Journal of Statistics & Management Systems
ISSN 0972-0510 (Print), ISSN 2169-0014 (Online)
Vol. 28 (2025), No. 6, pp. 1073–1090
DOI : 10.47974/JSMS-1415

Detection of overlap community in social networks founded on game theory

Mohammad Wali Hairan †
Ali Delavarkhalafi *
Saeid Alikhani §
Department of Mathematical Science
Yazd University
89175-741 Yazd
Iran

Abstract

We study a novel approach based on game theory for the identification of overlapping associations and hierarchical structures within a social network. The approach models association detection as a coalition formation game, where individuals in the network act as rational players striving to enhance the overall utility of their groups. To achieve this objective, players collaborate and establish alliances. Each player has the flexibility to join multiple coalitions, and smaller coalitions can merge with one another to create larger ones, as long as the merger leads to an improvement in the coalition's utility. This approach enables the simultaneous identification of overlapping associations and hierarchical structures in the social network. The utility (service) function of apiece coalition is definite as a grouping of a income function and a price function. The income function events the extent to which the group's objectives are enhanced, while the cost function counts the level of communication between coalition members and the remaining network. This article investigates two categories of methods based on cooperative and non-cooperative game, and reports the results of tests performed on both artificial and real-life network data. The findings demonstrate that the CoCo-game method and the overlapping method outperform other approaches and result in larger coalitions compared to the clique method and the CoCo-game method. Additionally, the CoCo-game method and the observation method indicate superior performance in identifying associations within clique networks, karate graphs, and Zachary graphs. Notably, in clique graphs, they exhibit exceptional performance of up to

† *ORCID-ID:* **0009-0005-2915-1966**
* *ORCID-ID:* **0000-0002-1178-1178**
§ *ORCID-ID:* **0000-0002-1801-203X**

† *E-mail:* **mwalihairan8@gmail.com**
* *E-mail:* **delavarkh@yazd.ac.ir (Corresponding Author)**
§ *E-mail:* **alikhani@yazd.ac.ir**

90 percent. In the Zachary graph, they perform well by employing a value of μ = 0.1 for the cross edges section. Furthermore, in networks comprising 10000 nodes and 1000 edges, the CoCo-game method surpasses the clique method in terms of performance and exhibits reduced vertex overlap.

Subject Classification: *35Q91, 05C82.*

Keywords: *Game theory, CoCo-game, Clique, Overlapping, Detection of overlapping communities, Social networks.*

1. Introduction

Discovering community is significant for understanding the physical features of social network and improving user-centricity [1], service such as identifying influential users and setting up efficient recommender systems for targeted marketing. The increasing popularity of online social network, community recognition has attracted much care. In the environment of social networks, people's behavior is not completely independent; there are cooperations and conflicts between people. Therefore, social networks can be analyzed founded on game theory.

Online social networks provide the means for communication between individuals over the internet. One of the key features of social network is their communities organization. Communities are subsets of network nodes such that the ratio of internal edges within the community is higher than the edges between nodes of that community and other communities. Community detection is important for understanding the structural possessions of social network and improving user-centric amenities such as identifying influential users and setting up effectual systems aimed at sharing, viewing, and redistributing content for targeted marketing. Given the increasing popularity of connected social networks, community detection has received significant attention. In the social network environment where individuals' behaviors are interdependent, interactions between decision-making individuals exist, and so social network can be examined founded on Game theory. In this approach, each network node is considered as a rational agent. Game theory is a mathematical instrument aimed at learning complex behaviors among rational agents in a game, each striving to maximize their own utility function. For more information on the applying game theory refer to [2, 3].

To this end, it is necessary to first define the set of players, actions, and utility functions. Community detection often utilizes both cooperative and non-cooperative game theory approaches. Non-cooperative game theory methods focus on individual rewards, whereas cooperative approaches prioritize the benefits of the group as a whole. Therefore, improving current methods to get logical results is a challenging task. Each individual, as a self-interested representative, selects communities for membership or departure based on their own utility measurement.

There are various algorithms and methods developed for community detection in social networks. These methods utilize different approaches such as modularity optimal, hierarchical cluster, spectral clustering, and label propagation, among others [4]. The excellent of method be contingent on the characteristics of the network then the specific goals of the analysis. Some algorithms aim to maximize the modularity score, which measures the excellence of the community construction. Others focus on finding communities with high intra-cluster connectivity and low inter-cluster connectivity. In recent years, machine learning techniques, such as bottomless learning and graph neural networks, have also been applied to community detection problems [5].

This article investigates two categories of methods based on cooperative (or a colleague) and non-cooperative game theory and presents the results of experiments conducted on together artificial and real-world graphs. The findings designate that the CoCo-game method and the overlapping method outperform other approaches and result in larger coalitions compared to the batch method and the Coco-game method. Moreover, the CoCo-game method and the observation method exhibit superior performance in identifying associations in group networks, karate graphs, and Zachary graphs. Notably, in clique graphs, they achieve exceptional performance, reaching up to 90 percent accuracy. In the case of the Zachary graph, they perform well when utilizing a value of $\mu = 0.1$ for the NMI graph section. Furthermore, in networks containing of 10000 nodules and 1000 edges, the CoCo-game method surpasses the collection method in terms of efficiency and demonstrates a reduction in vertex overlap.

We have organized the remainder of this paper in the following manner: Section 2 examines background research and Section 3 investigates the game-theory agenda for communities detections. 4 investigate a game-theoretic framework for coalition formation. Section 5 investigates the Algorithm for communities detection in game theory.

2. Related works

The initial result connected to communities detections in computer skill is often attributed to the graph divider problematic [6].

One drawback is that traditional graph partitioning methods assume that the amount of nodes and the amount of collections to be divided are known in advance, which is different after the communities' detections problem where the number of community is usually unidentified. Another limitation is that these methods focus on minimizing the number of journey edges between partitions, without considering the inherent connections amongst nodes cutting-edge the graph. This may not accurately identify communities.

Newman introduced the concept of modularity, which addressed these drawbacks and marked a significant advancement in community detection [7]. Modularity is definite based on a divider of node in a graph. Let $G = (V, E)$ be an purposeless graph representing a social networks by n node and m edge. Assuming apiece node v fits to communities c_v, the modularity Q of a specific community partition is calculated using the following formula:

$$Q = \frac{1}{2m}\sum_{u,v}\left(A_{uv} - \frac{d_u d_v}{2m}\right)\delta(c_u, c_v),$$

where the adjacency matrix A of G, A_{uv} indicates the presence (1) or absence (0) of an edge between nodes u and v. Also, d_u besides d_v represent the grades (degrees) of nodes u and v, respectively, and $\delta(c_u, c_v)$ is a function that outputs 1 if a certain condition holds, and 0 otherwise, that equals 1 if $c_u = c_v$ (i.e., u and v stand in the similar communities), and 0 otherwise.

Newman's modularity measurement quantifies the amount of edges within community and compares it to the predictable number of such edge if the similar number of edge were randomly rewired. Optimizing modularity is believed to correspond to effectively identifying communities [8].

However, modularity-based algorithms have some limitations. Firstly, they assume that community do not interconnect, which is often not the circumstance in real world scenarios. Secondly, modularity founded approach hurt from resolution limit [9], meaning they tend to merge small communities.

Community detection is a complex task with various approaches, each having its own considerations and limitations. Researchers and practitioners have developed different methods and algorithms to address this problem, considering factors such as network structure, individual behaviors, and group dynamics. Some approaches focus on identifying communities based on local connections and graph properties, while others take into account the influence of individual agents or the cooperation among groups.

It's significant to note that no single approach is universally applicable or perfect in all scenarios. The excellent of method be contingent on the specific physiognomies of the network, the objectives of the analysis, and the assumptions made about the behavior of individuals or groups. Additionally, limitations can arise from issues like scalability, noise in the data, overlapping communities, or the subjective definition and interpretation of what constitutes a community.

The motivation of this work is to explore and refine community detection methods to overcome these limitations and adapt near the evolving countryside of social networks.

Here we state some definitions that we need in this paper.

Definition 1: A k-clique, represented by c_k, is a set of k nodes where each node has an edge to every other node within the set.

For example, in Figure 1, nodes $1, 3, 4$, and 6 are fully connected, making them a 4-clique.

Definition 2: A pair of k-cliques are adjacent if, and only if have $k - 1$ nodes in common.

Figure 1 demonstrates the adjacency of the 4-cliques $\{1, 3, 4, 6\}$ and $\{1, 3, 6, 9\}$.

A *k-clique community* is a group of vertices that belong to a set of interconnected k-cliques (complete subgraphs with k nodes). These cliques are linked together in a chain-like fashion. While the exact relationship between these communities and real-world communities is not always straightforward, this definition has several advantages ([7]):

Clear Definition: It is formally precise and does not rely on subjective or complex methods.

Deterministic: The results are always the same given the same input.

Allows Overlap: Communities can share members, reflecting real-world situations.

Local Clarity: The membership of a vertex within a community is easily determined.

However, existing methods for finding these communities have limitations, particularly for medium-sized values of k, due to high memory requirements.

This work aims to address these limitations by:

Improving Efficiency: Utilizing a highly efficient algorithm for identifying k-cliques.

Reducing Memory Usage: Introducing a second algorithm that provides approximate results with lower memory demands, suitable for larger graphs.

In essence, this research focuses on making the identification of k-clique communities more practical for larger graphs by optimizing both computational speed and memory usage.

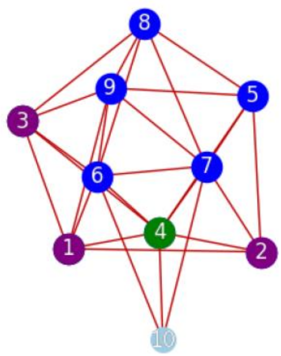

Figure 1
A simple social network

In Figure 2, we have $k = 4$ and $z = 2$. A community, defined by 4-clique percolation, is comprised of the nodes within the following 4-cliques:

$$\{1, 3, 4, 6\}, \{1, 3, 6, 9\}, \{3, 6, 8, 9\}, \{6, 7, 8, 9\}, \{5, 7, 8, 9\}, \{2, 5, 7, 8\}, \{2, 4, 5, 7\}.$$

The set $\{4, 6, 7\}$, which forms a 3-clique, acts as a bridge connecting these 4-cliques. This bridging 3-clique is created by the 2-clique and is not itself a member of the identified 4-clique community.

For example, if a new k-clique $\{4,6,7,10\}$ is found:

CPMZ: Would create a single community encompassing all vertices: $\{1,2,\ldots,10\}$.

Exact CPM: Would create two separate communities: {1, 2, 3, 4, 5, 6, 7, 8, 9} and {4, 6, 7, 10}.

This illustrates a key difference between CPM and CPMZ in how they handle overlapping cliques ([10]).

Theorem 3: [10] (CPMZ validity). *The CPMZ algorithm produces a set of combined k-clique communities. Importantly, each original CPM community is completely contained within one and only one of these combined communities.*

3. Game-theory agenda for communities detections

In the game-theory agenda for community's detections, social networks are viewed as games where self-interested agents make decisions regarding their participation in communities based on their individual utility functions. The objective of each agent is to maximize their own utility, which leads to the emergence of community structures as a result of these individual decisions.

This framework acknowledges that joining a community can bring benefits to an agent, such as social interactions and resource sharing. However, there are also associated costs or drawbacks, such as membership fees or time commitments. An agent's utility function consists of two components: the gain from joining communities and the loss incurred by joining them.

In the absence of specific drawbacks associated with community membership, agents are incentivized to join every community, reflecting the general understanding that participation in communities often requires some form of sacrifice. This may involve investing resources or making compromises to be a part of a community.

The utility of an agent is determined by the difference between their gain function, which captures the benefits obtained from joining communities, and their loss function, which represents the costs or drawbacks associated with community participation.

By adopting this game-theoretic perspective, the framework aims to imprisonment the dynamic of community creation in social networks and provide insights into the factors that influence agents' decisions regarding community membership. It offers a straightforward and intuitive approach to understanding community formation driven by self-interested behavior within the social network.

Further exploration and analysis within this game-theoretic framework can deepen our understanding of community dynamics, the strategic behavior of agents, and the emergence of community's constructions in social networks. This approach has the potential to shed light on various aspects of community detection, including the stability and evolution of communities, the role of incentives and disincentives in community formation, and the impact of individual decision-making on the overall community structure [11].

Here, we deliver a formal definition of our communities creation game. We are assumed an fundamental static associate graphs, denoted as $G = (V, E)$, where n represents the number of nodes, i.e., $n = |V|$, and m signifies the number of edges, i.e., $m = |E|$. It is assumed that the graph G has no directionality or edge weights, although our results container be widespread to directed and weighted graph as healthy. Each element in the set V can be referred to as a Factor a bulge or node.

Apiece node in the graph selects a collection of community that it desires to joint. The customary of all likely community is meant as $[k] = \{1,2,\dots,k\}$, wherever k is a polynomial function of n. It is significant to note that the actual community construction we obtain may consist of a significantly smaller number of communities compared to k. Detecting a view amount of community is together theoretically infeasible and unrealistic in practice.

The value of the ith Factor or agent, denoted as $v_i \in V$, is quantified using a improvement function $g_i(\cdot)$ and a cost function $\ell_i(\cdot)$. Additionally, for any domestic of function $\{f_1, f_2, \dots, f_n\}$ clear over the or Factors agents, we define the aggregate function $f(\cdot)$ as the sum of these individual functions, i.e., $f(\cdot) = \sum_{i\in[n]} f_i(\cdot)$. For example, we have $g(\cdot) = \sum_{i\in[n]} g_i(\cdot)$ and $\ell(\cdot) = \sum_{i\in[n]} \ell_i(\cdot)$.

By establishing this formal framework, we aim to analyze the community formation process as a game and investigate the utility-maximizing strategies of the Factors or agents. This method allows us to examine the interaction between gain and loss functions, leading to a better understanding of community's formations dynamics in social networks [11].

4. A game-theoretic framework for coalition formation

Communities detections is a vital task in network analysis as it involves identifying clusters of nodes in a network that are thickly connected within themselves while being poorly integrated with the rest of the network. One approach to community detection is the utilization of game theory, particularly the theory of coalition formation games, which models the behavior of agents forming coalitions to achieve their goals. In a coalition creation game, the nodes of a network are treated as players, and they form coalitions by joining together in groups. These coalitions consist of actors who share common interests and work collaboratively towards goals such as maximizing their collective influence or minimizing their collective cost. The success of a coalition depends on the strength of the relationships between its members and the resources they bring to the coalition.

The coalition formation game approach to community detection involves modeling the network as a game, where the players are the nodes of the network. Each player has a set of neighbors, and the strength of their relationships with those neighbors determines their profit in the game. Players can form alliances by joining their neighbors, and the success of an alliance depends on the strength of the relationships between its members. To classify communities in the network, the coalition formation game is played iteratively, and players form and modify coalitions in each iteration. This process lasts pending a stable set of coalitions is touched, where no player has an incentive to leave their current coalition and join another. A stable set of alliances represents communities in the network, where nodes within each coalition are densely connected, while nodes between coalitions are sparsely connected

Several algorithms based on coalition creation game-theory can be employed for community's detections in network analysis. One popular algorithm is the Core-Periphery (CP) model, which aims to identify a small group of nodes that form the core of the network and a larger group of nodes that form the periphery.

In the CP model, a variant of the coalition formation game is utilized. The core nodes, which are the most influential players in the network, act as attractors for other nodes. Peripheral nodes, on the other hand, form coalitions to gain access to the core and benefit from their influence. This algorithm seeks to find a balance between the core nodes' desire to maintain their dominance and the peripheral nodes' goal of accessing the core.

Another algorithm based on coalition formation is the Coco-game algorithm. This algorithm focuses on creating small coalitions within the network. It starts with individual nodes forming coalitions and then allows these coalitions to merge or split based on the players' interests and the strength of their relationships. The Coco-game algorithm aims to find stable coalitions where players are satisfied with their participation.

The clique algorithm, on the other hand, aims to create large coalitions within the network. It begins with each node forming a coalition of its own and then allows these coalitions to merge if they have a sufficiently high level of connectivity. The goal is to form large coalitions that encompass densely connected groups of nodes.

These algorithms, along with other approaches founded on coalition creation game-theory, provide different perspectives and techniques aimed at community detection in complex networks. They offer the advantage of capturing the dynamics of node interactions and the formation of coalitions, which can lead to the discovery of meaningful and cohesive communities within networks.

Another algorithm based on the theory of coalition formation games is the modularity optimization algorithm, which aims to maximize the modularity of the network. Modularity events the thickness the relative density of connections within versus between communities. The modularity optimization algorithm employs a variant of the coalition formation game, where players form coalitions to maximize the modularity of the resulting network.

Overall, the game-theoretic framework of coalition formation provides a powerful tool for community's detections in complex networks. By modeling the network as a game and utilizing coalition formation to identify communities, this approach can uncover communities that may not be easily distinguishable using traditional clustering algorithms.

Definition 4: Assume that S_i represents a set of communities called a coalition in a network $G = (V, E)$, and $e(S_i)$ denotes the number of edges between the internal vertices of the coalition S_i, and $d(x)$ represents the degree of vertex x.

In this circumstance, the coalition utility or value function $v(S_i)$ is defined as follows:

$$v(S_i) = \left(\frac{2e(S_i)}{\sum_{x\in S_i} d(x)} - \left(\frac{\sum_{x\in S_i} d(x)}{2m}\right)^2\right). \quad (1)$$

Essentially, $v(S_i)$ is an improved utility function, known as the Newman-Girvan model which defined as:

$$Q = \sum_{c=1}^{n_c} \left(\frac{e(S_c)}{m} - \left(\frac{d(S_c)}{m}\right)^2\right).$$

This function is used to measure the quality of the coalition structure, where n_c is the amount of coalitions and m is the entire number of edges in the network.

4.1 *Beneficial performance*

Definition 5: Let S be a coalition of $G = (V, E)$, then the utility function $v(S)$ of S is clear by the following equation:

$$v(S) = \frac{2e(S)}{\sum_{x\in S} d(S)} - \alpha(\frac{\sum_{x\in S} d(S)}{2\beta|E|})^2. \quad (2)$$

The primary term and the additional term of Equation 2 are named interest functions and cost functions S, correspondingly. It is the increasing functions of the relation of the edges inside S to the over-all degree of vertices in S. The cost functions are used to express the relationship between a coalition's cumulative degree and the network's total degree. A large benefit value indicates many interactions between players in S and those outside of S. A large cost value indicates many interactions between players within S and a weak connection to the rest of the network. Equation 1 implies that the formation of a coalition transports benefits to its memberships, then the benefits are imperfect by the cost of coalitions formation.

The parameter α is a factor of scale which is regulate the cost of the coalitions S. Note that $\alpha \in [0,1]$ and $\alpha = 0$ is used for starting a coalition that is continuously beneficial. In this circumstance, the utility function $v(S)$ because it is clear only by the gain functions, it is superadditive. So $v(V) = 1, V(\{x\}) = 0, x \in V$ This income that the Utility function of the outstanding coalitions takes a maximum value, though the utility function of a singleton coalition has a value of 0. When $\alpha = 1$, the cost of starting coalition are maximum and therefore, $v(V) = 0$. So

$$V(\{x\}) = -\alpha(\frac{\sum_{x\in V} d(S)}{2\beta|E|})^2, x \in V.$$

Therefore, the set of small coalitions are rarely optimal constructions. In addition, whatever $d(x)$ be smaller, determination of $v(x)$ be bigger. This income that vertices by small degrees are more willing to cooperate with additional vertices to recover their applications. The parameter β is additional parameter that is used to set the setting of the coalition S, and note that $\beta \in (0,1]$. From $\beta = 1$ income that the setting of coalitions is the entire network; and

$\beta < 1$ income that the setting of local coalition of complex network usually includes many vertices and the edges, thus the cost of a coalition with lower degree might be deserted compared to the whole network.

Definition 6: When $\Gamma = \{S_1, S_2, \dots, S_k\}$, represents a community structure, the total utility $v(\Gamma)$ is determined by the equation below:

$$v(\Gamma) = \sum_{S_i \in \Gamma} v(S_i),$$

where $v(S_i)$ is the Utility function of coalitions S_i.

The total utility $v(\Gamma)$ measures the overall quality of the coalition structure Γ, which is a divider of the nodes in the network into disjoint coalition. The higher the total utility, the better the coalition structure. The goal of coalition formation games is to find the coalition construction that minimizes the over-all utility.

The utility function is a mathematical function used to measure the benefits or payoffs that a coalition of players can gain from working together in a coalitional game. In Definition 6, the utility function of a coalition S in a graph $G = (V, E)$ is clear by Equation 2 as follows:

$$v(S) = \alpha \cdot gain(S) - \beta \cdot cost(S)$$

where $\alpha \in [0,1]$ and $\beta \in (0,1]$ are two parameters used to adjust the cost and context of forming coalitions, respectively. The improvement function and cost function are defined as follows:

$$gain(S) = \frac{\sum_{(i,j) \in E_S} w_{i,j}}{\sum_{i \in S} d_i},$$

$$cost(S) = \frac{\sum_{i \in S} d_i}{\sum_{i \in V} d_i},$$

where E_S is the set of edges with both endpoints in S, $w_{i,j}$ is the weighted of the link between i and j, d_i is the degree of vertex i in the network, and V is the set of all nodes in the network.

The improvement function $gain(S)$ measures the proportion of edges inside the coalitions S compared to the over-all degree of the nodes in S, while the cost function $cost(S)$ measures the proportion of the over-all degree of the vertices in S compared to the total degree in the network. The improvement function represents the benefit or gain that the coalition S can obtain by working together, though the cost function signifies the cost or disadvantage of forming the coalition S compared to not forming any coalition.

The parameter α adjusts the price of forming the coalitions S, where $\alpha = 0$ income there is no cost for starting the coalition, while $\alpha = 1$ indicates maximal costs for forming the coalition. The parameter β regulates the context of the coalitions S, where $\beta = 1$ income the context of the coalition is the entire network, while $0 < \beta < 1$ income that the setting of the coalition is a native part of the network.

The utility function $v(S)$ measures the net benefit of forming the coalition S, where a positive value designates a net gain, and a negative value designates a clear loss. By computing the utility function for different coalitions, players can determine the optimal coalition structure that maximizes their payoffs. The optimal coalition structure is one that leads to the highest utility function value, or equivalently, the highest net benefit.

In summary, the utility function is an significant concept in coalitional game theory as it lets us to quantify the benefits and costs of forming coalitions and helps us identify the optimal coalition structures. The parameters α and β allow us to adjust the trade-off between the gain and cost of forming coalitions based on the specific context of the network and the goals of the players. By using the utility function, players can make informed decisions about which coalitions to form and which to avoid, leading to more effective and efficient collaborations in different real-world applications.

5. The Algorithm for Communities Detection in Game Theory

In general, the CoCo-game algorithm presents a promising approach for community detection in networks using game theory concepts. By iteratively optimizing the utilities of nodes, the algorithm can discover stable communities that reflect the underlying network structure [12].

It has been shown that the CoCo-game algorithm does better than extra community detection algorithms in footings of accuracy and robustness, especially in sparse and heterogeneous networks. Additionally, this algorithm is computationally efficient and can handle large-scale networks.

Algorithm 1: CoCo-game Algorithm for Community Detection

Initialization Phase: In this phase, each node is assigned to a separate community. In other words, $\Gamma = \{S_1, S_2, \dots, S_k\}$
forms a set of communities.

Game Adjustment Phase: For each node, a utility function is defined to indicate its willingness to interact with other nodes within a community. This utility function is based on the number of a node's neighbors within its own community and the number of its neighbors in other communities.

In stage 2, for each S_i and S_j in Γ, S_{ij} is added if $e(S_i, S_j) \neq 0$ and the conditions $\Delta\nu(S_i, S_{ij}) > 0$ and $\Delta\nu(S_j, S_{ij}) > 0$
are satisfied. In this case, the new Γ is formed as
$\Gamma = \Gamma + \{S_{ij}\} - \{S_i\} - \{S_j\}$.

Game Iteration Phase: In this phase, the game between each node and its neighbors is repeated to improve their utilities. In each iteration, a node is selected to join a neighboring community or invite its neighbors to its own community. The goal of this phase is to select an operation that creates improvement in a node's utility.

Finalization Phase: The game iteration stops when no node can improve its utility by changing its strategy. In stage 4, for each $x \in V$, if
$x \notin S_i$ and $V_x(S_i) > \omega$, then $S_i = S_i + \{x\}$
will join. Also, if $x \in S_i$ and $v_x < \varepsilon$, then $S_i = S_i - \{x\}$ will leave.

Post-Processing Phase: In this phase, communities with significant overlap in their members are merged. This is done by calculating the similarity between the members of the communities and combining communities that have higher similarity.
Finally, the final structure of the stable communities
$\Gamma = \{S_1, S_2, \dots, S_k\}$.

The algorithm of the Clique Percolation Method is as follows:

Algorithm 2: Clique Percolation Method (CPM)

```
Require: Graph G, clique size k
Initialize empty list of cliques: cliques
Initialize empty list of communities: communities
nodes → set of nodes in G
for each node in nodes do
    candidates → set of neighbors of node
    Add node to candidates
    for each clique in cliques do
        if candidates is a subset of clique then
            Break
        end for
    end for
    if no clique contains candidates then
        Add candidates to cliques
    end if
end for
for each clique in cliques do
    for j from i + 1 to length of cliques do
        other_clique → cliques[j]
        if size of intersection between clique and other_clique is at least k − 1 then
            merged_clique → union of clique and other_clique
            for each community in communities do
                if merged_clique is a subset of community then
                    Break
                end if
            end for
            if no community contains merged_clique then
                Add merged_clique to communities
            end if
        end if
    end for
end for
return communities
```

5.1 *Evaluation of real networks*

It appears that you are likening the results of two dissimilar community detection algorithms, Clique and CoCo-game, in the context of the Karate Club network. The Karate Club network is a famous social network rummage-sale as a Benchmark for community detection algorithms.

According to the information provided, at the first level of analysis with $k = 1$, the Clique algorithm identifies 6 communities and one overlapping vertex (vertex 0) in the Karate Club network. This means that Clique divides the network into 6 distinct communities, and vertex 0 has connections to vertices in different communities.

On the other hand, the CoCo-game algorithm detects 4 communities and three overlapping Nodes (vertices 0, 32, and 33) in the same network. CoCo-game corrects the community structures exposed in the previous work by Newman and Girvan (2004), where the network was divided into two parts (or three parts in the case of the Karate Club network). The algorithm identifies 4 communities in total, and three vertices (0, 32, and 33) have connections to multiple communities.

It's important to note that community detection algorithms can produce different results based on their underlying methodology and assumptions. The number of communities and the presence of overlapping vertices can vary depending on the algorithm used. Each algorithm may have its own advantages and limitations in capturing the community structure of a network.

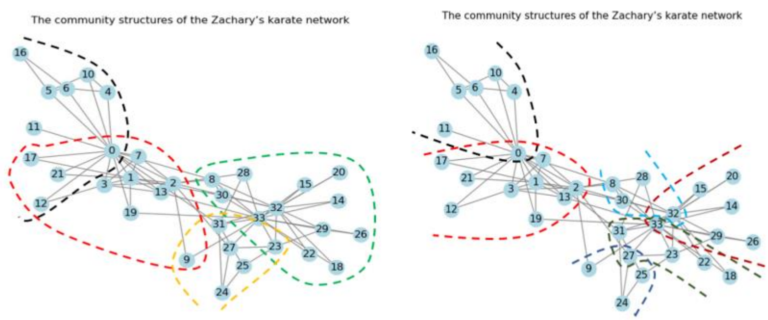

Figure 2
Zachary karate social network based CoCo-game and Clique.

Figure 2. Social structures of Zachary karate network. (a) Clique (b) Compare CoCo-game figure 2. Community selections for Zachary Karate network $(CoaSet^1, \alpha = 1, \beta = 1)$ [13].

In this particular study, the authors [11] evaluated their results using three distinct metrics: Normalized mutual information (NMI), Fraction of properly classified vertices, and modularity. The trial results showed that their approach outperformed other methods accomplished of detecting overlapping community' such as Clique and CoCo-Game [11].

5.2 *Normalized Mutual Information (NMI)*

The authors [11] adopted an extended version of NMI, which is a amount of similarity rented from information theory ideas. The lengthy version supports overlapping communities. NMI is intended using the formula:

$$N(X|Y) = 1 - (1/2)(H(X|Y)_{norm} + H(Y|X)_{norm}).$$

This Variable ranges from [0,1] and equals 1 once the two dividers are exactly coextensive. To calculate this metric, the authors initial calculate the following equation:

$$H(X|Y)_{norm} = (1/|\varsigma|) \sum (H(X_k|Y)_{norm}),$$

where:

$$H(X_k|Y)_{norm} = H(X_k|Y)/H(X_k)$$
$$H(X_k|Y)_{norm} = min(H(X_k|Y_l))$$
$$H(X_k|Yl) = H(X_k, Y_l) - H(Y_l)$$

In these equations, X_k represents the k-th component of X, $H(X)$ and $H(Y)$ are entropies of random variables X and Y, and $H(X_k|Y)$ is the provisional entropy of X_k by admiration to all the components of Y. Lastly, $H(X|Y)$ standard is the normalized provisional entropy of X with admiration to Y.

5.3 *Analysis and results in the NMI graph*

Networks based on the x-axis represent a subset of vertices that belong to multiple communities.

We generate a sequence of Benchmark networks by means of the Lanchichineti and Fortunato [1] method. The parameters rummage-sale are registered as follows: normal degree, maximum degree, number of Associations overlapping vertexs, and fraction of overlapping nodes amid 0 and 0.5.

In the results analysis, the writers evaluated the presentation of their method using the Normalized Mutual Information (NMI) measure on different datasets. Specifically, they focused on synthetic LFR datasets and performed 100 runs on each dataset to calculate randomness.

The results obtained from graphs with 1000 and 10000 nodes are presented in Figure 3, respectively. The y axis of these figures shows the NMI values, while the x axis demonstrations the fraction of overlapping node in the benchmark graphs, which range from 0 to 0.5. These fractions show the proportion of nodes that belong to several communities simultaneously. These figures also contain information about the Minimum ($s_{\min}$) and Maximum ($s_{\max}$) sizes.

We provided parameter locations used for these evaluations, including values such as $\tau_1 = 2$, $\tau_2 = 1$, $k_{\text{avg}} = 20$, $k_{\max} = 50$, $o_m = 2$, $\mu = 0.1$, and $\mu = 0.3$. The mixing parameter, μ, represents the amount of intersecting edges in the graph.

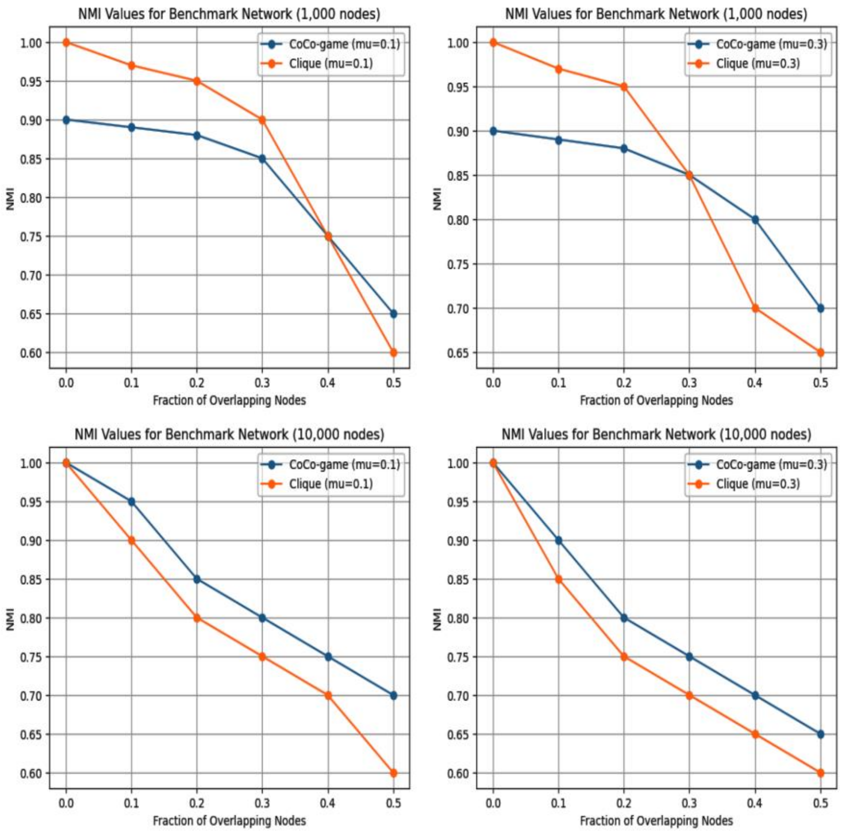

Figure 3
NMI values between identified community structures

Figure 3 shows the NMI values between the community constructions detection by CoCo-game and the Benchmark community constructions under dissimilar settings. The x-axis signifies the value of μ. The Benchmark networks used to generate Figure 3(a) consist of 1000 nodes, while the networks in Figure 3(b) consist of 10,000 nodes. The community size in Fig. 3(a) ranges from $minc = 10$ to $maxc = 50$, and the communities size in Fig. 3(b) ranges from $minc = 20$ to $maxc = 100$. Mixing limit $\mu = 0.1$ for Fig. 3(a), (b) and $\mu = 0.3$ for Fig. 3(b). Additional parameters are $\tau_1 = 2$, $\tau_2 = 1$, $k = 20$, $maxk = 50$, then $om = 2$.

Authors applied their method to Erdös-Rèyni random graphs, although specific details about this evaluation are not provided in the excerpt.

Their results using NMI and the obtained NMI values are presented in Figures. They also specified the parameter settings used for the evaluations. This method showed promising performance on synthetic LFR datasets and its effectiveness was evaluated on unstructured community graphs.

From Figure 3, we can see that in 1000 nodes, Clique is much faster than CoCo-game in the first cases, and finally, CoCo-game finds associations better than Clique. We also see that at 10,000 nodes CoCo-game is much faster than Clique in all cases: Clique has the longest running times of 11 and 169 seconds for $N = 1000$ and $N = 10000$, correspondingly, while the shortest consecutively time is. CoCo-game is 185 and 203 seconds for $N = 1000$ and $N = 10000$, correspondingly The parameters and values of the Benchmark network are written in the table 1 below [14].

Table 1
Summary of Eight Synthetic Networks

Network	Nodes	D_{Avg}	D_{Max}	τ_1	τ_2	μ	C_{Min}	C_{Max}	O_{Nodes}
LFRa	1000	10	50	2	1	0.1	10	50	100
LFRb	1000	10	50	2	1	0.3	10	50	100
LFRc	10000	20	100	2	1	0.1	20	100	500
LFRd	10000	20	100	2	1	0.3	20	100	500

Summary and Conclusion

Approach of allowing overlapping communities and utilizing simple utility functions has shown effectiveness in learning overlapping communities in both Benchmark graphs and practical networks. The usage of native operations in the algorithm has resulted in a fast running time and suitability for a similar framework.

There are still exciting exposed problems to explore within this outline. One interesting way is the search for more suitable gain and loss function. Although the ones future in we paper were simple and real, there is room for improvement by gaining a profounder sympathetic of the community formation process in practical networks.

This paper have developed a new approach that uses cooperative game theory for improved accuracy in identifying overlapping communities. Since each agent can choose manifold coalitions, overlapping community container be logically recognized. The empirical results of our method demonstrate its characteristics, showing that the combined usage of cooperative games is effective and efficient in detecting overlapping communities.

Overall, our approach contributes to the field of community detection by incorporating game theory principles, allowing for a more nuanced understanding of complex network structures. The ability to identify overlapping communities can provide valuable visions into the underlying organization and dynamics of social networks.

References

[1] A. Lancichinetti, S. Fortunato and J. Kertesz, "Detecting the overlapping and hierarchical community structure in complex networks", New Journal of Physics 11 122-128, (2009).

[2] C.H. Chen, "Applying game theory in Newsvendor's supply chain model", *Journal of Information and Optimization Sciences*, 42(6), 1367-1382, (2021).

[3] K.D. Gupta, A.Q. Ansari, P. Saxena and R. Dwivedi, "Optimizing channel allocation in WiMAX MIMO model using fuzzy game theory" *Journal of Information and Optimization Sciences*, 41(7), 1633-1644 (2020).

[4] C. Alós-Ferrer and A.B. Ania, "Ltextocal equilibria in economic games", *Economics Letters*, 70(2), 165-173 (2001).

[5] J. Chen and B. Yuan, "Detecting functional modules in the yeast protein–protein interaction network", *Bioinformatics*, 22(18), 2283-2290 (2006).

[6] P. O. Fjällström, "Algorithms for graph partitioning: A survey", Linköping University Electronic Press, (1998).

[7] M.E. Newman, "Who is the best connected scientist? A study of scientific coauthorship networks", *Complex networks* (pp. 337-370). Berlin, Heidelberg: Springer Berlin Heidelberg (2004).

[8] M.E. Newman, "Modularity and community structure in networks", *Proceedings of the national academy of sciences*, 103(23), 8577-8582 (2006).

[9] S. Fortunato and M. Barthelemy, "Resolution limit in community detection", *Proceedings of the national academy of sciences*, 104(1), 36-41 (2007).

[10] A. Baudin, M. Danisch, S. Kirgizov, C. Magnien and M. Ghanem, "Clique percolation method: memory efficient almost exact communities", International Conference on Advanced Data Mining and Applications (pp. 113-127). Cham: Springer International Publishing (2022, January).

[11] W. Chen, Z. Liu, X. Sun and Y. Wang, "A game-theoretic framework to identify overlapping communities in social networks", *Data Mining and Knowledge Discovery*, 21, 224-240 (2010).

[12] L. Zhou, P. Yang, K. Lü, L. Wang and H. Chen, "A fast approach for detecting overlapping communities in social networks based on game theory", *British International Conference on Databases* (pp. 62-73). Springer, Cham. (2015).

[13] G.L. Zacharias, J.E. Macmillan and S.B. Van Hemel, "Behavioral modeling and simulation: From individuals to societies", *National Academies Press* (2008).

[14] L. Zhou, C. Cheng, K. Lü and H. Chen, "Using coalitional games to detect communities in social networks", *International Conference on Web-Age Information Management* (pp. 326-331), Springer, Berlin, Heidelberg (2013).

Received July, 2024