setminus U|$
 - 6: **end for**
 - 7: Choose the dichotomy achieving the minimum δ_i .

graph with less than 1,000 nodes, and the heuristic within 10 minutes for a graph with less than 50,000 nodes.

4.4 Experiment

In this section, we analyzed the graphs in the previous section. For the graphs except the graph of scientific collaborations, the exact ratio cut bipartition algorithm was adopted. The heuristic bipartition algorithm based on spectral clustering was applied for the graph of scientific collaboration networks because the exact bipartition algorithm did not terminate within a week.

We begin by analyzing the synthetic graph shown in Fig. 4.2 (a). The dendrograms constructed with modularity, truss and dense core are shown in Fig. 4.2 (b)-(d) where the vertical axis represents the resolution parameter, and the circles at the bottom represent the nodes in Fig. 4.2 (a). We observed that modularity divided the core nodes into different partitions, while truss and dense core did not. Truss divided the peripheries first, while dense core divided the graph into two communities first. From the dendrogram constructed with the dense core model, we noticed that there are two different kinds of peripheries. The peripheries at the left of the dendrogram connect densely with the core, but the other peripheries at right connect more weakly with the core at the right. It should be noted that such a different kind of core/periphery structures are hardly found with two partition model without through examinations.

Next, we show our analysis on real-life graphs. The dendrograms of nodes in the karate-club network are shown in Fig. 4.3. We observed that modularity and dense core divided two leaders #1 and #34 into two communities at first, while truss did not. We noticed that the clique { #14, #8, #4, #3, #2, #1 } lacking one edge received the highest index in truss, but received the second highest in dense core. By comparing the results of the previous proposed model shown in Fig. 3.6, we noticed that more detailed structures are found in (d). For example, it seems that the nodes #12, #10, #13, #20, #22, #18 are the peripheries of the core nodes #1, #2, #3, #4, #8, and the node #14 is the sub-periphery.

Fig. 4.4 shows the dendrogram of the nodes of political book networks. The color of nodes at the bottom of the dendrograms represents conservative (red), centrist (black) and liberal (blue). We observed that all models divided the graph into large communities of conservatives and liberals. Then modularity divided each community into small communities of one or two nodes. Truss and dense core divided the communities into singleton nodes and large communities about 6 nodes. We also observed that dense core found two core/periphery

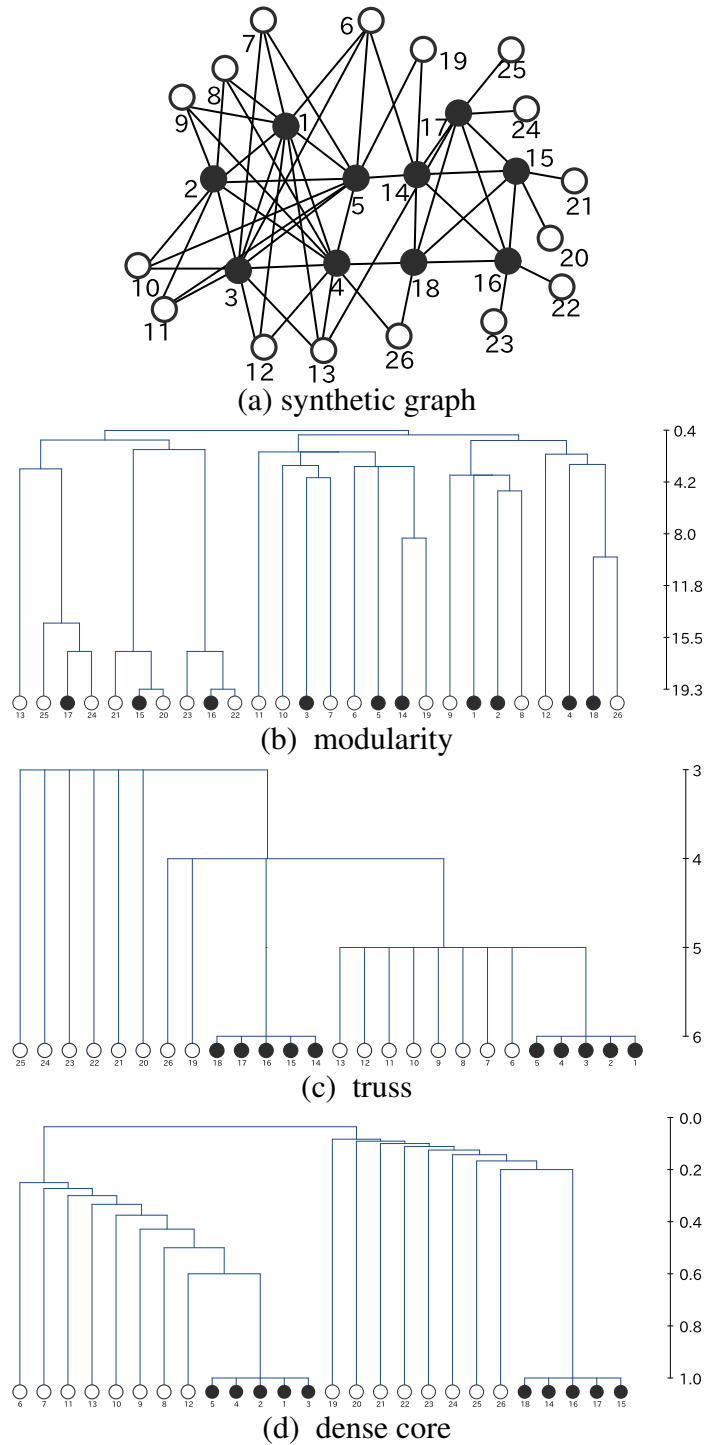
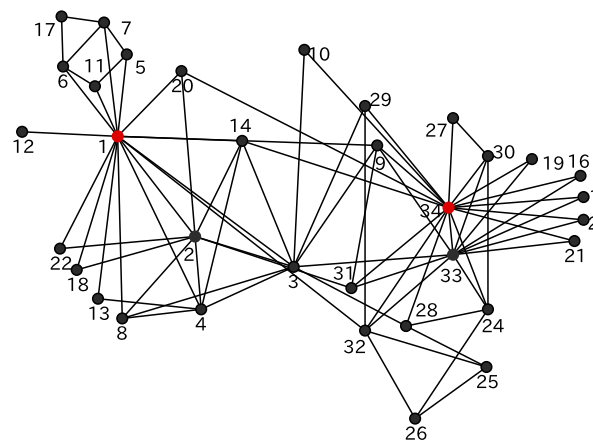
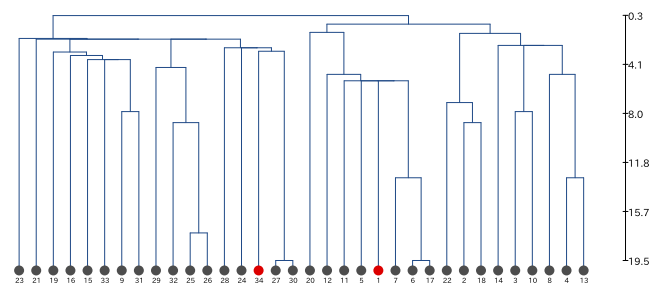


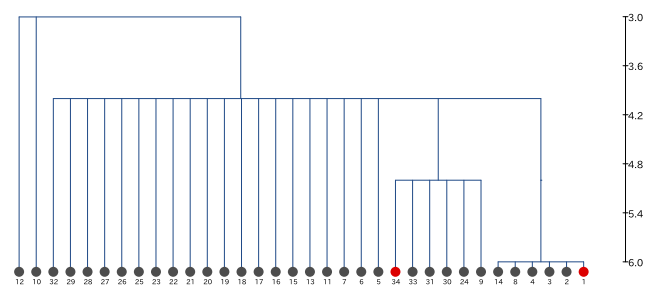
Fig. 4.2 Dendrograms of nodes in synthetic graph.



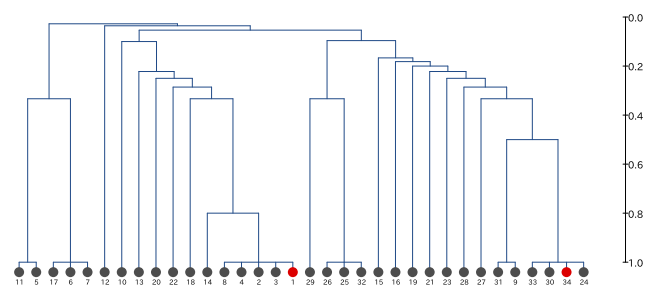
(a) Karate-club network



(b) modularity



(c) truss



(d) dense core

Fig. 4.3 Dendrograms of nodes in karate-club network.

structures surrounded by the rectangles (A) and (B), which correspond to the communities of conservatives and liberals, respectively.

Lastly, we show our analysis on the network of scientific collaboration shown in Fig. 4.5. We note that the exact algorithm did not terminate within one week, so that we applied heuristic algorithm, which terminated within one minute. Fig. 4.5 shows the dendrograms of the nodes. Note that the labels of nodes are omitted due to the lack of space. We noticed that the size of communities were one or two nodes in modularity even if there were large cliques. On the other hand, the dense core and truss kept the cliques. This is the one important nature of these two models, that is (almost) perfect subgraph tends not to be divided. We also noticed that dense core divided the graph into several large communities at first, while truss extracted core communities one by one. This is another important nature of dense core. Although it may be too intensive, we observed 12 core/periphery structures consisting of nodes surrounded by rectangles in Fig. 4.5 (c). They have different number of core and peripheral nodes as well as the strength of core-periphery ties. These diverse core/periphery structures are hardly found with the conventional models without examining many parameter configurations.

4.5 Discussion

Dense core model and modularity divided the graph into large communities in the network of karate club and political books. On the other hand, truss excludes peripheral nodes one by one. The difference is crucial for finding core/periphery structures, not just cores. We also noticed that the dense core model captured sub-peripheries in the network of karate-club, political book, and collaboration among scientists. Even in the same network, we found that there are diverse core/periphery structures. For example, in the network collaborations among scientists, there is one large core/periphery community of biologists, and many small core/periphery communities of web scientists, physicians, and computer scientists. The tie-strength of these communities are different, which is hardly found with two partition model.

The computational time is one drawback of the multiple partition model. For the network of karate-club and political blogs, the exact algorithm terminated within 3 minutes. However, it did not within a week for collaboration networks. This is due to the fact that the number of nodes increases, the number of possible dichotomies increases exponentially. Heuristic algorithm processed the graphs much faster than the exact algorithm. It processed the collaboration network within 10 seconds. Further speedup may be possible if we adopt faster ratio cut algorithms.

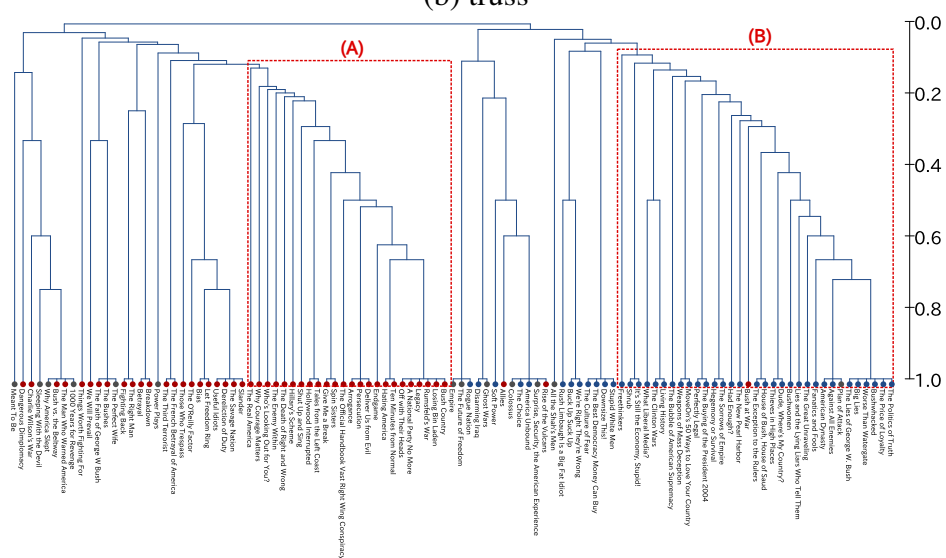
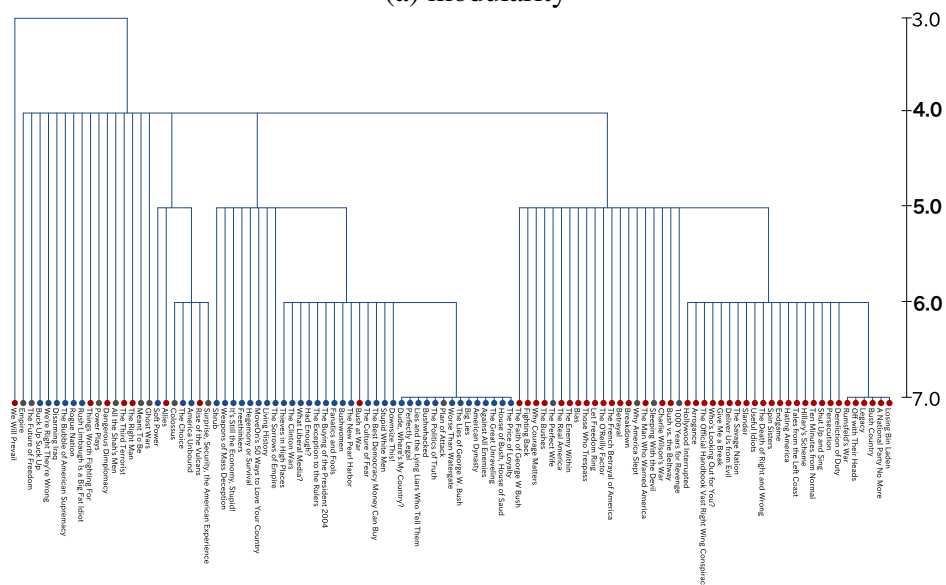
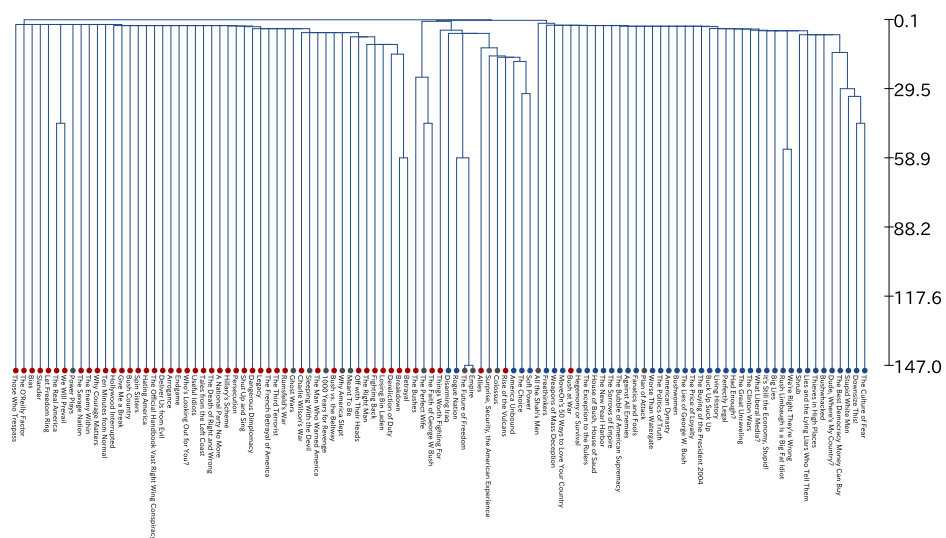
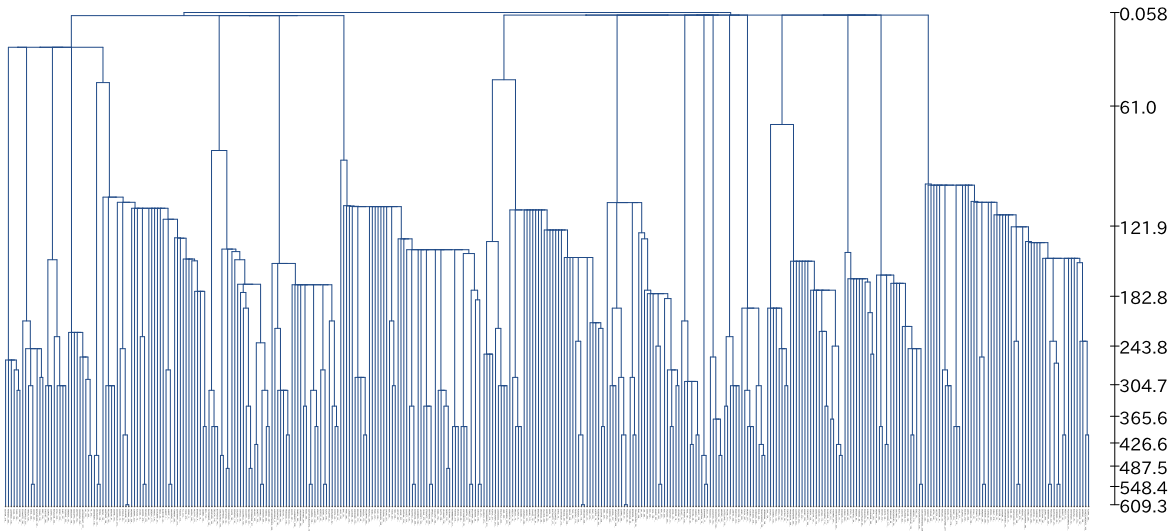
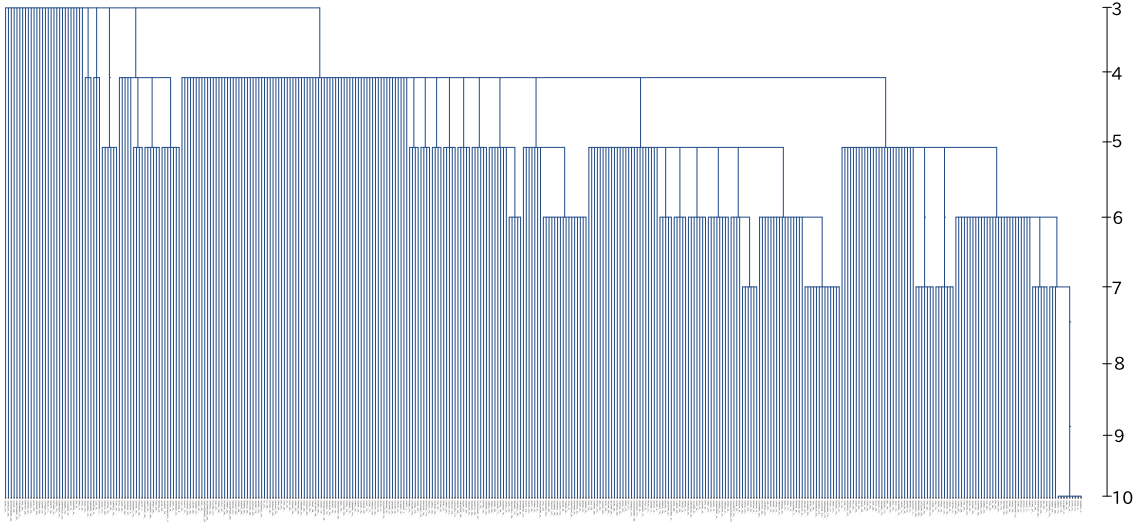


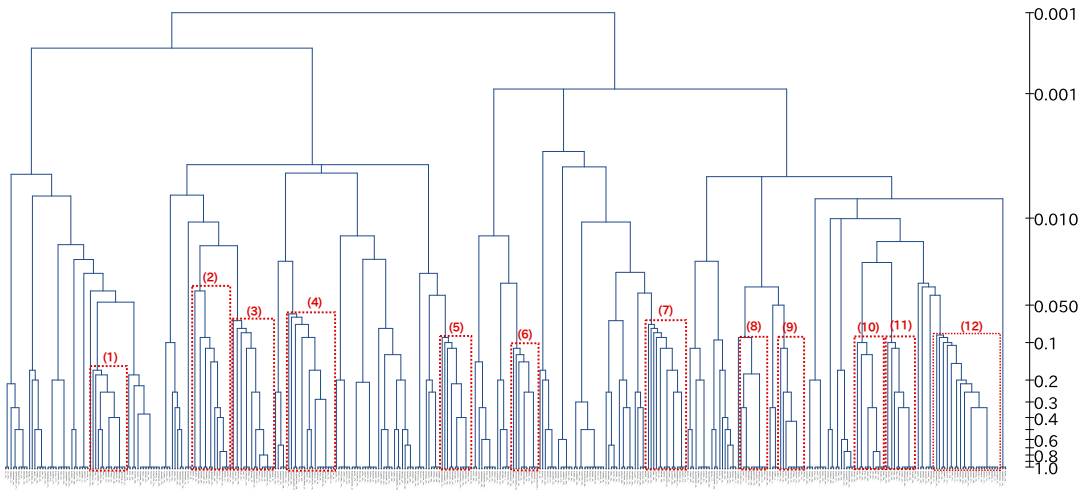
Fig. 4.4 Dendrograms of nodes in political book network.



(a) modularity



(b) truss



(c) dense core

Fig. 4.5 Dendrograms of nodes in collaboration networks.

4.6 Summary

In this chapter, an extended variant of dense core model consisting of more than two partitions is presented to capture sub-peripheral nodes. We have shown that the partitions can be found by dichotomizing the graph until all elements are either clique or singleton nodes. With this model, we are able to find diverse core/periphery structures having strong/weak ties between core and peripheries, that among cores, and that among peripheries, which are hardly found with two partition model. It should also be noted that the dendrogram is also very useful to determine the resolution parameters of two partition model. By using empirical model selection methods [62], we may find the appropriate resolutions parameters of two partition model. Unlike the other traditional models, the multiple partition model divided the graph into large communities first, and then divided each communities into cores, sub-peripheries and peripheries. The order of the divisions is of great importance if we are interested in core/periphery structures, not just cores.

One crucial limitation is that it contains too rich information to understand. As we have seen, it requires a knock to find core/periphery structures. In addition, the dendrogram turns out to be more difficult to understand as the graph gets larger. Therefore, further developments are needed to depict the result in order to interpret the result easily.

Chapter 5

Application to Graph Visualization

As we have seen, the series of partitions $(\mathcal{P}_\rho)_{\rho=0}^1$ have very rich structural information. This contains the community structure and the hierarchy of core and periphery of nodes. In many applications, experimenters appreciate if they could understand the overview of the results easily with a diagram. We, therefore, depicted the information with sociogram and dendrogram. These diagrams are very useful for understanding the structures of networks if the network is small. However, it turns out to be difficult to draw as well as to understand for large graphs. Here we introduce another graph diagram that illustrates both community structures and core/periphery relationships.

5.1 Layout

Given a series of partition $(\mathcal{P}_\rho)_{\rho=0}^1$ of a graph $G = (V, E)$, we consider the positions of nodes $(x_i, y_i), i = 1, 2, \dots, |V|$ on canvas so as to satisfy our two criteria;

1. Two nodes in the same community should be as close as possible.
2. Cores and peripheries should be as far as possible.

Let w, h be the width and height of the canvas and we assume that the origin $(0, 0)$ of the canvas is at the left upper corner and the end of the canvas (w, h) is at the right bottom corner. Our layout is based on so-called *circular layout* in which nodes are put along with a circle. The coordinate of node i is defined as

$$x_i = x_0 + R_i \cos(\theta_i), y_i = y_0 - R_i \sin(\theta_i),$$

where (x_0, y_0) is the center of the circle, R_i is the radius of node i and θ_i is the angle of node i . Depending on how to determine the angles and radius, there are several kinds

of circular layouts. One way is that put nodes such that the number of crossing edges is minimized in order to maintain the clarity of the diagram. Another way is to keep two nodes in the same community being as close as possible. Here we adopt the later approach and determine the angles $\theta_i, i = 1, 2, \dots, |V|$ as follow. Let $\lambda_1, \lambda_2, \dots, \lambda_{|V|}$ be an order of nodes, and $c_1, c_2, \dots, c_{|V|}$ ($c_i \in [0, 1]$) be a score of the coreness of nodes. The nodes are ordered according to the dendrogram. In particular, the node at the right most in the dendrogram is ordered at first, and the second right most node follows, and so forth. Then, we define the angle θ_i by

$$\theta_i = \frac{2\pi\lambda_i}{|V| + 1}. \quad (5.1)$$

It is easily conform that two nodes in the same elements over all partitions are very close. We also define the radius of node i by

$$R_i = R_{\min} + c_i(R_{\max} - R_{\min}), \quad (5.2)$$

where $R_{\min}$ and $R_{\max}$ are the minimum and maximum radius of the circle, respectively. It should be noted that the nodes having higher coreness positioned closer to center.

We consider that a node with higher quality of dense core should receive higher coreness values. Following this idea, we define the coreness of node i at resolution ρ by

$$\Delta q(i; \rho) = \frac{1}{2} (q(U_{\sigma(i, \rho)}; \rho) - q(U_{\sigma(i, \rho)} \setminus \{i\}; \rho)), \quad (5.3)$$

$$= \frac{1}{2} (B(i, U_{\sigma(i, \rho)} \setminus \{i\}) - \rho(|U_{\sigma(i, \rho)}| - 1)), \quad (5.4)$$

where $\sigma(i, \rho)$ is the index such that $U_{\sigma(i, \rho)}$ is the element of $\mathcal{P}_\rho$ including node i , that is $i \in U_{\sigma(i, \rho)} \in \mathcal{P}_\rho$. It should be noted that, for an optimal element $U \in \mathcal{P}_\rho$, we have

$$\sum_{i \in U} \Delta q(i; \rho) = q(U; \rho). \quad (5.5)$$

Therefore, $\Delta q(i; \rho)$ can be seen as a contribution of node i to the quality of dense core. Summing over all parameter values, we define a coreness c_i of node i by

$$c_i = \int_0^1 \Delta q(i; \rho) d\rho. \quad (5.6)$$

Note that c_i is non-negative since $\Delta q(i; \rho)$ only takes non-negative values. The upper bound of c_i depends on the topology of graphs, so that it is convenient to normalize the coreness

values by its maximum

$$\bar{c}_i = \frac{c_i}{\max_j c_j}. \quad (5.7)$$

Hereinafter, we call $\bar{c}_i$ as the coreness of node i .

Fig. 5.1 shows how we illustrate our diagram. We begin by constructing a dendrogram (b) from a given graph (a), and then embedding the dendrogram into a circle (c), and lastly change the radius of nodes according to their coreness (d).

Our diagram illustrates both community structure and the hierarchy of core and periphery, which are necessary informations to understand multiple core/periphery structures. It should be noted that the coreness (or centrality) and dendrogram are needed to draw this diagram, which are not available in traditional community models including modularity, truss, discrete core/periphery model.

5.2 Algorithm

The calculation of the coreness c_i defined in (5.6) involves integrations, which seem to be hard to compute. Fortunately, (5.6) can be rewritten in the linear form:

$$c_i = \sum_{s=0}^{S-1} \int_{\tau_s}^{\tau_{s+1}} \Delta q(i; \rho) d\rho = \sum_{s=0}^{S-1} \Delta Q(i, U_{\sigma(i, \tau_s)}; \tau_{s+1} - \tau_s),$$

where

$$\Delta Q(i, U; \tau) = \tau B(i, U \setminus \{i\}) - \frac{1}{2} \tau^2 (|U| - 1). \quad (5.8)$$

Here we have made use of the fact that the element $U_{\sigma(i, \tau)}$ is the same between transition points. From (5.8), the coreness of nodes can be computed as follow. At each transition point τ_s , we compute $\Delta Q(i, U; \tau_{s+1} - \tau_s)$ for all nodes $i \in V$. Then, go to the next transition point τ_s , and so forth. By summing overall $\Delta Q(i, U, \tau)$ for all $\tau = \tau_0, \dots, \tau_S$, we can compute the coreness of nodes. The pseudo code of the algorithm is shown in Algorithm 5. This algorithm involves the computation of $\Delta Q(i, U; \tau)$ for all nodes at each transition point, each of which takes time in order $\mathcal{O}(|V|^2)$ for a graph $G = (V, E)$. There is at most $|V| - 1$ transition points for the graph G . Therefore, the complexity of this algorithm is $\mathcal{O}(|V|^3)$, which is equal to the betweenness centrality, and larger than the eigenvector centrality taking $\mathcal{O}(|V|^2)$. In practice, the number of transition points are much smaller than $|V|$ and thus takes $\mathcal{O}(|V|^2)$.

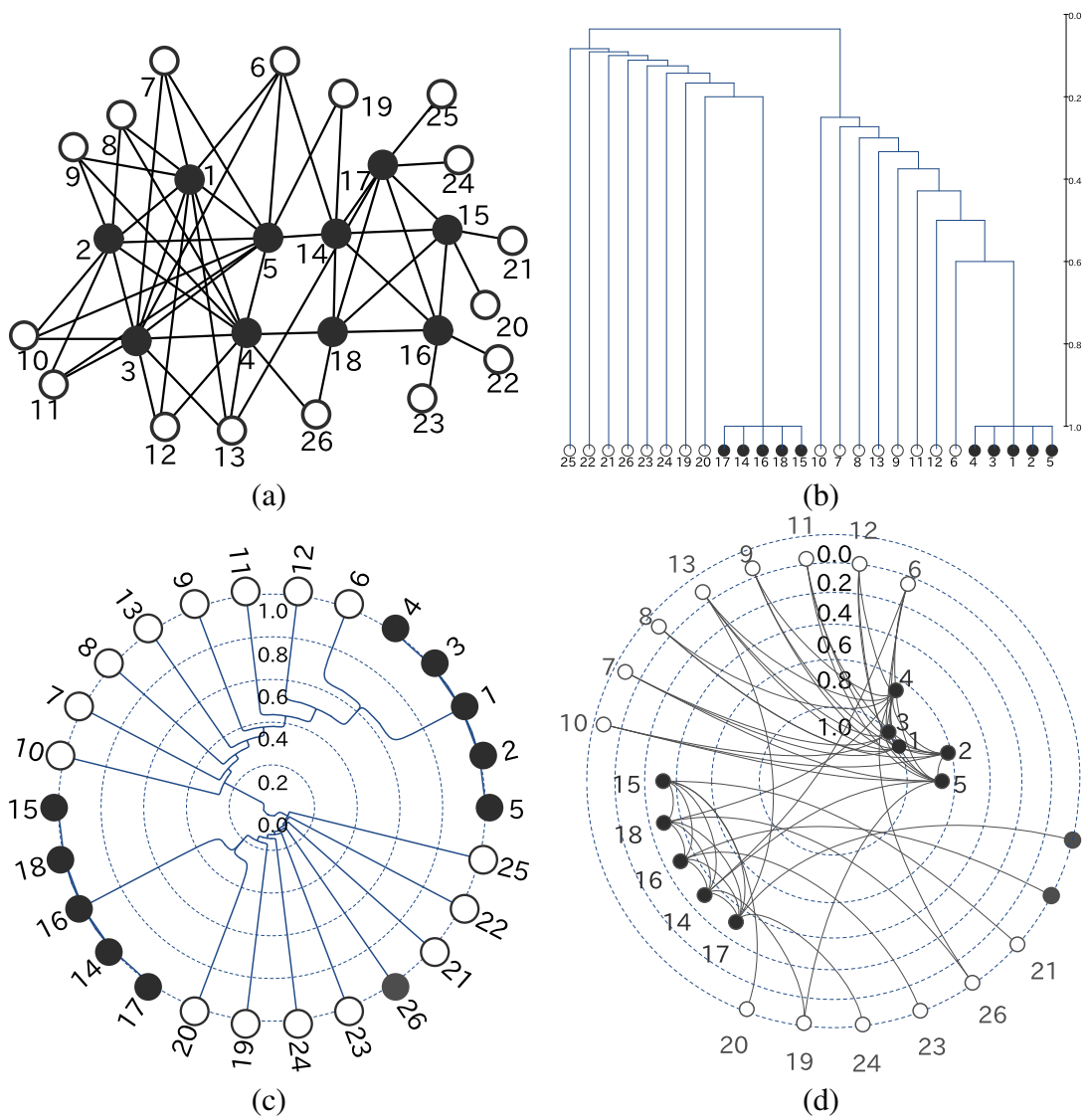


Fig. 5.1 Illustration of how to draw proposed graph diagram.

Algorithm 5 DENSE CORE CORENESS

Input: Graph $G = (V, E)$, Refinement sequence $(\mathcal{P}_\rho)_{\rho=0}^1$
Output: Coreness $\mathbf{c} = [c_1, c_2, \dots, c_{|V|}]$

- 1: $c_i = 0$ for all $i \in V$
- 2: **for** $s=0, 1, \dots, S-1$ **do**
- 3: **for** $i \in V$ **do**
- 4: Find the element $U \in \mathcal{P}_{\tau_s}$ of $\mathcal{P}_\tau$ including node i
- 5: $c_i = c_i + \Delta Q(i, U; \tau_{s+1} - \tau_s)$
- 6: **end for**
- 7: **end for**
- 8: Normalize the coreness by $c_i = c_i / \max_j c_j$ for all i .

5.3 Related work

One problem is that it cannot capture sub-peripheral nodes. As we have seen, real-life community cannot be simply dichotomized into core and periphery, rather there are sub-peripheries laying between them. To capture them, Borgatti and Everett introduced so-called *continuous model* that assigns a real value to a node, instead of assigning binary value expressing core node or not [8]. One seminal model is *eigenvector centrality*, which is the principle eigenvector of the adjacency matrix of a graph. It assigns high score for the nodes forming a dense subgraph, and is consistent with the ideal core/periphery model.

Followed by the study, several coreness measures have been proposed. They are classified into two categories depending on how a core is characterized: density-based and connectivity-based. In density model, a core is seen as a subset of nodes forming a dense subgraph. Several variants have been proposed to capture core in bipartite network [9], and to control core size and the sharpness of core periphery dichotomy [64]. In connectivity-based model, a core is seen as a subset of nodes positioned at the center of the network. Da Silva et al. proposed a connectivity-based coreness measures [18] for the analysis of metabolic networks, which takes the shortest paths between nodes into account. For social networks, Rossa et al. proposed another coreness measures [66], which takes the paths of random walk into account. In addition to these measures, various traditional centrality measures such as degree centrality, betweenness measure, eccentricity [26, 30, 68] can be also used to find the center of the network [44]. We note that these two measures can capture different kinds of cores. The density-based measures capture block-structured core, and the connectivity-based measures capture star-structured core. Therefore, they can be seen as two complementary measures.

One remaining issue is that these measures focus only on finding single (or global) core in a whole network. However, secondary cores exist even in a small network. Indeed, as

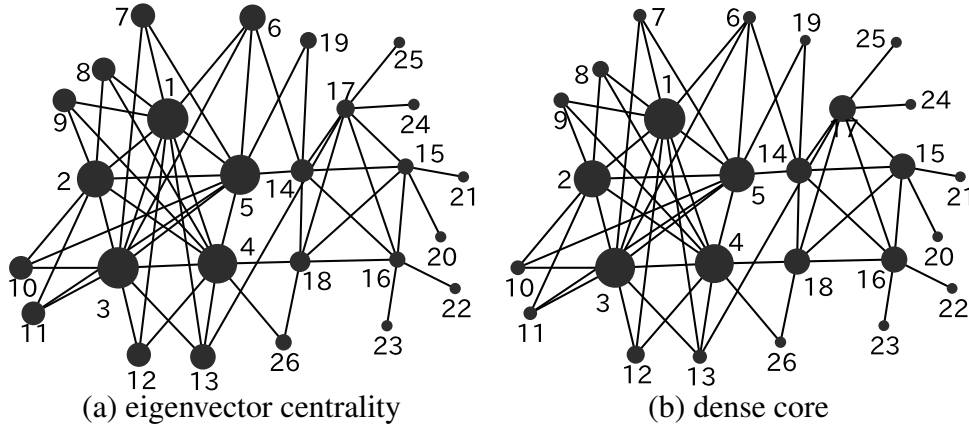


Fig. 5.2 Centralities of nodes in synthetic graph scored by eigenvector centrality and dense core model

suggested by recent study [64], multiple (or local) cores are observed in different parameter configurations. An example is shown in Fig. 5.2 where the radius of a node represents the degree of coreness. We observed that the eigenvector centrality [8] assigned high coreness values for the core nodes at the left, while low values for those at the right. This is because the eigenvector centrality was designed to capture single core in a graph, not multiple cores. Dense core, on the other hands, assigned higher scores for both right and left core nodes, and lower scores for the peripheries. This result shows the usefulness of dense core model in exploring multiple cores. It is possible to find multiple cores by aggregating the results of different parameter configurations, but it requires a skillful know-how.

5.4 Experiment

In this section, we depicted several graphs with our proposed circular layout. We depict the three small graphs we have seen so far. We also depicted one large graph whose dendrogram is hard to understand due to the scale of the network.

Fig. 5.3 illustrates the graph of karate club. We observed that two individuals #1 and #34, which are the leaders of the club, are at the center of the network, while others are far from the center. We notice that there are two core/periphery structures embedded in the graph. One core consists of the node #1, #2, #3, #4, other core consists of #33 and #34. We also notice that there are three two kinds of peripheries; one periphery connects only to the core such as #18, #22, and #21, the other connect both the cores such as #29, #10, #20 and #28. The later nodes bind the two cores together so that they would be important to resolve the conflicts between two cores.

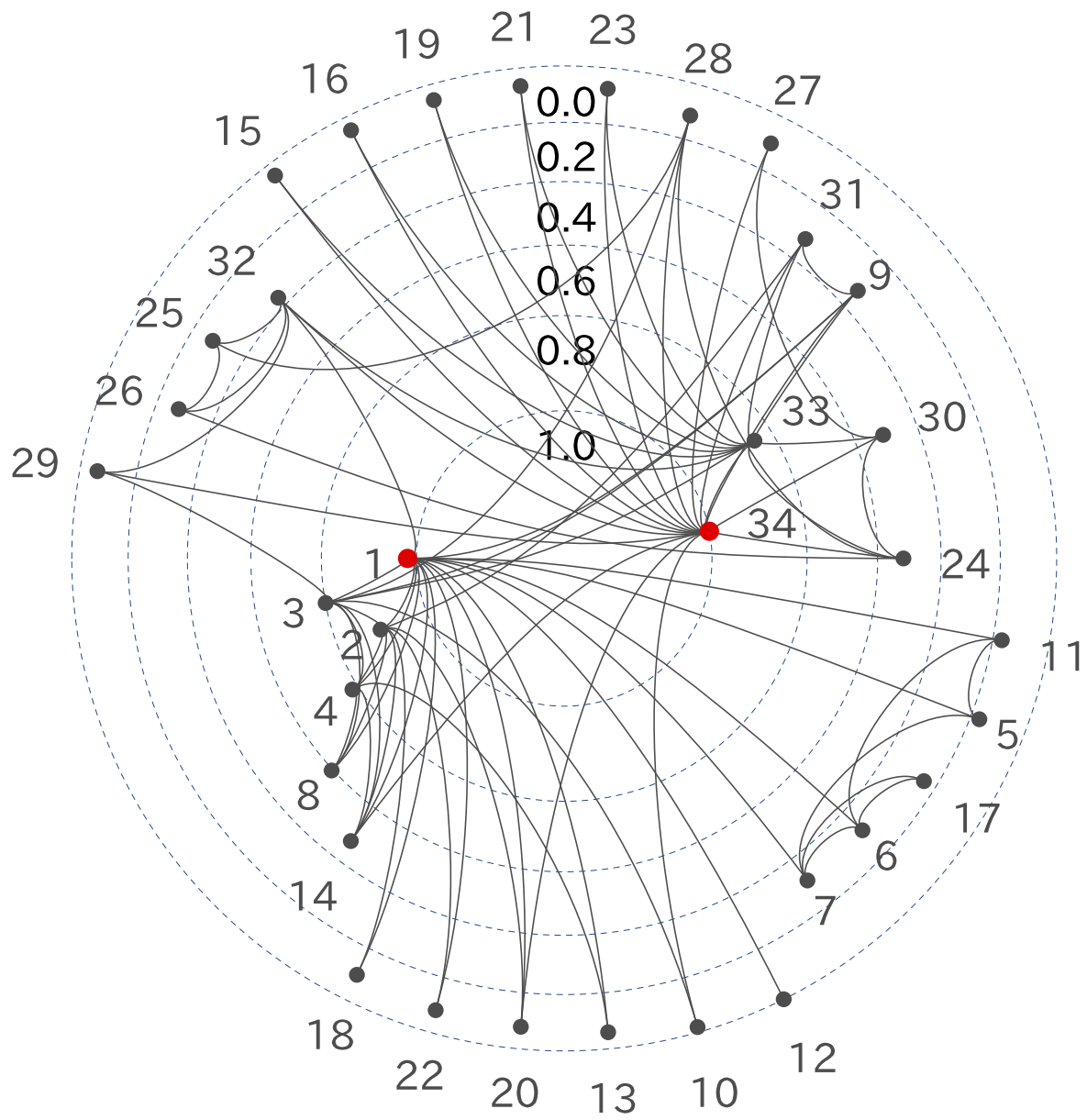


Fig. 5.3 Illustration of karate club network

Table 5.1 Ranking of scientists according to coreness measure

Rank	Eigenvector centrality	(Field)	Dense core	(Field)
1	BARABASI	(Biology)	BARABASI	(Biology)
2	JEONG	(Biology)	OLTVAI	(Biology)
3	OLTVAI	(Biology)	JEONG	(Biology)
4	VICSEK	(Biology)	RAGHAVAN	(World Wide Web)
5	RAVASZ	(Biology)	RAJAGOPALAN	(World Wide Web)
6	DERENYI	(Biology)	TOMKINS	(World Wide Web)
7	NEDA	(Biology)	RAVASZ	(Biology)
8	FARKAS	(Biology)	VICSEK	(Biology)
9	SCHUBERT	(Biology)	ROSENBLUM	(World Wide Web)
10	TOMBOR	(Biology)	KUMAR	(World Wide Web)

Fig. 5.4 illustrates the graph of political books where the color of nodes represent the political stands (blue is liberal and red is conservative). We observed that the network was roughly divided into the communities of liberals (blue) and conservative communities (red). These communities have one cores and many peripheries (including sub-peripheries). We also noticed that the liberal and conservative communities do not directly connect. This implies the tendency of people's preference, which is people are likely to purchase either liberal or conservative books, not both books.

Fig. 5.5 illustrates the graph of collaborations among scientists. We observed that, unlike the graph of karate-club and political books, this graph consists of many communities connecting weakly with each other. The community (A), (B) and (C) reveals core/periphery structures. The subjects of these communities correspond to (A) Biology, (B) World Wide Web, and (C) Physics. Table 5.1 shows the top 10 scientists receiving the highest scores. We noticed that Biologist were ranked at the top in eigenvector centrality, and scientists studying World Wide Web were also ranked in dense core. It should be noted that these scientists have published epoch-making papers such as "Emergency of scaling in random networks" cited by more than 20,000 papers and "Graph structure in the web" cited by more than 3000 papers. This indicates that dense core model captured not only one leading group, but also more than two leading groups in this network.

Next, we analyzed one large graph with our layout. Fig. 5.6 illustrates the network of references among blogs [2] on US politics in which the liberal blogs are colored by blue, and the conservative blog are colored by red, which was judges by the authors [2]. This network consists of 1490 nodes and 19025 edges. Nodes are blogs and two nodes are connected if they provide links each other. Note that we omitted the unidirectional links. We observed that this network was divided into two communities of liberals and conservatives. Within the liberal community, one highly cohesive core (A) exists, and multiple cores (B)-(E) are observed in

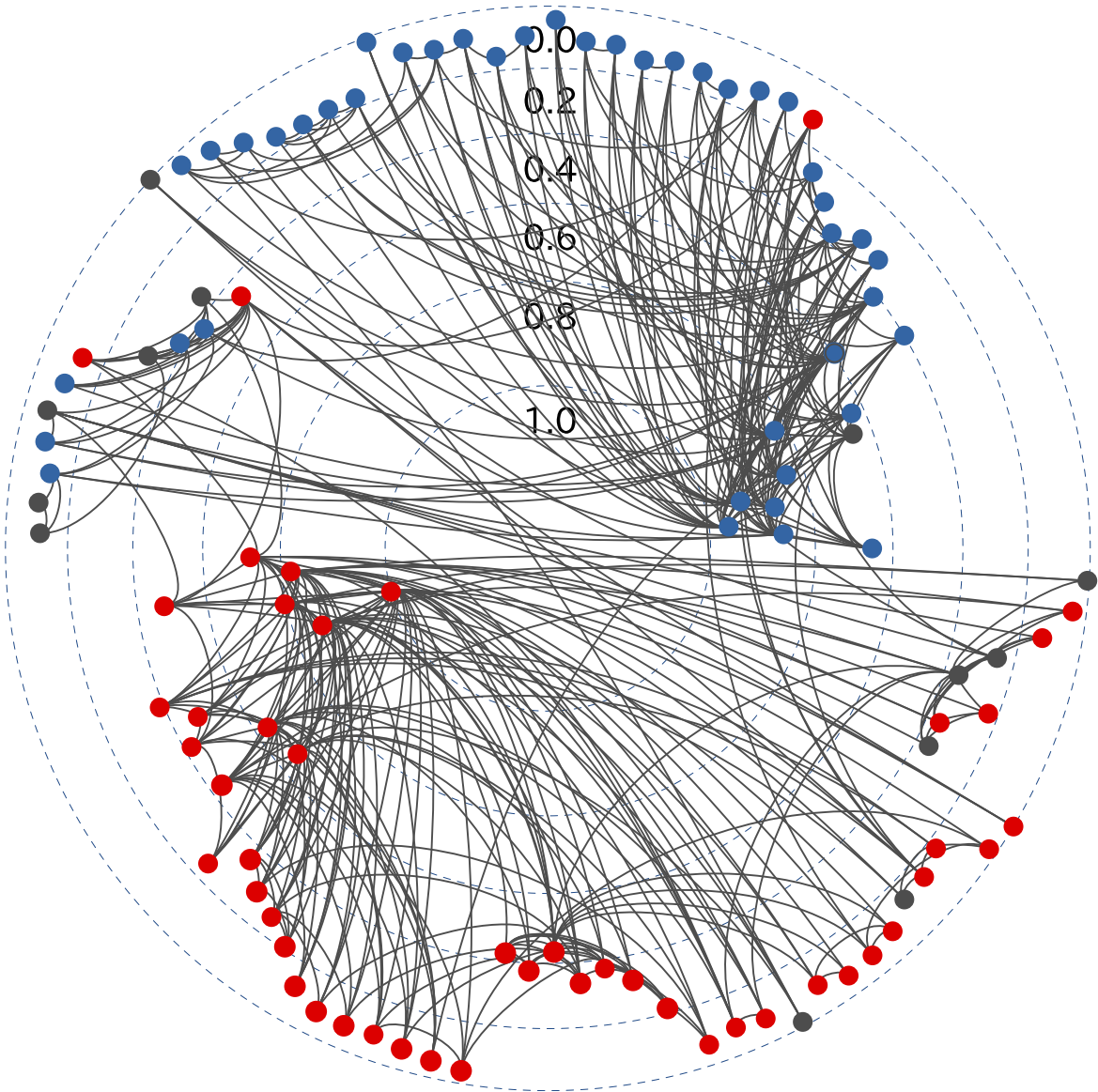


Fig. 5.4 Illustration political book network

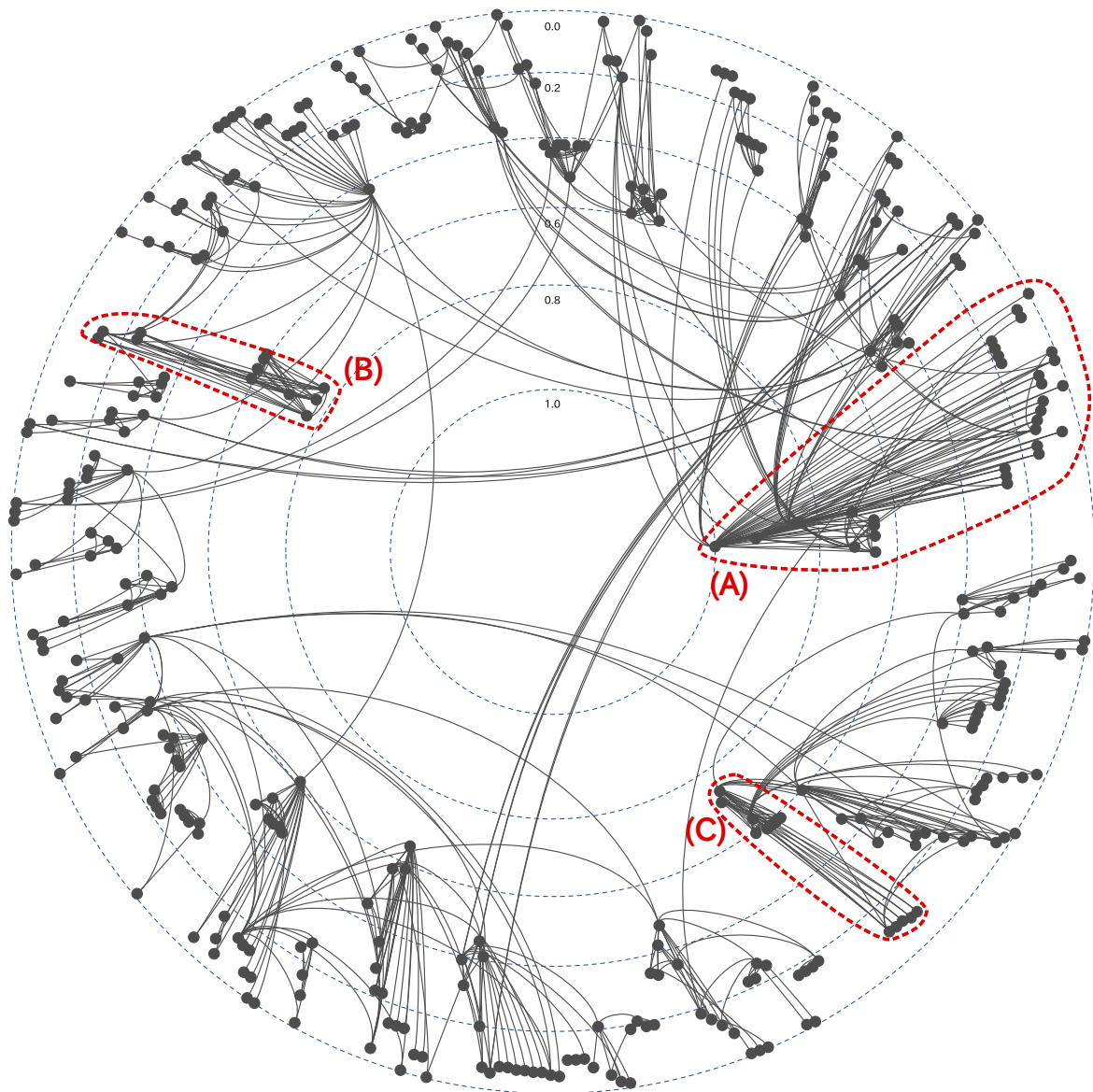


Fig. 5.5 Illustration of collaboration network

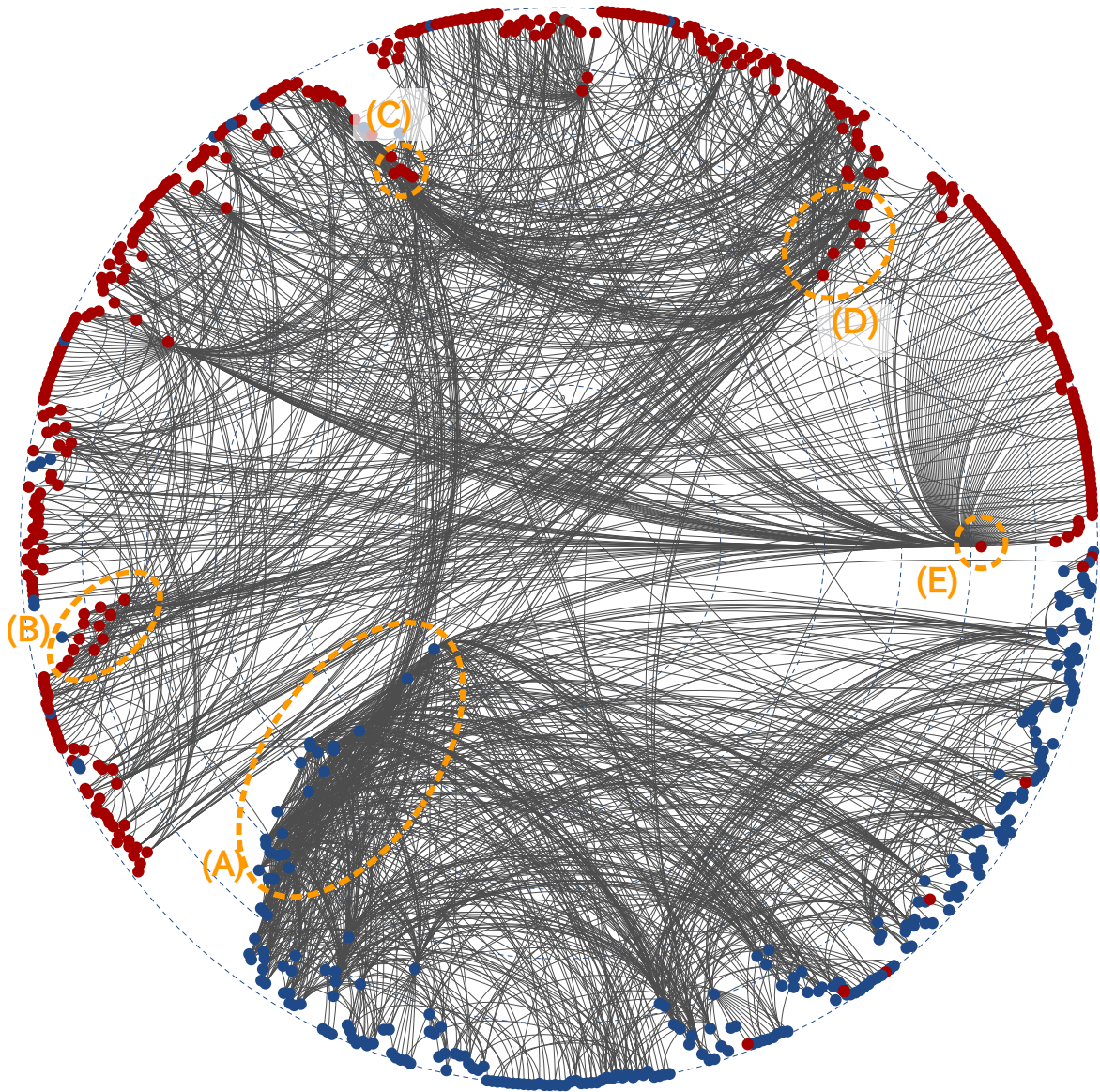


Fig. 5.6 Illustration of blog network

the conservative community. Unlike the network of scientists, the cores of conservatives have many connections with other cores. This indicates that the core blogs in the conservative communities have many agreement points, but do not have with the cores in the liberals.

5.5 Summary

We have shown that our graph diagram shows both community structures and the core-periphery relationships in the same diagram without losing the clarities of diagrams. From the diagram, multiple core/periphery structures are more clearly found. In our experiments, we confirmed that the proposed coreness measure assigned high score for leaders in karate club and scientists publishing epoch making papers. Unlike the traditional coreness measures, the proposed measure captured multiple cores in different positions such as conservative and liberal cores in political book networks, and core scientists in different field. One limitation is that it is computationally very expensive to compute. We need to construct a dendrogram to calculate the coreness measures, which requires $\mathcal{O}(|V|^3)$ even if we adopt heuristic algorithm. In addition to this, we need another computational time to calculate the centrality, which amounts to $\mathcal{O}(|V|^3)$ in order. In total, the calculation of our coreness measure requires computational cost $\mathcal{O}(|V|^3)$ in order, which is equal to the order of betweenness centrality, and larger than the eigenvector centrality. Our coreness measure, therefore, is not scalable for the size of networks, so that further developments are needed to analyze large graphs. One direction is to employ parallel computing to construct as well as to calculate the coreness measure. In particular, the given graph is divided into several subgraphs and then the subgraphs are processed with distributed computers. The parallel computing may reduce the actual computational time.

Chapter 6

Application to change detection task

The relation among people are always changing over time, and thus social networks are not static rather dynamic. In this chapter, here we investigate the dynamics of dense core model and its effectiveness in the task of detecting changes in dynamic networks [41].

6.1 Background

There has been considerable interest in the change analysis of social networks, such as communications among individuals through text messages [58], relations of opinions among bloggers [3, 29], and collaboration among coauthored scientists [57]. Detection of their anomalous changes is of great interests across various applications such as transitions of social interests [4, 77], outbreaks of rumors and infections [29]. A dynamic network is typically represented by a time series of graphs. A change in a graph sequence falls into one of three main categories according to the structure level of interest: global structure, local structure and the community-level structure. Changes in global structure are detected by a statistic of the whole graph such as the *network diameter* [45]. This is useful to understand how the graph evolves as a whole. Changes in local structures are detected around a node of interest or a small subgraph of interest. Such a change detection is useful when we know in advance that some nodes or subgraphs have a high potential of change [4, 37]. Changes in community-level structure are seen in between above two levels. This level focuses some densely connected groups of nodes, called *communities* [57]. In social networks, one group of communities repeats merging/splitting or growing/shrinking, while another group of communities remains unchanged for a long time [61]. The former groups are ad-hoc communities formed temporally with a short life. The latter group has a longer life and is often of key communities such as a board of directors [57, 63] and a group of opinion leaders

in blog networks [3]. Therefore, changes in such a stable community are one of important signs of crucial events in practice.

To carry out change detection on stationary communities, we have to 1) establish a methodology to extract only stationary communities, along with an appropriate definition of a community, and 2) choose an appropriate statistic showing changes in each community, and 3) establish a way to detect changes according to the value of the statistic. Here we mainly concentrate on the first issue and compare several community models for stationary community extraction, and then apply them to change detection in a time series of graphs.

6.2 Preliminaries

Suppose a dynamic system producing a random sequence of graphs

$$G^{(1)}, G^{(1)}, \dots, G^{(t)}, \dots,$$

where $G^{(t)} = (V, E^{(t)})$, $\forall t$. We assume the existence of a noise such that an edge in $E^{(t)}$ disappears with the probability ε ($0 < \varepsilon < \frac{1}{2}$) and an absent edge $e' \notin E^{(t)}$ turns to an existing edge with the probability ε as well. This means that, by this noise, the probability p of the appearance of an edge in $E^{(t)}$ changes to p' as

$$p' = p(1 - \varepsilon) + (1 - p)\varepsilon = (1 - 2\varepsilon)p + \varepsilon. \quad (6.1)$$

Note that $p' > p$ for $p < 1/2$ and $p' < p$ for $p > 1/2$. That is, density becomes close to $1/2$ by this noise. We assume that edges occur at random according to the above probability, while V remains unchanged. Under this assumption, the expected graph $\bar{G} = (V, \bar{E})$ with the noise is an weighted clique with edge weight p'_{ij} between every node pair of i and j . The *finite sample-averaged* weight in time period $[1, t]$ is given by

$$\hat{p}'_{ij} := \frac{1}{t} \sum_{s=1}^t \mathbf{1}\{e_{ij} \in E^{(s)}\}, \quad (6.2)$$

where $\mathbf{1}$ is the indicator function. We denote $\hat{G}^{[1,t]}$ as the graph with edge weight $\hat{p}'_{ij}$.

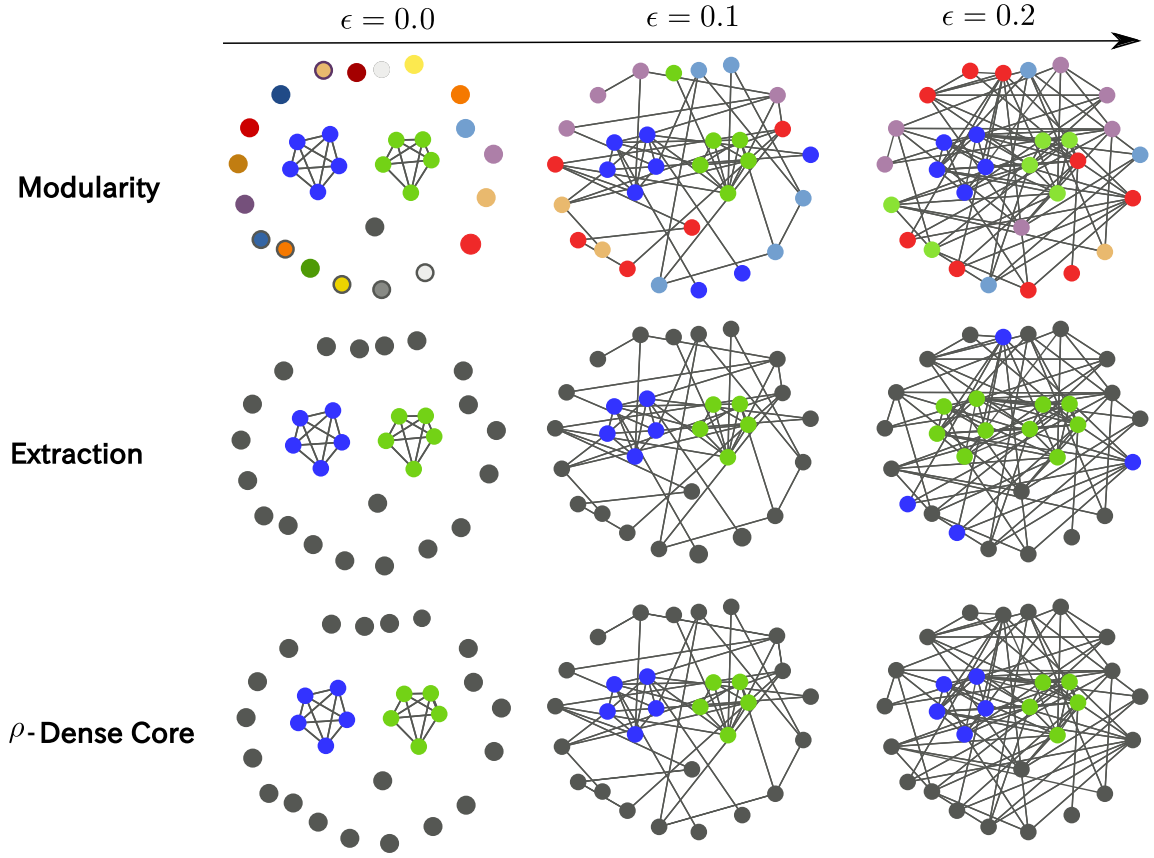


Fig. 6.1 Communities found by Modularity, Extraction and ρ -dense core ($\rho = 0.5$). The value of ε is the strength of noise. The color of nodes expresses the membership of the nodes, while a gray node does not belongs to any community.

6.3 Change detection of Communities

6.3.1 Community Models

Fig. 6.1 shows how stably three algorithms can find the communities in different levels of noises. Two main points of evaluation of algorithms are a) if they extract only stationary communities, and b) how robust they are against noise. From Fig. 6.1, we notice that 1) *Modularity* [57] assigns forcibly every node to one of the communities (against our evaluation point a)), 2) *Extraction* [89] is a little weak to the strong noise (against our evaluation point b)), and 3) Dense core succeeds to find only stationary communities in the noisy environment. We need such a robustness as the ρ -dense core shows in noisy communication networks.

Table 6.1 summarizes the representative quality functions including the above three algorithms. Other than these algorithms, various models have been proposed such as *Ratio cut* [32], *EigenCluster* [70] and *Normalized cut* [38], mainly in computer science. A group of

Table 6.1 Summary of graph partitioning criteria

Aiming Community	Criterion (Who, When)	Quality function $q(U)$	Robustness
Balanced Community	Ratio Cut (Hagen, '92)	$-\frac{B(U, \bar{U})}{ U }$	○
	EigenCluster (Sarkar, '96)	$\frac{W(U)}{ U }$	○
	Normalized Cut (Malik, '00)	$-\frac{B(U, \bar{U})}{W(U) + B(U, \bar{U})}$	○
	Modularity (Newman, '04)	$W(U) - \frac{(W(U) + B(U, \bar{U}))^2}{2W(V)}$	×
Dense Community	Graph Scan (*) (Wang, '08)	$\log \left(\frac{W(U)}{\mu(U)} \right)^{W(U)} + \log \left(\frac{W(V) - W(\bar{U})}{\mu(V) - \mu(\bar{U})} \right)^{(W(V) - W(\bar{U}))}$	×
	Extraction (Zhao, '11)	$\frac{ \bar{U} }{ U } W(U) - B(U, \bar{U})$	×
	ρ-Dense Core (Koujaku, '14)	$W(U) - \frac{\rho}{2} U (U - 1)$	○

(*) $\mu(U)$ is the expected number of edges inside U defined as $\mu(U) = \frac{1}{2} |U|(|U| - 1) \eta(V)$.

them prefers to divide a graph into the subgraphs of almost equal sizes. Such a trial is effective in some applications such as an equal distribution of resources, for example, an assignment of tasks in parallel computing. The number K of communities may be given by experimenters (Ratio cut, EigenCluster and Normalized Cut) or by referring to the complexity of networks as a whole (Modularity). These criteria forcibly assign every node to one of the communities. However, in social networks, such a forcible assignment is not always preferred since a large fraction of nodes have only few edges and they do not form communities (isolated nodes) [58, 89] which are hardly dealt with these criteria. Therefore, another group of quality functions are recently gathering interest such as *Graph Scan* [83], *Extraction* [89] and dense core that are able to deal with such isolated nodes. These criteria find the densest communities only. Graph Scan and Extraction find only a single community, while ρ -dense core finds multiple communities at once. In order to extract multiple communities, either of the former two criteria repeats the process in a greedy manner as long as the quality of a newly found community is higher than a pre-determined threshold. The dense core excludes automatically isolated nodes. For this reason, we compare these three criteria of Graph Scan, Extraction, and dense core in the following.

6.3.2 Framework of change analysis

The framework of change detection on the basis of communities is shown in Fig. 6.2 [41]. We construct the past graph G^- and the present graph G^+ by *double sliding windows*

$$G^- := \hat{G}^{[t-w+1, t]}, \quad G^+ := \hat{G}^{[t+1, t+w]} \quad (6.3)$$

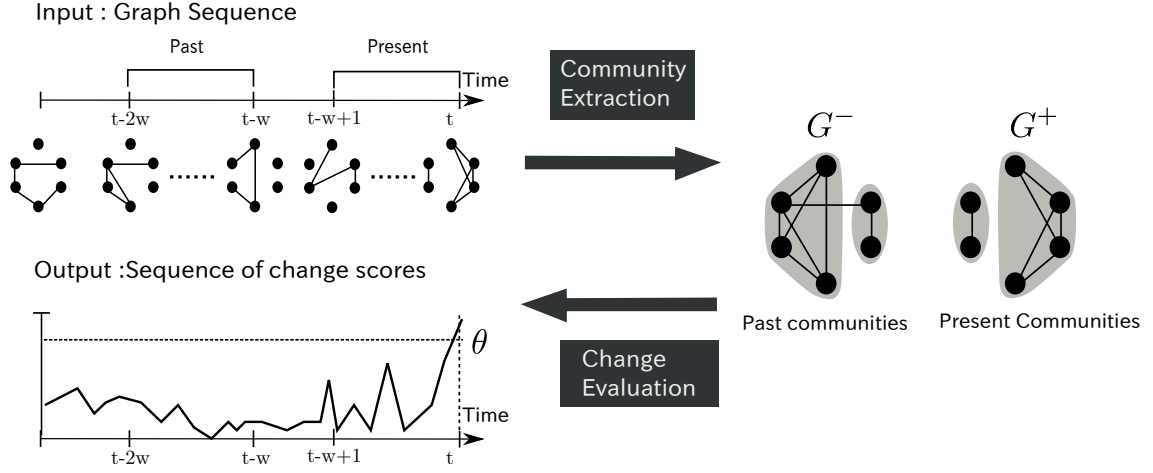


Fig. 6.2 Change detection framework with double sliding windows

where w is the window size. From these two graphs, we extract the past communities $\mathcal{U}^-$ and the present communities $\mathcal{U}^+$. Then, we calculate a *change score* z^t between $\mathcal{U}^-$ and $\mathcal{U}^+$ by *Variation of Information* (VI) [53]:

$$z^t = \text{VI}(\mathcal{U}^+, \mathcal{U}^-) = H(\mathcal{U}^+) + H(\mathcal{U}^-) - 2I(\mathcal{U}^+, \mathcal{U}^-), \quad (6.4)$$

where H and I are the entropy and the mutual entropy of the partition:

$$H(\mathcal{U}) = - \sum_{U \in \mathcal{U}} \frac{|U|}{|V|} \log \left(\frac{|U|}{|V|} \right), \quad (6.5)$$

$$I(\mathcal{U}, \mathcal{U}') = \sum_{U \in \mathcal{U}} \sum_{U' \in \mathcal{U}'} \frac{|U \cap U'|}{|V|} \log \left(\frac{|V||U \cap U'|}{|U||U'|} \right). \quad (6.6)$$

Finally, we regard the time t as a *changing point* if z^t is larger than a threshold θ . There is a trade-off between sensitivity and robustness. For detecting an abrupt change, we need a shorter w , but then we suffer from many false alarms instead. Reversely, with a longer w , we may detect a gradual change, but loose sensitivity. In general, abrupt changes are detected with a small value of w , while gradual changes are found with a large value of w .

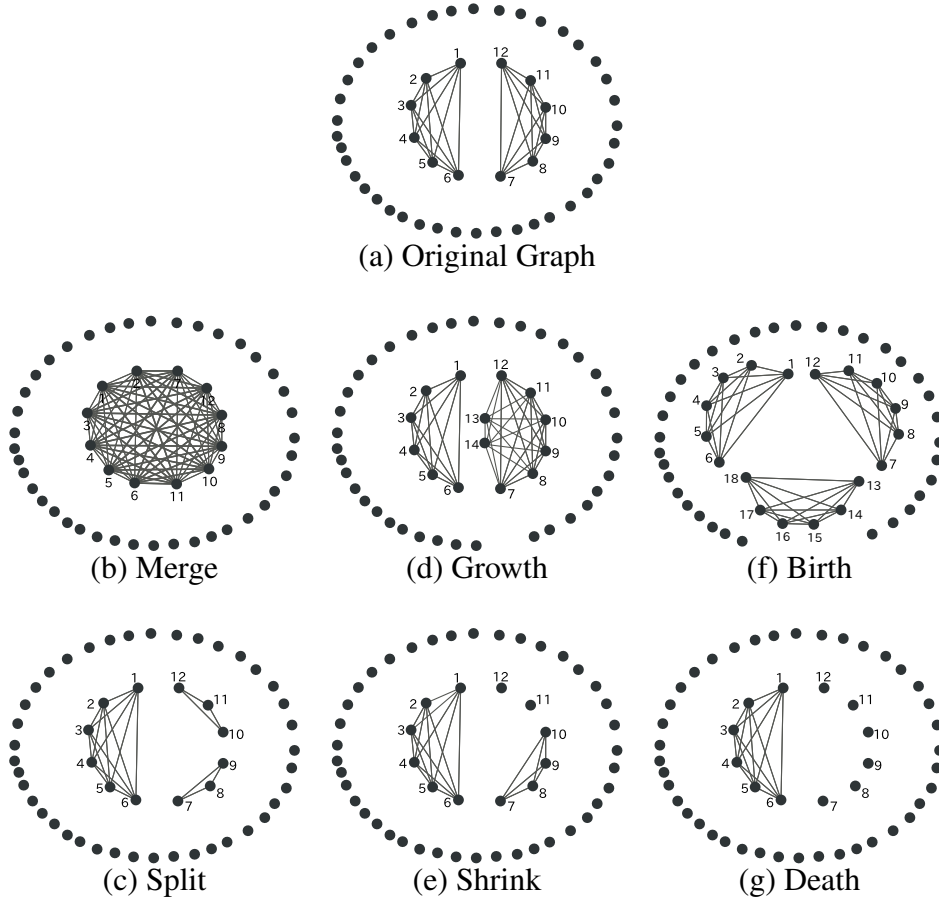


Fig. 6.3 Base graphs of six scenarios of changes

6.4 Experiment

6.4.1 Analysis of Synthetic graphs

We generated a sequence of 100 graphs $G^{(1)}, G^{(2)}, \dots, G^{(100)}$ based on seven graphs shown in Fig. 6.3. The original graph Fig. 6.3 (a) consists of two cliques expressing communities that are surrounded by 50 isolated nodes. From $t = 1$ to $t = 50$, the graph $G^{(t)}$ stays in the original with a flip noise with the probability ε . At $t = 51$, the graph generation mechanism changes and starts producing a difference sequence according to one of six scenarios of changes: (b) merge, (c) split, (d) growth, (e) shrink, (f) birth or (g) death.

We say that a detector succeeded if it detected at least one change within $[51 - w + 1, 51 + w]$, otherwise failed. A change was reported when the change score becomes larger than a pre-defined threshold θ . We generated 100 graph sequences in each scenario and evaluated the *Area Under the receive operator Curve* (AUC) for different values of θ . The curve is drawn on the plane of the false alarm rate and the recall rate. After drawing the curve,

Table 6.2 AUCs of community change detection tasks

	$\varepsilon = 0.1$						$\varepsilon = 0.2$					
	Merge	Split	Growth	Shrink	Birth	Death	Merge	Split	Growth	Shrink	Birth	Death
Modularity	0.46	0.66	0.77	0.75	0.73	0.76	0.31	0.66	0.68	0.74	0.80	0.72
Graph scan	0.98	0.39	0.96	0.94	0.98	1.00	0.70	0.65	0.62	0.69	0.72	0.65
Extraction	0.57	0.78	0.83	0.78	0.86	0.77	0.54	0.62	0.83	0.67	0.84	0.84
ρ -Dense core ($\rho = 0.5$)	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.82	0.90	0.89	0.97	0.98

we calculated the AUC by the trapezoid integration. We set the parameter of community detectors so as to achieve the highest AUCs i.e., we set significance level of Graph Scan and Extraction to 0.01 and the density parameter ρ of ρ -dense core to $\rho = 0.5$. In order to test the performance under difficult conditions, we set the very short window size $w = 5$ and strong noise intensity $\varepsilon = 0.1, 0.2$.

Table 6.2 shows the AUCs of four criteria. Graph Scan gained the high values of AUCs in five scenario with $\varepsilon = 0.1$ only, but no with $\varepsilon = 0.2$. The ρ -dense core achieved the highest AUCs in all scenarios of changes both for $\varepsilon = 0.1$ and $\varepsilon = 0.2$. Extraction showed stable and acceptable performance for both $\varepsilon = 0.1$ and $\varepsilon = 0.2$.

From these results, we can say that Modularity and Graph Scan are not appropriate for our goal. Therefore, we compared only the ρ -dense core and Extraction in the following experiments.

6.4.2 Analysis of E-mail interactions

Next, we examined the network constructed from the daily record [76] of e-mail interactions among 151 employees in Enron Inc. A node represents an employee and two nodes are connected at day t when one employee sends an e-mail to another employee in the day. We analyzed the network with ρ -dense core and Extraction. We trimmed the parameters of community detectors and window size so as to showed the clear peaks in their change scores. The parameter of Extraction and ρ -dense core are set to 0.01 and 0.3, respectively and the window size w is set to $w = 15$. The major social events are summarized in Table 6.3. In any event, it seems necessary for some persons to communicate to each other.

Fig. 6.4 shows the change scores of ρ -dense core and Extraction. The detected change points are shown by red bars. Note that the scales of Y -axis are different. We notice that the change scores of Extraction are larger than those of ρ -dense core. This indicates that the ρ -dense core is more pessimistic than Extraction. We examined detail changes at (A) detected by ρ -dense core. There is no clear change in the amount of e-mails (Fig. 6.4 (a)) and Extraction does not regard this day as a change point (Fig. 6.4 (c)). We visualized the

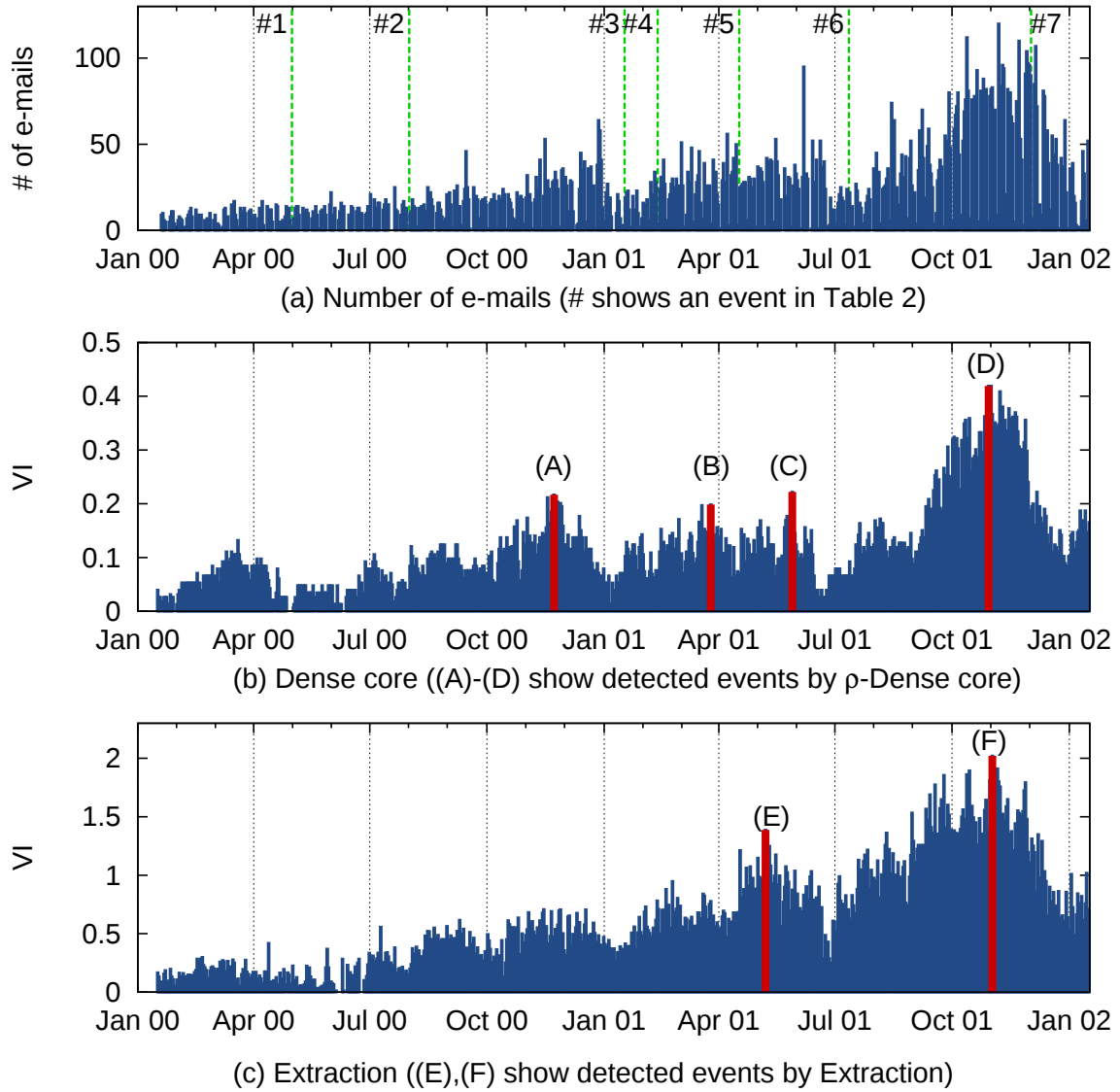


Fig. 6.4 Frequency of e-mails (a), change scores (VI) of ρ -dense core (b) and Extraction (c) in Enron data. Detected change points are red bars.

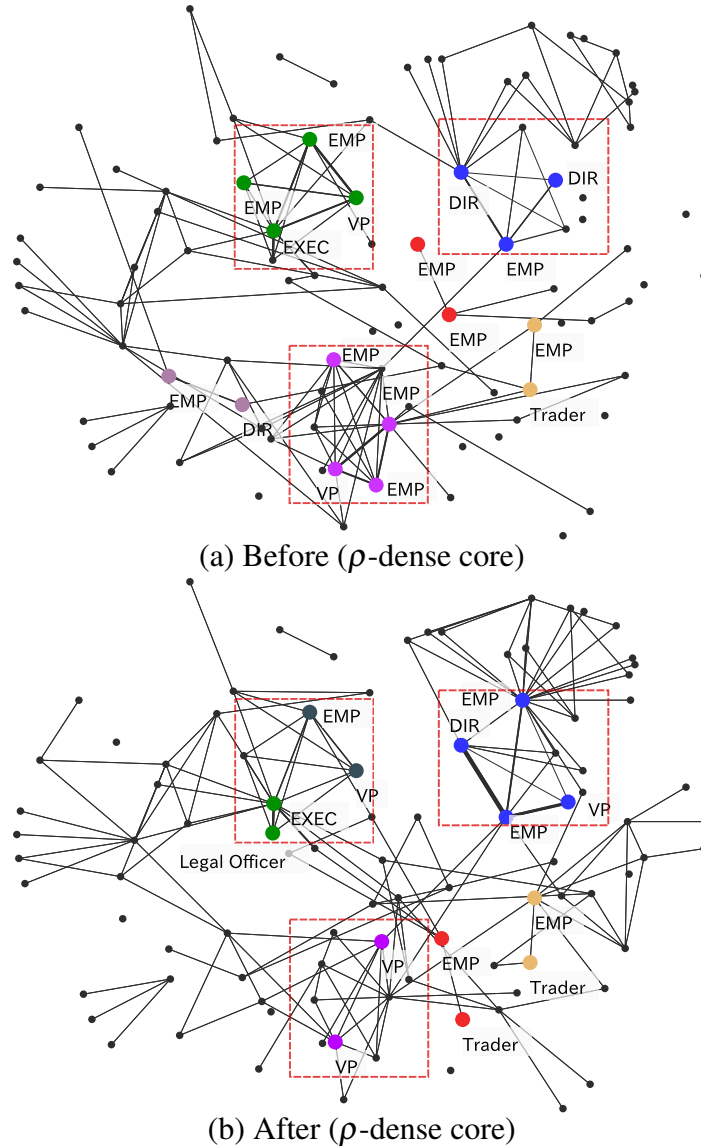


Fig. 6.5 Graphs before/after change point (A) where the width of an edge represents the value of its weight. EMP, MGR, VP, P and DIR stand for Employee, Managers Vice President, President and Director. ρ -dense core extracts the change of communities surrounded by dotted rectangles. For example, the community in the bottom rectangle seems to have been collapsed after this change.

Table 6.3 List of major events in Enron Inc.

ID	Month/Year	Event description
#1	May 00	Significant rise in energy price in California
#2	Aug 00	Enron's stock price attained its greatest value
#3	Jan 01	Electricity crisis led by Enron
#4	Feb 01	Alternation in CEO
#5	Apr 01	CEO gave offensive comments at public
#6	Jul 01	Alternation in CEO
#7	Sep 01	Large amounts of Enron stock was sold
#8	Dec 01	Enron bankrupted
#9	Jan 02	CEO was resigned

Table 6.4 List of major events in School.

ID	Day/Mon	Related Event
# 1	20 Sep	New semester
# 2	6 Dec	Winter Vacation begins
# 4	17 Jan	Thesis due
# 3	3 Feb	First day of classes

graphs before and after this change point (A) in Fig. 6.5. We can recognize some amount of change in the communities surrounded by the red dotted rectangles.

6.4.3 Analysis of Phone-call interactions

The network we examined last is constructed from the daily record of phone-call interactions among 106 students in MIT [23]. A node is a student and two nodes are connected if the corresponding two students communicate at day t . We set the window size w to $w = 15$ and the parameter of Extraction and ρ -dense core to 0.01 and 0.2, respectively which were determined as the same with the analysis of E-mail interactions. The major events are summarized in Table 6.4.

Fig. 6.6 shows the number of phone-calls and the change scores of two algorithms. The change points detected by the two algorithms are different although (C) and (F) are close. The graphs before/after the change point (A) and (D) are shown in Figs. 6.7 and 6.8. We can see that ρ -dense core finds the changes of many small communities, while Extraction finds the changes of larger communities.

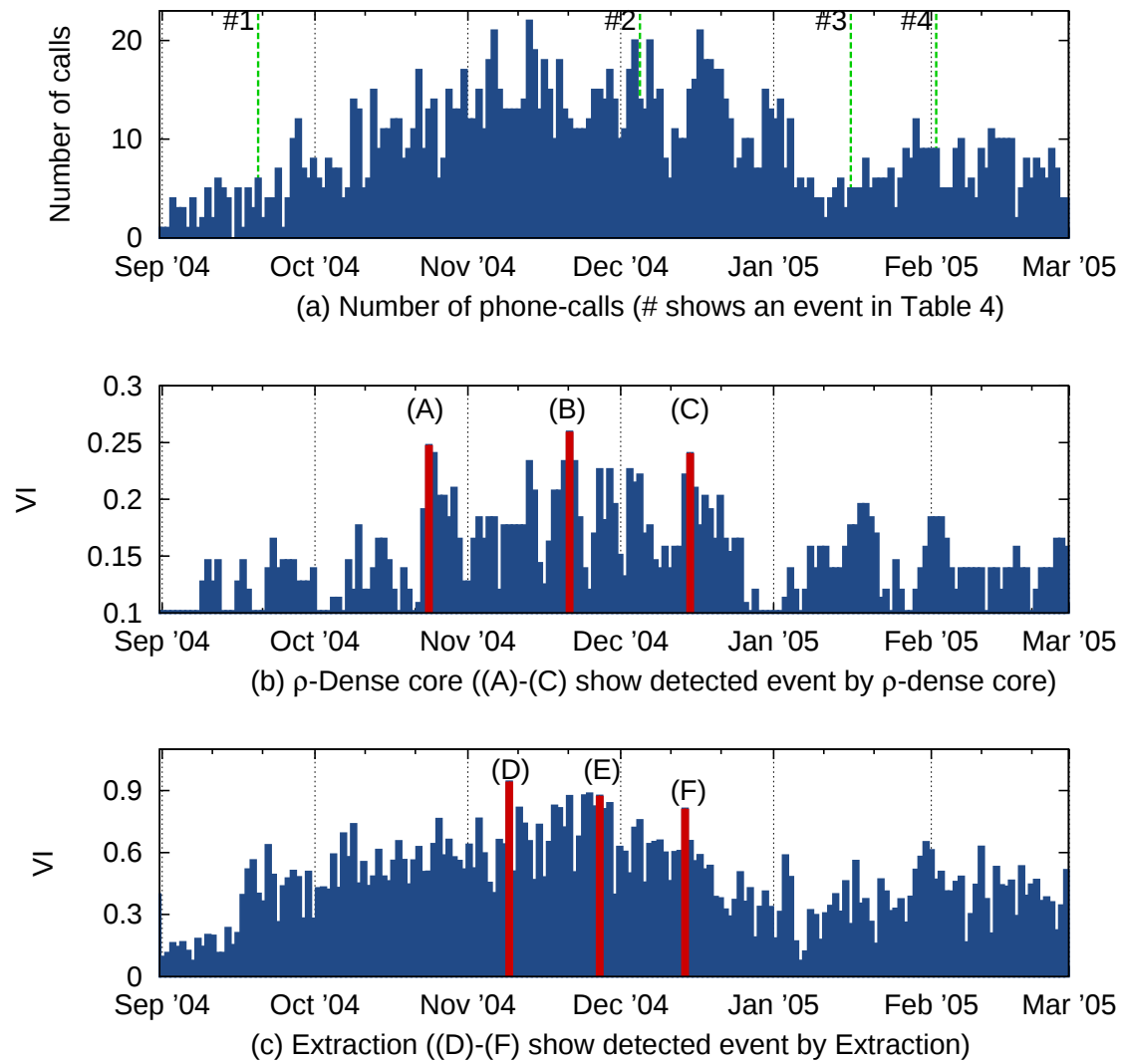


Fig. 6.6 Frequency of phone-calls (a), change scores (VI) of ρ -dense core (b) and Extraction (c) in MIT student data. The detected change points are red bars.

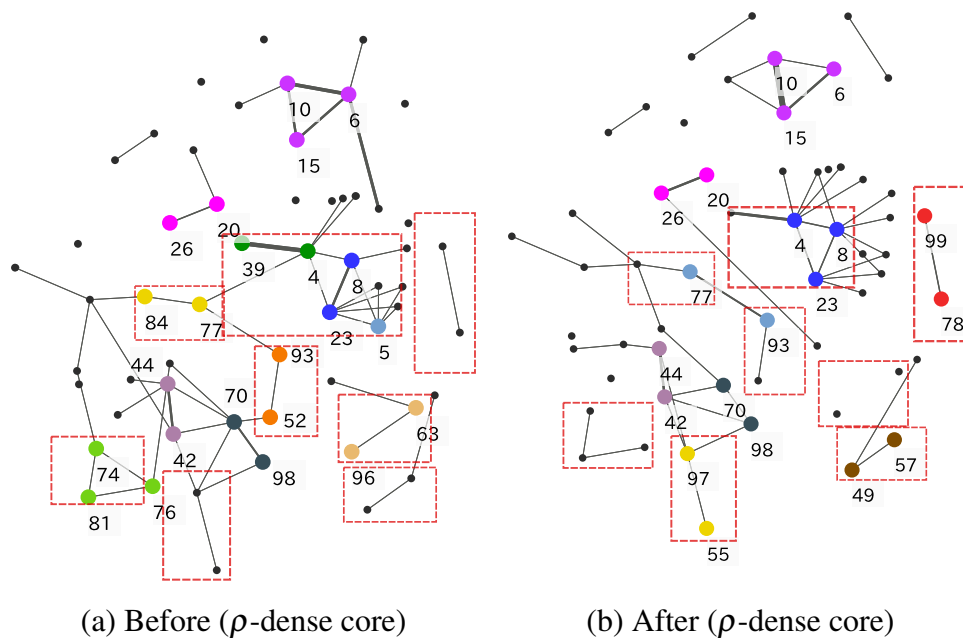


Fig. 6.7 Graphs before/after change point (A) detected by ρ -dense core. We can see that many small communities have changed.

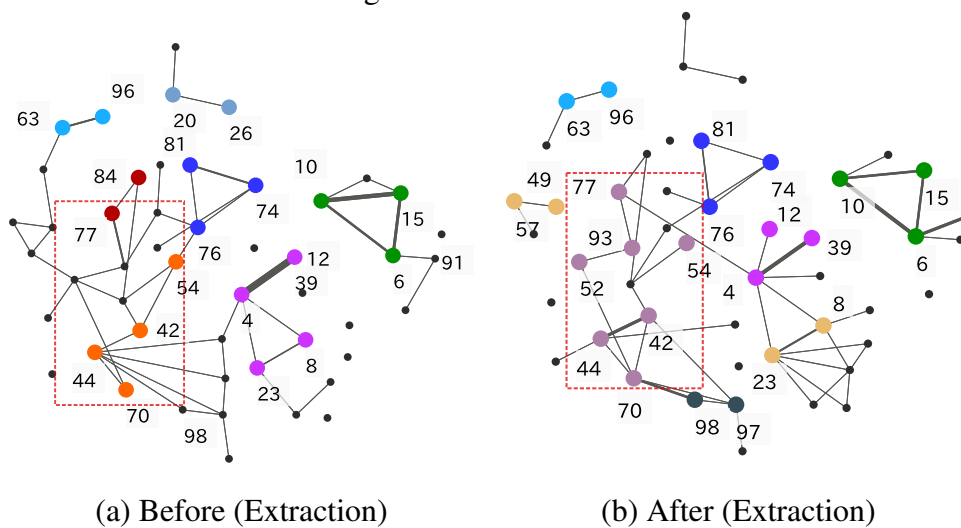


Fig. 6.8 Graphs before/after change point (D) detected by Extraction. Two small communities have been combined into a large community

6.5 Discussion

The ρ -dense core model focuses on only the changes in densely connected relatively small subgraphs, rather the changes of the whole graph. This is preferred if such a subgraph corresponds to a small but powerful/special community in practice. In the example of e-mail networks, monitored groups might be a community of executives or a project-based group. In MIT student data, such a community might be a small group of friends or a circle.

We notice that the change points detected by the two algorithms are totally different from the points of the social events in both networks. In some sense, this is natural because these algorithms monitor only relatively small but stationary communities and their changes are not always connected to social events happening in the whole networks. Reversely speaking, that is attraction of these algorithms. That is, from the detected community-level changes, we might find several interesting communities whose change may give a large impact later or show some potential cause of accidents.

ρ -dense core model has a parameter ρ to determine the standard level of the density of communities to be extracted. A larger value of ρ is useful to find smaller but densely connected communities, and a smaller value is useful for larger but sparser communities. However, it is hard to find an appropriate value of ρ for the problem at hand. Therefore, a few number of trial and error is necessary while monitoring the change scores and graphs.

6.6 Summary

We have presented a framework for finding changes in communities and compared the performance of several models including dense core model. Real-life graphs are very noisy; some messages may be intentionally hided, may be wrongly sent, and so on. We, therefore, consider a noisy environment in which the absent and present edges are added and removed, respectively. With the noisy model, we have numerically shown that the dense core model is the most robust and found the changes in the stationary communities almost perfectly. However, the proposed method is not sufficiently sensitive for the changes of real-life networks. This is due to the fact that the density as well as the size of networks is always changing over time, and the scale of changes is also changed accordingly. Therefore, the performance may be improved by using another change metric other than variation of information.

Chapter 7

Conclusion

Multiple core communities in noisy and large networks are findable with dense core model. A dense core is a subgraph in which any part is equally connected to some extent. It is advantageous in exploring block structured subgraphs. In addition, it can capture well-separated subgraphs by adjusting its parameter. On the basis of this unique property, we introduced a novel core/periphery model for finding multiple cores communities in networks. Compared with the traditional models, dense core model can capture finer and more cohesive cores, which is very useful for finding multiple local cores, not single global core in networks. It is also shown that dense core model shares many properties with pseudo cliques such as n -clan, k -core and ρ -dense. In addition, the spontaneity is supported by the fact that a set of dense cores gives the optimal partition of a graph.

In Chapter 3, we introduced a novel quality function that produces optimal partitions by dense cores. Typically, finding optimal partitions is hard due to the huge number of possible partitions to be inspected. We, therefore, introduce an efficient algorithm that finds optimal partitions within reasonable time in case the graph is sparse. In our simulation, the proposed algorithm finds optimal partition of graphs with up to 1,000 nodes and We also showed its effectiveness in finding core/periphery structures embedded in a graph. We also introduced a novel quality function that produces optimal partitions by dense cores. Our analysis on several real-life networks showed that the dense core is effective for finding key social groups such as scientists publishing epoch-making papers.

In Chapter 4, we extended the dense core model to capture more realistic communities containing sub-peripheries between cores and peripheries. We proposed two algorithms, which processes the network with nodes up to 10,000 nodes within 30 minutes. Analysis on several real-life networks had shown that the extended model found more detail structures.

One limitation of the extended dense core model is that it has too much information to interpret. Therefore, in Chapter 5, we presented a new graph diagram to illustrate both

the community structures and core-periphery relations found with dense core model. In the diagram, the nodes in the same communities are put as close as possible, and at the same time, core node and peripheral node are put as far as possible. We confirmed that the diagram of several real-life networks illustrated the multiple core/periphery structures more clearly than the traditional sociogram and dendrogram.

In Chapter 6, we applied dense core model to the change detection task of dynamic social networks. We compared the state-of-the-art community extraction algorithms and shown that the dense core model achieves the best robustness towards the flip noise. In addition, dense core model could detect several types of changes such as birth, grow, grow, shrink, birth and death. The effectiveness was confirmed on two real-life networks of communications through e-mails and phone-calls.

In summary, we presented dense core model for finding multiple core/periphery structures in noisy and large social networks. With this model, we are able to uncover the behaviors, patterns, and dynamics of influential social groups in a more detailed manner, e.g., how the behavior of core individuals in a working group affects its efficiency, lifetime, the time to make consensus, and so on. It should be noted that dense core model may omit star-structured cores. In addition, it is not scalable with respect to the size of networks. Therefore, future study includes the development of another core/periphery model to capture both star and block structured cores, and the reduction of the computational cost. It would be also interesting to investigate the effectiveness of the proposed algorithm in the task of anomaly detection, change point detection and influence maximization problem.

References

- [1] Abello, J., Resende, M. G. C., and Sudarsky, S. (2002). Massive Quasi-Clique Detection. In Rajsbaum, S., editor, *LATIN 2002: Theoretical Informatics*, pages 598–612. Springer Berlin Heidelberg.
- [2] Adamic, L. A. and Glance, N. (2005). The political blogosphere and the 2004 U.S. election. In *Proceedings of the 3rd international workshop on Link discovery - LinkKDD '05*, pages 36–43, New York, New York, USA. ACM Press.
- [3] Agarwal, N., Liu, H., Tang, L., and Yu, P. S. (2008). Identifying the influential bloggers in a community. *Proceedings of the international conference on Web search and web data mining - WSDM '08*, pages 207–218.
- [4] Akoglu, L. and Faloutsos, C. (2010). Event detection in time series of mobile communication graphs. In *27th Army Science Conference*, page 865, New York, New York, USA. ACM Press.
- [5] Aldridge, J. (1985). Discrete optimization algorithms (with pascal programs). *Microprocessors and Microsystems*, 9(2):86.
- [6] Arenas, a., Fernández, a., and Gómez, S. (2008). Analysis of the structure of complex networks at different resolution levels. *New Journal of Physics*, 10(5):053039.
- [7] Atmar, W. and Patterson, B. D. (1993). The measure of order and disorder in the distribution of species in fragmented habitat. *Oecologia*, 96(3):373–382.
- [8] Borgatti, S. P. and Everett, M. G. (2000). Models of core/periphery structures. *Social Networks*, 21(4):375–395.
- [9] Boyd, J. P., Fitzgerald, W. J., Mahutga, M. C., and Smith, D. a. (2010). Computing continuous core/periphery structures for social relations data with MINRES/SVD. *Social Networks*, 32(2):125–137.
- [10] Bron, C. and Kerbosch, J. (1973). Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577.
- [11] Caccetta, L. and Smyth, W. (1992). Graphs of maximum diameter. *Discrete Mathematics*, 102(2):121–141.
- [12] Cartwright, D. and Harary, F. (1956). Structural balance: a generalization of Heider’s theory. *Psychological Review*, 63(5):277–293.

- [13] Cohen, J. (2009). Graph Twiddling in a MapReduce World. *Computing in Science and Engineering*, 11(4):29–41.
- [14] Cohen, J. D. (2008). Trusses: Cohesive subgraphs for social network analysis. *National Security Agency Technical Report*.
- [15] Cplex, I. B. M. I. (2009). User 's Manual for CPLEX. *International Business Machines Corporation*, 46(53):157.
- [16] Cruz, J. D., Bothorel, C., and Poulet, F. (2012). Layout Algorithm for Clustered Graphs to Analyze Community Interactions in Social Networks. In *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 704–705. IEEE.
- [17] Csermely, P., Korcsmáros, T., Kiss, H. J., London, G., and Nussinov, R. (2013). Structure and dynamics of molecular networks: A novel paradigm of drug discovery. *Pharmacology & Therapeutics*, 138(3):333–408.
- [18] da Silva, M., Hongwu Ma, and An-Ping Zeng (2008). Centrality, Network Capacity, and Modularity as Parameters to Analyze the Core-Periphery Structure in Metabolic Networks. *Proceedings of the IEEE*, 96(8):1411–1420.
- [19] Davis, A., Gardner, B. B., and Gardner, M. R. (1941). *Deep South: A Social Anthropological Study of Caste and Class*. Univ of South Carolina Press.
- [20] Dinkelbach, W. (1967). On Nonlinear Fractional Programming. *Management Science*, 13(7):492–498.
- [21] Doerfel, M. L., Borgatti, S. P., Jones, C., Everett, M. G., Batagelj, V., and Mrvar, A. (1998). Network measure of social capital. *Connections*, 21(2):1–36.
- [22] Doreian, P. (1985). Structural equivalence in a psychology journal network. *Journal of the American Society for Information Science*, 36(6):411–417.
- [23] Eagle, N. and (Sandy) Pentland, A. (2006). Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268.
- [24] Fiduccia, C. and Mattheyses, R. (1982). A Linear-Time Heuristic for Improving Network Partitions. In *19th Design Automation Conference*, pages 175–181. IEEE.
- [25] Frahm, K. M. and Shepelyansky, D. L. (2012). Google matrix of Twitter. *The European Physical Journal B*, 85(10):355.
- [26] Freeman, L. C. (1977). A Set of Measures of Centrality Based on Betweenness. *Sociometry*, 40(1):35.
- [27] Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826.
- [28] Gómez-Gardeñes, J., Zamora-López, G., Moreno, Y., and Arenas, A. (2010). From Modular to Centralized Organization of Synchronization in Functional Areas of the Cat Cerebral Cortex. *PLoS ONE*, 5(8):e12313.

- [29] Gruhl, D., Liben-Nowell, D., Guha, R., and Tomkins, A. (2004). Information diffusion through blogspace. *ACM SIGKDD Explorations Newsletter*, 6(2):43–52.
- [30] Hage, P. and Harary, F. (1995). Eccentricity and centrality in networks. *Social Networks*, 17(1):57–63.
- [31] Hagen and Kahng (1992a). A new approach to effective circuit clustering. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 422–427. IEEE Comput. Soc. Press.
- [32] Hagen, L. and Kahng, A. B. (1992b). New spectral methods for ratio cut partitioning and clustering. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11(9):1074–1085.
- [33] Harary, F. (1994). Sum graphs over all the integers. *Discrete Mathematics*, 124(1-3):99–105.
- [34] Heimo, T., Kumpula, J. M., Kaski, K., and Saramäki, J. (2008). Detecting modules in dense weighted networks with the Potts method. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(08):P08007.
- [35] Herman, I., Melancon, G., and Marshall, M. (2000). Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43.
- [36] Hoffman, A. J. and Ray-Chaudhuri, D. K. (1965). On the line graph of a symmetric balanced incomplete block design. *Transactions of the American Mathematical Society*, 116:238–238.
- [37] Idé, T., Papadimitriou, S., and Vlachos, M. (2007). Computing correlation anomaly scores using stochastic nearest neighbors. *Proceedings - IEEE International Conference on Data Mining, ICDM*, pages 523–528.
- [38] Jianbo Shi and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.
- [39] Kandiah, V. and Shepelyansky, D. L. (2012). PageRank model of opinion formation on social networks. *Physica A: Statistical Mechanics and its Applications*, 391(22):5779–5793.
- [40] Kernighan, B. W. and Lin, S. (1970). An Efficient Heuristic Procedure for Partitioning Graphs. *Bell System Technical Journal*, 49(2):291–307.
- [41] Koujaku, S., Kudo, M., Takigawa, I., and Imai, H. (2015a). Community Change Detection in Dynamic Networks in Noisy Environment. In *Proceedings of the 24th International Conference on World Wide Web*, pages 793—798, Florence, Italy. International World Wide Web Conferences.
- [42] Koujaku, S., Takigawa, I., Kudo, M., and Imai, H. (2015b). Dense core model for cohesive subgraph discovery. *Social Networks*.

- [43] Lambiotte, R. (2010). Multi-scale modularity in complex networks. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2010 Proceedings of the 8th International Symposium on*, pages 546–553.
- [44] Lee, S. H., Cucuringu, M., and Porter, M. a. (2014). Density-based and transport-based core-periphery structures in networks. *Physical Review E*, 89(3):032810.
- [45] Leskovec, J., Kleinberg, J., and Faloutsos, C. (2005). Graphs over time. In *Proceeding of the 11th ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05*, page 177, New York, New York, USA. ACM Press.
- [46] Luce, R. D. (1950). Connectivity and generalized cliques in sociometric group structure. *Psychometrika*, 15(2):169–190.
- [47] Luce, R. D. and Perry, A. D. (1949). A method of matrix analysis of group structure. *Psychometrika*, 14(2):95–116.
- [48] Lundberg, G. a. and Moreno, J. L. (1937). Who Shall Survive? A New Approach to the Problem of Human Interrelations. *American Sociological Review*, 2(4):542.
- [49] Ma, H.-W. and Zeng, A.-P. (2003). The connectivity structure, giant strong component and centrality of metabolic networks. *Bioinformatics*, 19(11):1423–1430.
- [50] Mark S. Granovetter (1973). The strength of weak ties. *American Journal of Sociology*, 78(6):1360–1380.
- [51] Marsden, P. V. (2011). *Survey methods for network data*, pages 370–388. Sage Publications, London.
- [52] McAuley, J. J., da Fontoura Costa, L., and Caetano, T. S. (2007). Rich-club phenomenon across complex network hierarchies. *Applied Physics Letters*, 91(8):084103.
- [53] Meilă, M. (2007). Comparing clusterings—an information based distance. *Journal of Multivariate Analysis*, 98(5):873–895.
- [54] Mokken, R. J. (1979). Cliques, clubs and clans. *Quality & Quantity*, 13(2):161–173.
- [55] Mullins, N. C., Hargens, L. L., Hecht, P. K., and Kick, E. L. (1977). The Group Structure of Cocitation Clusters: A Comparative Study. *American Sociological Review*, 42(4):552.
- [56] Newman, M. E. J. (2004). Detecting community structure in networks. *The European Physical Journal B - Condensed Matter*, 38(2):321–330.
- [57] Newman, M. E. J. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582.
- [58] Newman, M. E. J., Forrest, S., and Balthrop, J. (2002). Email networks and the spread of computer viruses. *Physical Review E*, 66(3):035101.
- [59] Newman, M. E. J. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113.

- [60] Opsahl, T., Colizza, V., Panzarasa, P., and Ramasco, J. J. (2008). Prominence and Control: The Weighted Rich-Club Effect. *Physical Review Letters*, 101(16):168702.
- [61] Palla, G., Barabási, A.-L., and Vicsek, T. (2007). Quantifying social group evolution. *Nature*, 446(7136):664–667.
- [62] Pons, P. and Latapy, M. (2011). Post-processing hierarchical community structures: Quality improvements and multi-scale view. *Theoretical Computer Science*, 412(8-10):892–900.
- [63] Priebe, C. E., Conroy, J. M., Marchette, D. J., and Park, Y. (2005). Scan Statistics on Enron Graphs. *Computational and Mathematical Organization Theory*, 11(3):229–247.
- [64] Rombach, M. P., Porter, M. a., Fowler, J. H., and Mucha, P. J. (2014). Core-periphery structure in networks. *SIAM Journal on Applied Mathematics*, 74(1):1–27.
- [65] Ronhovde, P. and Nussinov, Z. (2009). Multiresolution community detection for megascale networks by information-based replica correlations. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 80(1):016109.
- [66] Rossa, F. D., Dercole, F., and Piccardi, C. (2013). Profiling core-periphery network structure by random walkers. *Scientific reports*, 3:1467.
- [67] Saavedra, S., Stouffer, D. B., Uzzi, B., and Bascompte, J. (2011). Strong contributors to network persistence are the most vulnerable to extinction. *Nature*, 478(7368):233–235.
- [68] Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika*, 31(4):581–603.
- [69] Saito, K. and Yamada, T. (2006). Extracting Communities from Complex Networks by the k-dense Method. In *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW'06)*, pages 300–304. IEEE.
- [70] Sarkar, S. and Boyer, K. (1996). Quantitative measures of change based on feature organization: eigenvalues and eigenvectors. *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 478–483.
- [71] Schaible, S. (1983). Fractional programming. *Zeitschrift für Operations Research*, 27(1):39–54.
- [72] Scott, J. (1990). *A matter of record: Documentary resources in social research*. John Wiley & Sons.
- [73] Scott, J. (2012). *Social Network Analysis: A Handbook*. SAGE Publications, third edit edition.
- [74] Seidman, S. B. (1983). Network structure and minimum degree. *Social Networks*, 5(3):269–287.
- [75] Seidman, S. B. and Foster, B. L. (1978). A graph-theoretic generalization of the clique concept. *The Journal of Mathematical Sociology*, 6(1):139–154.
- [76] Shetty, J. and Adibi, J. (2004). Enron email dataset. *USC Information Sciences Institute, Tech. Rep.*

- [77] Sun, J., Faloutsos, C., Papadimitriou, S., and Yu, P. S. (2007). GraphScope: parameter-free mining of large time-evolving graphs. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 687–696. ACM.
- [78] Tolliver, D. a. and Miller, G. L. (2006). Graph partitioning by spectral rounding: Applications in image segmentation and clustering. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1053–1060. IEEE.
- [79] Tomita, E., Tanaka, A., and Takahashi, H. (2006). The worst-case time complexity for generating all maximal cliques and computational experiments. *Theoretical Computer Science*, 363(1):28–42.
- [80] Trenkler, G. (1997). *Handbook of Matrices*, volume 25. Wiley.
- [81] Uno, T. (2007). An Efficient Algorithm for Enumerating Pseudo Cliques. In *Algorithms and Computation*, pages 402–414. Springer Berlin Heidelberg.
- [82] von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416.
- [83] Wang, B., Phillips, J. M., Schreiber, R., Wilkinson, D., Mishra, N., and Tarjan, R. (2008a). Spatial Scan Statistics for Graph Clustering. *SDM*, pages 727–738.
- [84] Wang, G., Shen, Y., and Ouyang, M. (2008b). A vector partitioning approach to detecting community structure in complex networks. *Computers and Mathematics with Applications*, 55(12):2746–2752.
- [85] Wang, N., Zhang, J., Tan, K.-L., and Tung, A. K. H. (2010). On triangulation-based dense neighborhood graph discovery. *Proceedings of the VLDB Endowment*, 4(2):58–68.
- [86] Wang, Y., Di, Z., and Fan, Y. (2011). Identifying and characterizing nodes important to community structure using the spectrum of the graph. *PLoS ONE*, 6(11):e27418.
- [87] You, F., Castro, P. M., and Grossmann, I. E. (2009). Dinkelbach’s algorithm as an efficient method to solve a class of MINLP models for large-scale cyclic scheduling problems. *Computers and Chemical Engineering*, 33(11):1879–1889.
- [88] Zhang, S. and Zhao, H. (2012). Community identification in networks with unbalanced structure. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 85(6):066114.
- [89] Zhao, Y., Levina, E., and Zhu, J. (2011). Community extraction for social networks. *Proceedings of the National Academy of Sciences*, 108(18):7321–7326.
- [90] Zhou, S. and Mondragón, R. J. (2004). The rich-club phenomenon in the internet topology. *IEEE Communications Letters*, 8(3):180–182.

Appendix A

Dinkelbach's Algorithm

The general fractional programming can be formulated as the following problem

$$\min_{\mathbf{x} \in S} \frac{f(\mathbf{x})}{g(\mathbf{x})},$$

where $f, g, h, j = 1, \dots, m$ is real-valued function defined on a set $S = \{x \in \mathbb{R} : h_j(\mathbf{x}) \leq 0, j = 1, \dots, m\}$. Furthermore, the following assumption is also made:

$$g(\mathbf{x}) \geq 0, \text{ on } S.$$

Dinkelbach developed an algorithm to find optimal solution by solving non-linear fractional subproblems. Let $\theta(\mathbf{x}) = f(\mathbf{x})/g(\mathbf{x})$. The original Dinkelbach's algorithm is stated as follows.

1. Let $\mathbf{x}_1$ be a feasible point of S , $\delta_1 = \theta(\mathbf{x}_1)$, and $k = 1$.
2. By means of a convex programming method, solve the following sub problem:

$$\min_{\mathbf{x} \in S} f(\mathbf{x}) - \delta_k g(\mathbf{x}), \tag{A.1}$$

and denote any solution point by $\mathbf{x}_{k+1}$

3. If the solution is zero, stop and $\mathbf{x}_k$ is optimal. Otherwise, set $\delta_{k+1} = \theta(\mathbf{x}_{k+1})$ and $k = k + 1$, and go to Step 2.

It is shown later that this algorithm converges an optimal solution super linearly if $f(x)$ and $g(x)$ are convex functions [87].

Appendix B

Spectral clustering

Given a graph $G = (V, E)$ with adjacency matrix A where $A_{ij} = 1$ if $\{i, j\} \in E$, otherwise $A_{ij} = 0$. Let us express a bipartition U_1, U_2 over V ($U_1 \cup U_2 = V$, $U_1 \cap U_2 = \emptyset$) with an indicator vector as $\mathbf{x}$ with entries

$$\mathbf{x} = (x_i), x_i = \begin{cases} \frac{|U_2|}{\sqrt{|U_1|}} & (i \in U_1) \\ -\frac{|U_1|}{\sqrt{|U_2|}} & (i \in U_2). \end{cases} \quad (\text{B.1})$$

Then, the ratio cut objective is rewritten using laplacian matrix defined in (4.5):

$$\begin{aligned} \frac{B(U_1, U_2)}{|U_1||U_2|} &= B(U_1, U_2) \left(\frac{1}{|U_1|} + \frac{1}{|U_2|} \right) \frac{1}{|U_1| + |U_2|} \\ &= B(U_1, U_2) \left(\frac{|U_1| + |U_2|}{|U_1|} + \frac{|U_1| + |U_2|}{|U_2|} \right) \frac{1}{(|U_1| + |U_2|)^2} \\ &= B(U_1, U_2) \left(\frac{|U_2|}{|U_1|} + \frac{|U_1|}{|U_2|} + 2 \right) \frac{1}{|V|^2} \\ &= B(U_1, U_2) \left(\sqrt{\frac{|U_2|}{|U_1|}} + \sqrt{\frac{|U_1|}{|U_2|}} \right)^2 \frac{1}{|V|^2} \\ &= \frac{1}{2|V|^2} \sum_{i \in V, j \in V} a_{ij} (x_i - x_j)^2 \\ &= \frac{1}{2|V|^2} \mathbf{x}' \mathbf{L} \mathbf{x} \end{aligned}$$

where L is the laplacian matrix. We also have

$$\begin{aligned}\sum_{i \in V} x_i &= \sum_{i \in U_1} \sqrt{\frac{|U_2|}{|U_1|}} - \sum_{i \in U_2} \sqrt{\frac{|U_1|}{|U_2|}} = 0, \\ \sum_{i \in V} x_i^2 &= \sum_{i \in U_1} \frac{|U_2|}{|U_1|} + \sum_{i \in U_2} \frac{|U_1|}{|U_2|} = n.\end{aligned}$$

In other words, the indicator vector $\mathbf{x}$ is orthogonal to the constant one vector $\mathbf{1}$ and also its norm is n . Altogether the ratio cut problem can be equivalently rewritten as

$$\min_{\mathbf{x}} \mathbf{x}' \mathbf{L} \mathbf{x}, \text{ subject to } \mathbf{x}' \mathbf{1} = 0, x_i \text{ is defined in (B.1), } \|\mathbf{x}\| = \sqrt{n}.$$

This is a discrete optimization problem in which $\mathbf{x}$ is allowed to take two particular values, and is an NP-hard problem. In spectral partitioning, a relaxation is made by discarding the discreteness condition and instead allowing x_i to take real value. This leads to the following relaxed optimization problem.

$$\min_{\mathbf{x}} \mathbf{x}' \mathbf{L} \mathbf{x}, \text{ subject to } \mathbf{x}' \mathbf{1} = 0, \|\mathbf{x}\| = \sqrt{n}.$$

By the Rayleigh-Ritz theorem [80], the solution is given by the eigenvector corresponding to the second smallest eigenvalue of $\mathbf{L}$. In order to obtain a partition of the graph, we need to re-transform the real-valued solution into a discrete indicator vector. One of the simplest way is to order the nodes according to the value of $\mathbf{x}$ and to examine the objective of the cut between i th and $i + 1$ th nodes for all i :

$$\begin{cases} j \in U_1 (x_j \leq x_i) \\ j \in U_2 (x_j > x_i) \end{cases},$$

and choose the cut that achieves the minimum ratio cut objective.