An Approach for Overlapping and Hierarchical Community Detection in Social Networks Based on Coalition Formation Game Theory

Lihua Zhou[1], Kevin Lü[2*], Peizhong Yang[1], Lizheng Wang[1], Bing Kong[1]

[1]Department of Computer Science, Yunnan University, Kunming 650091, China
[2]Brunel University, Uxbridge, UB8 3PH, UK
[*]Corresponding Author: address: Room 104, ESGW Building, Brunel University Uxbeidge UB8 3PH, UK
email: kevin.lu@brunel.ac.uk
phone number 0044(1895)265254
email addresses: lhzhou@ynu.edu.cn (Lihua Zhou), pzyang@ynu.edu.cn (Peizhong Yang), chenchao@ynu.edu.cn (Chao Chen), zdzhang@ynu.edu.cn (Zidong Zhang).

Abstract: With greater availability of data and increasing interaction activities taking place on social media, to detect overlapping and hierarchical communities has become an important issue and one that is essential to social media analysis. In this paper, we propose a coalition formation game theory-based approach to identify overlapping and hierarchical communities. We model community detection as a coalition formation game in which individuals in a social network are modelled as rational players aiming to improve the group's utilities by cooperating with other players to form coalitions. Each player is allowed to join multiple coalitions, and those coalitions with fewer players can merge into a larger coalition as long as the merge operation is beneficial to the utilities of the merged coalitions, thus overlapping and hierarchical communities can be revealed simultaneously. The utility function of each coalition is defined as the combination of a gain function and a cost function. The gain function measures the degree of interactions amongst the players inside a coalition, while the cost function instead represents the degree of the interactions between the players of the coalition and the rest of the network. As game theory provides a formal analytical framework with a set of mathematical tools to study the complex interactions among rational players, to apply game theory for detecting communities helps to identify communities more rationally. Some desirable properties of the utility function, such as the non-resolution limit and the non-scaling behaviour, have been examined theoretically. To solve the issue of pre-setting the number and size for communities and to improve the efficiency of the detection process, we have developed a *greedy agglomerative manner* to identify communities. Extensive experiments have been conducted on synthetic and real networks to evaluate the effectiveness and efficiency of the proposed approach.

Keywords: Social network; community detection; coalition formation game theory

1 Introduction

With social networks gaining in popularity, social network analysis has become an

important research issue, with a significant impact on society (Fortunato 2010; Li *et al.* 2014). One major and fundamental topic in social network analysis is community detection, i.e. to identify groups of vertices in a network such that the vertices within a group are much more connected to each other than to the rest of the network (Newman and Girvan 2004; Fortunato 2010). Because individuals belonging to the same community are more likely to have common features, such as social functions, interests on some topics, viewpoints, etc. (Zhao et al. 2012), the identified communities can be used in the improvement of services (Krishnamurthy and Wang 2000), knowledge sharing (Liu *et al*. 2010), collaborative recommendation (Yuan et al. 2010), information spreading (Wu *et al*. 2004), structure visualizing (Wu and Li 2011), and other applications. In recent years, community detection has received a great deal of attention as it has significance relating to online influence analysis, online marketing and ebusiness (Bagrow 2012; Papadopoulous et al. 2012; Li *et al*. 2014b; Francesco and Clara 2014; Zhou and Lü 2014).

However, community detection is not a straightforward task, because in real networks communities can be overlapped or hierarchical, and these features often occur simultaneously. The overlap of communities implies that vertices simultaneously belong to more than one group, for instance, people belong to different social groups, depending on their activities, interests, etc. (Palla *et al*. 2005). This breaks the assumption that a community should have more internal than external connections (because highly overlapping communities can have many more external than internal connections), and demands a method that is able to detect either overlapping or non-overlapping communities (Lancichinetti et al. 2009). The hierarchical form of communities implies that the communities are recursively grouped into a hierarchical structure, i.e. small communities can form larger ones, which in turn can group more communities together to form even larger ones, etc. In the presence of hierarchies, the concept of community structure becomes richer, and demands a method that is able to detect communities at different levels, not just within a single level (Lancichinetti et al. 2009). Another two essential challenges in community detection are the efficiency of algorithms and the prior knowledge on the

number and size of communities, because the presence of many vertices and links in a large network results in heavy computation, and the number and size of communities are usually unknown beforehand. At present, these issues have not been solved satisfactorily. In existing community detection algorithms, some require a priori knowledge on the number and size of communities before performing the task of detecting communities, some are not able to detect overlapping and hierarchical communities, and some are not applicable to large-scale networks due to the low efficiency.

Motivated by the need for developing an algorithm that can detect both overlapping and hierarchical communities without prior knowledge on the number and size of communities in large-scale networks, we develop an approach by applying cooperative game theory (Zlotkin and Rosenschein 1994) to detect communities in this study. Cooperative game theory (Zlotkin and Rosenschein 1994) studies the cooperative behaviours of groups of rational players, where players cooperate with each other for improving the group's utility, such a group of players is called a coalition. One class of cooperative games is coalition formation games (Saad *et al*. 2009), whose main objective is to analyse the formation of coalitional structures through players' interaction. Coalition formation games are generally not superadditive due to the presence of costs that reduce the gains from forming the coalition. In social network environments, the behaviours of individuals are not independent (Zacharias et al. 2008), and joining a community provides one with tremendous benefits, such as members feeling rewarded in some ways for their participation in the community, and gaining honour and status for being members (Sarason 1974). In which case, every individual has an incentive to join communities; however, in real-world cases not only does each individual receive benefit(s) from the communities it belongs to, but the individual must also pay a certain price to maintain its membership within these communities (Chen et al. 2010). These characteristics make coalition formation game theory applicable to community detection.

In this study, we first model the process of community detection as a coalition formation game, in which individuals in a social network are modelled as rational

players aiming to achieve the maximal group's utility by cooperating with other players to form coalitions. A coalition is a subset of players. Each player is allowed to join multiple coalitions, which reflects the concept of "overlapping communities". Meanwhile, coalitions with fewer players can merge into a larger coalition as long as such merge operations could improve the utilities of the coalitions. This process reveals, in fact, the hierarchical structure of communities. A coalition is regarded as a *stable community* if it cannot further improve its utility by merging with other coalitions. If no coalition can further improve its utility by merging with other coalitions, the game achieves an *equilibrium state of coalitions*, and the configuration of communities at this state is called *the stable community structure*.

Next, we introduce the utility function for each coalition, which is the combination of a gain function and a cost function. The gain function measures the degree of the interaction amongst the players inside a coalition, while the cost function represents the degree of the interaction between the players of that coalition and the rest of the network. Based on the defined utility function, two coalitions without any link between them cannot improve their utilities by merging into a larger coalition, thus whether a coalition is merged with others can be decided by looking only at its neighbours (coalitions that have at least one link between them), rather than necessitating the performance of an exhaustive search over the entire network. This can speed up the computation considerably.

Then, we develop a *greedy agglomerative manner* to identify communities, which starts from the vertices as separate coalitions (singletons); coalitions are iteratively merged to improve the group's utilities until no further merging of coalitions is needed. This *greedy agglomerative manner* does not require a priori knowledge on the number and size of the communities, and it matches the real-world scenario, in which communities are formed gradually from bottom to top.

Finally, we conduct extensive experiments on different networks to assess the performance of our approach. Meanwhile, we also compare our results with other related studies. The experimental results show that our algorithm is effective and efficient in identifying overlapping and hierarchical communities.

The main contributions of this study can be summarized as follows:

- The coalition formation game theory is applied to address the community detection problem. This approach considers community formation as the result of the group behaviours of rational players who cooperate with each other to form coalitions for achieving and improving a group's utilities.
- A utility function for modelling the benefit and cost of each coalition is introduced, and the properties of the utility function, such as the non-resolution limit and the non-scaling behaviour, have been examined theoretically.
- An algorithm based on the greedy agglomerative manner is proposed to identify communities. The proposed algorithm does not require a priori knowledge on the number and size of communities, and it can detect the overlapping and hierarchical communities simultaneously.
- Extensive experiments on synthetic and real networks have been conducted to evaluate the effectiveness and efficiency of the proposed approach.

The rest of this paper is organized as follows: Section 2 introduces related work; Section 3 presents a coalition formation game theory-based framework for community detection; Section 4 provides a community detection algorithm that uses the greedy *agglomerative manner* to identify communities. The experimental results on the synthetic and real networks are presented in Section 5, and Section 6 concludes this paper.

2 Related work

A well-known method for detecting non-overlapping and non-hierarchical communities is the use of modularity-based methods (Newman and Girvan 2004), which is based on the idea that a random graph is not expected to have a cluster structure, so the possible existence of clusters is revealed by the comparison between the actual density of edges in a subgraph and the density one would expect to have in the subgraph, if the vertices of the graph were attached regardless of community structure (Fortunato 2010). However, modularity-based methods implicitly assume that communities do not intersect with one another, which is usually not the case for

real-world communities (Chen *et al*. 2010). Fortunato and Barthélemy (2006) found that modularity optimization may fail to identify communities smaller than a scale which depends on the total number of links of the network and on the degree of interconnectedness of the communities, even in cases where communities are unambiguously defined. Brandes et al. (2008) also identified counterintuitive properties of modularity, such as non-locality and sensitivity to satellites.

To detect overlapping communities, Palla et al. (2005) defined a *k*-clique-community as the union of all *k*-cliques that can be reached from each other through a series of adjacent *k*-cliques. But their algorithm requires the size of clique as an input, which is usually unknown in practical applications. Ahn et al. (2010) considered a community to be a set of closely interrelated links instead of a set of vertices with many links between them. Comparing with vertex communities, link communities incorporate overlap while revealing hierarchical organizations. In general, the number of links is greater than the number of vertices, so link-based approaches may suffer from greater computation cost than a vertex-based approach in the process of detecting communities. Ball et al. (2011) proposed a probabilistic model of link communities to detect communities, either overlapping or not, and used a fast, closed-form expectation-–maximization algorithm to analyse networks of millions of vertices in reasonable running times. However, the approach of Ball et al. offers no criterion for determining the number of communities in a network. Galbrun et al. (2014) adapted efficient approximation algorithms to find *k* communities of labelled graphs so that the total edge density over all *k* communities is maximized and each community is succinctly described by a set of labels. To detect overlapping communities in semantic social networks, Xin et al. (2015a; 2015b) proposed methods, in which it is not necessary to pre-set the number of communities; Wu et al. (2015) provided an algorithm to solve the query biased densest connected subgraph (QDC) problem, where overlapping local communities and multiple disjointed local communities can also be found.

Hierarchical clustering algorithms are usually used to reveal hierarchical communities of graphs. Sales-Pardo et al. (2007) proposed a top-down approach to

identify the hierarchical communities of a graph from the similarity matrix of vertices, but the algorithm is not fast enough (Fortunato 2010). Clauset et al. (2008) used a dendrogram and a set of probabilities associated to the internal vertices of the dendrogram to describe the hierarchical organization of a graph. This method is capable of describing closely the graph properties, but it is impossible to rank community structures according to their relevance. Shen et al. (2009) handled the set of maximal cliques and adopted an agglomerative framework to detect both the overlapping and hierarchical properties of a complex community structure, but the efficiency of their algorithm requires improvement. Blondel et al. (2008) proposed a rapid method to unfold hierarchical community structures of large networks based on modularity optimization, but this method cannot detect overlapping communities.

Game theories have been used to solve community detection problems. For example, Chen et al. (2010) addressed the community detection problem by a non-cooperative game theory-based framework (Nash 1951) that considers community formation as the result of individual agents' rational behaviours and a community structure as an equilibrium of a game. This framework can identify overlapping communities because each agent is allowed to select multiple communities, but hierarchies between communities cannot be revealed. Alvari et al. (2011) considered the formation of communities in social networks as an iterative game in a multiagent environment, in which each vertex is regarded as an agent aiming to be in the communities with members such that they are structurally equivalent. Lung et al. (2012) formulated the community detection problem from a game theory point of view and solved this problem by using a crowding based differential evolution algorithm adapted for detecting Nash equilibria of non-cooperative games. Hajibagheri et al. (2012) used a framework based on an information diffusion model and Shapley Value concept to address the community detection problem. In Hajibagheri et al.'s framework, each vertex of the underlying graph is attributed to a rational agent aiming to maximize its Shapley value in the form of information it receives, and the Nash equilibrium of the game corresponds to the community structure of the graph.

In our previous studies (Zhou et al. 2013a; 2013b), we proposed two coalitional game models for community detection. But the coalitional game theories used in (Zhou et al. 2013a~~,~~ ; 2013b) are canonical coalitional games due to the characteristic functions defined in the models satisfying superadditive (Saad et al. 2009).

In our previous studies (Zhou et al. 2013a; 2013b), we proposed two coalitional game models for community detection. But both characteristic functions defined in these models satisfy superadditive, thus players are willing to form grand coalitions (the coalition of all players). In (Zhou et al. 2015a),

Based on those characteristic functions, players are willing to form grand coalitions (the coalition of all players). In Zhou et al. (2015a), we combine cooperative and non-cooperative game theory to detect communities, while we propose a coalition formation game theory-based approach to detecting communities in multi-relational social networks, where multi-relational communities are defined as the shared communities over multiple single-relational graphs (Zhou et al. 2015b). Although the cooperative game is used in ~~(~~Zhou et al. (2015a; 2015b), the forms of the utility function are different from the one designed in this paper.

3 A coalition formation game theory-based framework for community detection

One of the main characteristics that make a game a coalition formation game is the presence of a cost for forming coalitions. It makes coalition formation games generally not superadditive, which implies that forming a coalition brings gains to its members, but those gains are limited by a *cost* for forming the coalition, hence the grand coalition is seldom the optimal structure (Saad et al. 2009). In a coalition formation game, *network structure* and *cost* for cooperation play major roles.

In this paper, we propose a coalition formation game theory-based framework to identify overlapping and hierarchical communities, thus individuals of a network

choose to form community structures after a social network is formed. Individuals in a social network are modelled as rational players aiming to achieve and improve utilities of groups by cooperating with other players to form coalitions. Coalitions with fewer players can merge into a larger coalition as long as the merge operation can contribute to improve the utilities of the merged coalitions. The process of merging coalitions actually illustrates the process of forming the hierarchy communities. Meanwhile, each player is allowed to join multiple coalitions, which could capture and reflect the concept of "overlapping communities". A community structure of a network is a *collection* of coalitions, and the number of coalitions in a *collection* of coalitions is the number of communities with respect to the community structure. Due to the hierarchical form amongst communities, there are different community structures at different levels. Amongst them, *a stable community structure* is an *equilibrium state of coalitions,* in which no group of players has an interest in performing a merge operation any further.

The utility function for each coalition is defined as a combination (summation) of a gain function and a cost function. The gain function is based on a ratio of links inside a coalition over the total degree of vertices inside the same coalition, while the cost function is based on a ratio of the total degree of vertices inside a coalition over the total links in the network. The gain function measures the degree of the interaction amongst the players inside a coalition, while the cost function represents the degree of the interaction between the players of the coalition and the rest of the network. A coalition is regarded as a *stable community* if it cannot further improve its utility by merging with other coalitions.

For a given social network, the objective of detecting communities is to detect and identify the overlapping and hierarchical communities of the network. For this objective, we first present the notations, definitions and properties of the utility function.

3.1 Notations

Let $G = (V, E)$ be an undirected unweighted graph representing a social network

with $|V|$ vertices (individuals) and $|E|$ links (interactions). Let A be an adjacency matrix of G with $A_{xy}=1$ if $(x,y)\in E$ for any pair of vertices $x,y\in V$ and 0 otherwise, and let $d(x)$ be the degree of vertex x.

Let S denote a subset of V, which is called a coalition, meanwhile let $e(S)$, $d(S)$ and $v(S)$ be the number of links amongst vertices inside S, the total degree of vertices in S and the utility function of S, respectively. For any coalition $S_1, S_2 \subseteq V$, let $e(S_1, S_2)$ be the number of links connecting vertices of the coalition S_1 to the vertices of the coalition S_2. Let S_{ij} be a super-coalition of S_i in a merge operation of S_i. Furthermore, let Γ be a community structure (a *collection* of coalitions), i.e. $\Gamma=\{S_1, S_2, ..., S_k\}$, and let $v(\Gamma)$ be the total utility achieved in Γ.

Depending on the context, an element in V may either be called a player or a vertex, and a subset of V may either be called a group or a coalition or a community. Also, a *collection* of coalitions, a coalition structure and a community structure can be used interchangeably.

Definition 1. *Stable community*. A coalition S is regarded as a *stable community* if S cannot further improve its utility by merging with other coalitions, i.e. $\forall S' \neq S$, $v(S+S') < v(S)$ and $\forall S' \subseteq S, v(S) > v(S')$. Specially, S is called the *grand coalition* (Saad et al. 2009) if $S=V$, i.e., the coalition of *all* the players, while S is called a *trivial coalition* if S solely consists of a single vertex, i.e. $S=\{x\}, x\in V$.

Definition 2. *Utility increment of a coalition*. The *utility increment* of coalition S_i with respect to S_{ij} is defined by $\Delta v(S_i, S_{ij}) = v(S_{ij}) - v(S_i)$.

Definition 3. *Stable community structure.* A *collection* of coalitions $\Gamma=\{S_1, S_2, ..., S_k\}$ is a *stable community structure* if

$\forall S_i \in \Gamma, \max(\max_{S_{ij}} \Delta v(S_i, S_{ij}), 0) = 0$ holds.

A stable community structure is a form of *equilibrium state of coalitions*, in which no group of players has an interest in performing a merge operation any further. When a game enters the *equilibrium state*, the number of coalitions in $\Gamma = \{S_1, S_2, ..., S_k\}$ is the number of communities, and the number of the vertices in S_k is the size of the community S_k.

Definition 4. *Total Utility*. Let $\Gamma = \{S_1, S_2, ..., S_k\}$, then the total utility $v(\Gamma)$ with respect to Γ is defined by the following equation (Equation 1):

$$v(\Gamma) = \sum_{S_i \in \Gamma} v(S_i) \qquad (1).$$

Theorem 1. The *collection* of coalitions $\Gamma = \{S_1, S_2, ..., S_k\}$ is a *stable community structure* if all $S_1, S_2, ..., S_k$ are *stable communities.*

Proof. From $S_1, S_2, ..., S_k$ are *stable communities*, we have, for any $S_i \in \Gamma$, $\forall S_{ij}, \Delta v(S_i, S_{ij}) < 0$, then $\forall S_i \in \Gamma, \max(\max_{S_{ij}} \Delta v(S_i, S_{ij}), 0) = 0$ holds. □

Theorem 2. A *stable community structure* $\Gamma = \{S_1, S_2, ..., S_k\}$ maximizes the total utility $v(\Gamma)$.

Proof. $\Gamma = \{S_1, S_2, ..., S_k\}$ is a stable community structure $\Rightarrow S_1, S_2, ..., S_k$ are *stable communities* $\Rightarrow$ $\forall S_i \in \Gamma, \max(\max_{S_{ij}} \Delta v(S_i, S_{ij}), 0) = 0$ $\Rightarrow$ $v(\Gamma)$ is maximal.

3.2 Utility function

Definition 5. *Utility function*. Let S be a coalition of $G = (V, E)$, then the utility function $v(S)$ of S is defined by the following equation (Equation 2):

$$v(S) = \frac{2e(S)}{d(S)} - \alpha\left(\frac{d(S)}{2\beta |E|}\right)^2 \qquad (2)$$

The first term and the second term in Equation (2) are called the gain function and the cost function of S, respectively. The gain function is the ratio of links inside S over the total degree of the vertices in S; the cost function instead represents the

ratio of the total degree in that coalition over the total degree in the network. The larger gain function value means that there are more interactions amongst the players inside S, and the larger cost function value means that there is greater interaction between the players of the coalition S and the rest of the network. Equation (1) means that forming a coalition brings gains to its members, but the gains are limited by a *cost* for forming the coalition.

α is a scale factor used to adjust the cost of coalition S, $\alpha \in [0,1]$. $\alpha = 0$ means no cost for forming coalitions, i.e. forming a coalition is always beneficial. In this case, the utility function $v(S)$ is superadditive due to it being defined only by the gain function. Thus, $v(V) = 1$, $v(\{x\}) = 0, x \in V$. That means that the utility function of the *grand coalition* has maximal value, while the utility function of a singleton coalition has a value of 0. When $\alpha = 1$, the costs for forming coalitions are maximal, Thus, $v(V) = 0$ and $v(\{x\}) = -\left(\frac{d(x)}{2\beta |E|}\right)^2, x \in V$. So, the grand coalition and the collection of trivial coalitions are seldom the optimal structures. Moreover, the smaller $d(x)$ is, the greater $v(\{x\})$ will be. Which means that vertices with small degrees are apt to be far more interested in collaborating with other vertices to improve their utilities.

β is another parameter used to adjust the context of the coalition S, $\beta \in (0,1]$. $\beta = 1$ means that the context of coalitions is the whole network; $\beta < 1$ means that the context of coalitions is a local of the network. Complex networks normally include many vertices and links, so the *cost* of a coalition with fewer degrees may be neglected with respect to the whole network. By using β, the costs are localized.

Example 1. Figure 1 shows two simple social networks. Figure 1(a) is a network with a 4-clique and Figure 1(b) is a network with two 3-cliques. In Figure 1(a), if $\alpha = 1$, $\beta = 1$, $v(S) = \frac{2e(S)}{d(S)} - \left(\frac{d(S)}{2|E|}\right)^2$, $v(\{1,2\}) = \frac{2}{6} - \left(\frac{6}{12}\right)^2 = 0.08$,

$v(\{1,2,3\}) = \frac{6}{9} - \left(\frac{9}{12}\right)^2 = 0.12$, $v(\{1,2,3,4\}) = 0$. The 4-clique cannot be assessed correctly.

If $\alpha = \frac{1}{\sqrt{|E|}}$, $\beta = 1$, $v(\{1,2\}) = \frac{2}{6} - \frac{1}{\sqrt{6}}\left(\frac{6}{12}\right)^2 = 0.23$,

$v(\{1,2,3\}) = \frac{6}{9} - \frac{1}{\sqrt{6}}\left(\frac{9}{12}\right)^2 = 0.44$, $v(\{1,2,3,4\}) = 1 - \frac{1}{\sqrt{6}} = 0.59$. So, the 4-clique can be assessed correctly.

In Figure 1(b), if $\alpha = 0$, $\beta = 1$, $v(\{1,2,3,4,5,6\}) = 1$. The grand coalition has maximal utility, and two 3-cliques cannot be assessed correctly.

If $\alpha = \frac{1}{\sqrt{|E|}}$, $\beta = 1$, $v(\{2,3\}) = \frac{2}{4} - \frac{1}{\sqrt{7}}\left(\frac{4}{14}\right)^2 = 0.47$,

$v(\{1,2,3\}) = \frac{6}{7} - \frac{1}{\sqrt{7}}\left(\frac{7}{14}\right)^2 = 0.76$, $v(\{1,2,3,4\}) = \frac{8}{10} - \frac{1}{\sqrt{7}}\left(\frac{8}{14}\right)^2 = 0.68$. The two 3-cliques can be assessed correctly. The inequality $v(\{4\}) < v(\{1,2,3,4\}) < v(\{1,2,3\})$ means that players 1, 2 and 3 do not collaborate with player 4 although player 4 intends to join the group $\{1,2,3\}$.

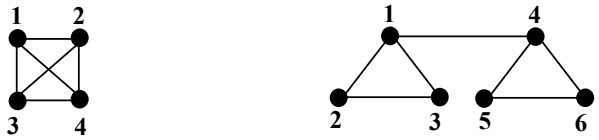

Figure 1. Two simple social networks. (a). A 4-clique network; (b). A network with two 3-cliques

3.3 Properties of the utility function

The utility function $v(S)$ of coalition S defined in Equation (1) has the following properties.

Property 1. Isolated vertices have no impact on $v(S)$.

This directly follows from the fact that $v(S)$ depends on links and degrees, thus, an isolated vertex does not contribute, regardless of its association to a group. Therefore, all vertices are assumed to be of a degree greater than zero in this study, i.e., isolated vertices are excluded from further consideration.

Property 2. The lower and upper bounds of $v(S)$ satisfy: $-\alpha\left(\frac{\max(d(x))}{2\beta|E|}\right)^2 \le v(S) \le 1, x \in V$.

Proof: When $\alpha = 0$, $v(S)$ is superadditive, $v(S)_{\max} = v(V) = 1$; when S is a singleton coalition, $e(S) = 0$, hence $v(S)_{\min} = -\alpha\left(\frac{\max(d(x))}{2\beta|E|}\right)^2, x \in V$.

Property 3. If S is a clique, $v(S) > v(S - \{x\})$, $x \in S$; $v(S + \{y\}) > v(S)$ if $d(y) < 3$; $v(S + \{y\}) < v(S)$ if $d(y) \ge 3$.

This property means that the utility of a clique is greater than the utility of each subset of the clique itself; the utility of a coalition composed of a clique and a vertex (that is not a member of the clique but is connected to a vertex of the clique) with degree 1 or 2 is greater than the utility of the clique, but the utility of the clique is greater than the utility of a coalition composed of the clique and a vertex (that is not a member of the clique but is connected to a vertex of the clique) with degree of at least 3.

Proof: Let S be a p-clique ($p \ge 3$, because a 3-clique is a trivial clique), vertex $x \in S$, $d(x) = p$, $y \notin S$. The relationships between x and y are shown in Figure 2. For simplicity, let $\alpha = 0$, then,

$$v(S - \{x\}) = \frac{2 \times \frac{(p-1)(p-2)}{2}}{(p-1)(p-1) + p - 1} = \frac{(p-1)(p-2)}{(p-1)^2 + p - 1} = \frac{p-2}{p},$$

$$v(S) = \frac{2 \times \frac{p(p-1)}{2}}{p(p-1) + p} = \frac{p(p-1)}{p^2} = \frac{p-1}{p} > v(S - \{x\});$$

$$v(S+\{y\}) = \frac{p(p-1)+2}{p(p-1)+p+d(y)} = \frac{p^2-p+2}{p^2+d(y)},$$

$$v(S+\{y\}) - v(S) = \frac{p^2-p+2}{p^2+d(y)} - \frac{p-1}{p} = \frac{p(2-d(y))+d(y)}{p(p^2+d(y))}.$$

If $d(y) \geq 3$, then, $v(S+\{y\}) - v(S) \leq 0$; if $d(y) < 3$, $v(S+\{y\}) - v(S) > 0$. □

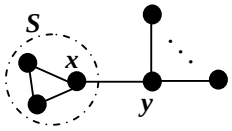

Figure 2. A network with a *p*-clique *S*

Property 4. $v(S)$ is not limited by the resolution limit of Newman and Girvan's *modularity* (it is defined by $Q(\Gamma) = \sum_{S \in \Gamma} \left[\frac{e(S)}{|E|} - \left(\frac{e(S) + \sum_{S' \in \Gamma} e(S,S')}{2|E|} \right)^2 \right]$ (Newman and Girvan 2004).

The resolution limit of Newman and Girvan's *modularity* means that modularity optimization may fail to identify communities smaller than a scale which depends on the total number of links of the network and on the degree of interconnectedness of the communities, even in cases where communities are unambiguously defined. For example, Figure 3 shows a network with four pairwise identical cliques (S_3, S_4 are two *m*-cliques and S_1, S_2 are two *p*-cliques, $p < m$); if *m* is large enough with respect to *p* (e.g. $m = 20$, $p = 5$), modularity optimization merges the two smallest groups into one (shown with a dotted line) (Fortunato and Barthélemy 2006).

Proof: (1) Because $d(y) \geq 3$, $v(S_1 + \{y\}) < v(S_1)$ (Property 3), $v(S)$ can evaluate S_1 at lower level.

(2) Because the utility function defined in Equation (1) is the combination of a gain function and a cost function, S_1 does not merge with S_2 at higher level if α and β are suitable.

Therefore, $v(S)$ is not limited by the resolution limit of Newman and Girvan's

modularity. □

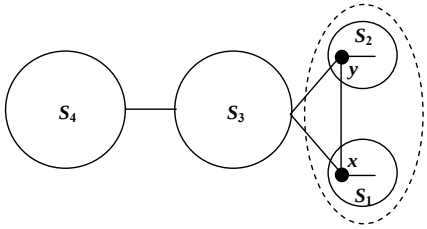

Figure 3. A network with four pairwise identical cliques (S_3, S_4 are two m-cliques, S_1, S_2 are two p-cliques, $p < m$).

Property 5. $v(S)$ is not limited by the non-locality of Newman and Girvan's *modularity* if S is a clique.

The non-locality of Newman and Girvan's *modularity* means that the memberships of some vertices may be changed by adding an additional vertex, although locally their neighbourhood structure has not changed. For example, based on Newman and Girvan's *modularity*, the vertices of the network shown in Figure 4 (a) are clustered into two groups ({1,2,3,4}, {5,6}, represented by different shading), but the vertices are clustered into three groups ({1,2}, {3,7}, {4,5,6} in which the membership of vertex 4 is shifted after additional vertex 7 is connected to vertex 3 (shown in Figure 4 (b)) (Brandes et al. 2008).

Proof: Because S is a clique and $d(y)=1$, thus $v(S+\{y\})>v(S)$ (Property 3), therefore, y is just joined to S rather than changing the membership of the vertex of S.□

For example, ({1,2,3,4}, {5,6}) is a stable community structure with respect to the vertices in Figure 4 (a), and ({1,2,3,4,7}, {5,6}) is a stable community structure after vertex 7 is connected to vertex 3 (shown in Figure 4 (c)). This shows that $v(S)$ does not suffer from the non-locality.

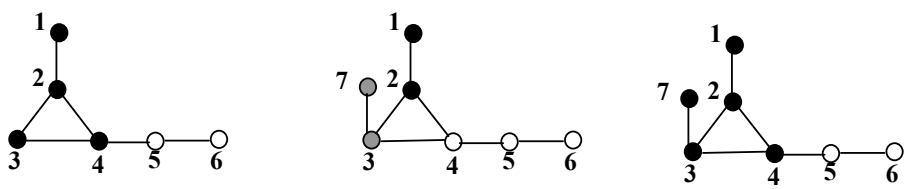

Figure 4. Non-locality behaviour. (a) The original community structure (based on Newman and Girvan's *modularity*); (b) The community structure after vertex 7 is added (based on Newman and Girvan's *modularity*); (c) The community structure after vertex 7 is added (based on $v(S)$)

Property 6. $v(S)$ is not sensitive to satellites.

The sensitivity to satellites means that for a clique with leaves, a network of $2p$ vertices that consists of a p-clique and p-leaf vertices of 1 degree, such that each vertex of the clique is connected to exactly one leaf vertex, the optimal community structure based on Newman and Girvan's *modularity* is composed of p groups, in which each group consists of a connected pair of a leaf and a clique vertex. Figure 5(a) shows an example (Brandes et al. 2008).

Proof: According to Property 3, p-leaf vertices form the same coalition with vertices of the p-clique, i.e. the community formed by all vertices in the graph is a stable community. □

Thus, the stable community structure corresponding to the network of Figure 5(a) is {1, 2, 3, 4, 5, 6} (shown in Figure 5(b)).

Figure 5. No sensitivity to satellites: (a) The community structure (based on Newman and Girvan's *modularity*); (b) The community structure (based on $v(S)$)

Property 7. $v(S)$ does not have the scaling behaviour of Newman and Girvan's *modularity*.

The scaling behaviour of Newman and Girvan's *modularity* means that by simply duplicating a network of $2p$ vertices that consists of a p-clique and p-leaf vertices of 1 degree such that each vertex of the clique is connected to exactly one leaf vertex, the optimal clustering is altered completely. For example, duplicating the network presented in Figure 5(a), three clusters in Figure 5(a) have been changed into two clusters, each of them being a network equivalent to the one in Figure 5(a)

(shown in Figure 6) (Brandes et al. 2008).

Proof: According to Property 6, the community formed by all vertices in a network, of 2*p* vertices that consists of a *p*-clique and *p*-leaf vertices of 1 degree such that each vertex of the clique is connected to exactly one leaf vertex, is a stable community formed by all vertices; in the same way, the community formed by all vertices in the duplicated graph is also a stable community. Therefore, $v(S)$ does not have the scaling behaviour of Newman and Girvan's *modularity*. □

For example, {1,2,3,4,5,6} is a stable community structure with respect to the vertices in Figure 5(a), and {{1,2,3,4,5,6}, {1',2',3',4',5',6'}} is a stable community structure after the network presented in Figure 5(a) is duplicated based on $v(S)$ (shown in Figure 6). From Figure 6, we can see that the interactions amongst vertices in each coalition form a connected subgraph.

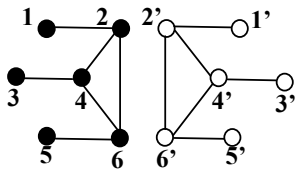

Figure 6. Without scaling behaviour

4 A community detection algorithm

In this study, we develop a *greedy agglomerative manner* to identify communities, the main idea of the *greedy agglomerative manner* is to start from the vertices as separate coalitions (singletons); coalitions that can result the highest utility increment are iteratively merged into a larger coalition to improve the group's utilities until no such merge operation can be performed. In this section, we first present the conditions of merging two coalitions, and then we give our *greedy agglomerative* algorithm, referred to as the *COFOGA* (*coalition formation game-based greedy agglomerative*) algorithm.

4.1 The conditions of merging two coalitions

Let S_1 and S_2 be two small coalitions with few players. S_1 and S_2 can merge into a larger coalition if and only if the following conditions are held.

Condition 1: $v(S_1+S_2)>v(S_1)$ & $v(S_1+S_2)>v(S_2)$. This condition means that the utilities of S_1 and S_2 have been improved through the merge operation. The unilateral meet of two inequalities shows that two coalitions fail to reach an agreement to cooperate, for example, the case of " $v(S_1+S_2)<v(S_1)$ but $v(S_1+S_2)>v(S_2)$" suggests that S_2 intends to cooperate with S_1 but S_1 may not agree.

Note that there can be several pairs i and j such that $v(S_i+S_j)$ is the maximum, meanwhile $v(S_i+S_j)>v(S_i)$ and $v(S_i+S_j)>v(S_j)$. In these cases the algorithm selects an arbitrary pair to merge.

Condition 1 ensures that a coalition formed by the merge operation has greater utility than that of its subsets.

Condition 2: $e(S_1,S_2)\neq 0$, i.e. S_1 does not cooperate with S_2 if $e(S_1,S_2)=0$.

Proof: $v(S_1)=\frac{2e(S_1)}{d(S_1)}-\alpha\left(\frac{d(S_1)}{2\beta|E|}\right)^2$, $v(S_2)=\frac{2e(S_2)}{d(S_2)}-\alpha\left(\frac{d(S_2)}{2\beta|E|}\right)^2$,

$$v(S_1+S_2)=\frac{2e(S_1)+2e(S_2)+2e(S_1,S_2)}{d(S_1)+d(S_2)}-\alpha\left(\frac{d(S_1)+d(S_2)}{2\beta|E|}\right)^2,$$

If $e(S_1,S_2)=0$, then

$$v(S_1+S_2)-v(S_1)=\frac{2e(S_2)d(S_1)-2e(S_1)d(S_2)}{[d(S_1)+d(S_2)]d(S_1)}-\alpha\frac{2d(S_1)d(S_2)+d^2(S_2)}{4\beta^2|E|^2},$$

$$v(S_1+S_2)-v(S_2)=\frac{2e(S_1)d(S_2)-2e(S_2)d(S_1)}{[d(S_1)+d(S_2)]d(S_1)}-\alpha\frac{2d(S_1)d(S_2)+d^2(S_1)}{4\beta^2|E|^2},$$

$v(S_1+S_2)-v(S_1)$ and $v(S_1+S_2)-v(S_2)$ cannot be greater than zero at the same time, because if $v(S_1+S_2)-v(S_1)>0$, then $e(S_2)d(S_1)>e(S_1)d(S_2)$ is inevitable, but under this condition, $v(S_1+S_2)-v(S_2)<0$. Therefore, if $e(S_1,S_2)=0$, S_1 does not

cooperate with S_2. This implies that two coalitions without a link between them cannot merge into a larger coalition. Based on this conclusion, whether a coalition is merged with others can be decided by looking only at its neighbours (coalitions that have links between them), without an exhaustive search over the entire network.

Condition 3: $e(S_1 + S_2) < \sqrt{2\beta |E|}$. The study of Fortunato and Barthélemy (2006) shows that a coalition S (with $e(S)$ internal links) found by modularity optimization may be a combination of two or more smaller communities if $e(S) < \sqrt{2|E|}$. Here, we use $e(S_1 + S_2) < \sqrt{2\beta |E|}$ as one of the conditions for merging two coalitions.

4.2 Description of the coalition formation game-based greedy agglomerative algorithm

The pseudo-code for the *greedy agglomerative algorithm* is given in the *COFOGA* algorithm.

COFOGA algorithm*:*

Input: A network $G(V, E)$

Output: The communities of the network

Variables:

k : The index of level in the hierarchical communities of the graph

$CoaSet^k$: The set of coalitions at k-th level

$CooSetMap$: The map of $CoaSet^k$, i.e. the copy of $CoaSet^k$

$CooSps$: A cooperative sponsor, i.e. a coalition with maximal utility in $CoaSet^k$

$CooCaSet$: The set of cooperative candidates, i.e. a cooperative candidate is a coalition in which there is at least one link between the coalition and $CooSps$

$CooCas^*$: A best cooperative candidate, i.e. such a coalition in $CooCaSet$ that the cooperation of the coalition with $CooSps$ can bring about the maximal increment of utility.

Steps:

1. $k = 0$

2. $CoaSet^k = \{\{1\},\{2\},...,\{V\}\}$

3. Do

4. $CoaSetMap = CoaSet^k$

5. $k = k + 1$

6. $CoaSet^k = \phi$

7. while $CoaSetMap \neq \phi$

8. $CooSps = \arg \max_{S \in CoaSetMap} v(S)$

9. $CoaSetMap = CoaSetMap - \{CooSps\}$

10. $CooCaSet(CooSps) = \{S \mid \exists A(x,y) \neq 0, x \in CooSps, y \in S, S \in CoaSet^{k-1}\}$

11. while $CooCaSet(CooSps) \neq \phi$

12. $CooCas^* = \arg \max_{CooCas \in CooCaSet(CooSps)} v(CooSps + CooCas)$

13. if $e(CooSps + CooCas^*) < \sqrt{2\beta \mid E \mid}$ &

$v(CooSps + CooCas^*) > v(CooSps)$ & $v(CooSps + CooCas^*) > v(CooCas^*)$

14. $CooSps = CooSps + CooCas^*$

15. $CoaSetMap = CoaSetMap - \{CooCas^*\}$

16. $CoaCaSet(CooSps) = CoaCaSet(CooSps) - \{CooCas^*\}$
$+ (CooCaSet(CooCas^*) - \{CooSps\})$

17. else $CoaCaSet(CooSps) = CoaCaSet(CooSps) - \{CooCas^*\}$

18. end if

19. end while

20. $CoaSet^k = CoaSet^k + \{CooSps\}$

21. end while

22. while $CoaSet^k \neq CoaSet^{k-1}$

23. for $i = 0$ to k

24. Output $CoaSet^k$

25. end for

Step 1~Step 2 are the initializations: each vertex forms a singleton coalition, and all singleton coalitions form $CoaSet^0$; the loop of Step 11~Step 19 creates a coalition for $CoaSet^{k+1}$, while the loop of Step 7~Step 21 creates all coalitions for $CoaSet^{k+1}$; the loop of Step 3~Step 22 reveals the hierarchical communities of the graph, and the loop of Step 23~Step 25 outputs community structures at different levels. For creating coalitions for $CoaSet^{k+1}$, Step 8 selects $CooSps$ from $CoaSetMap^k$, Step 9 deletes $CooSps$ from $CoaSetMap^k$, Step 10 selects cooperative candidates for $CooSps$ from $CoaSet^{k-1}$, and Step 12 finds the best cooperative candidate $CooCas^*$; if $CooSps$ and $CooCas^*$ meet conditions for merging, then merge operation can be carried out and $CooSps$ is replaced by the coalition formed by merging $CooSps$ and $CooCas^*$ in Step 14; Step 15 deletes $CooCas^*$ from $CoaSetMap^k$; Step 16 amends $CooCaSet(CooSps)$ by deleting $CooCas^*$ and adding cooperative candidates of $CooCas^*$ ($CooSps$ excepted). This process is repeated until no further merge operations can be performed.

$CooSps$ can only be selected from $CoaSetMap^k$, while cooperative candidates of $CooSps$ are selected from $CoaSet^{k-1}$. This strategy enables that $CooSps$ is not selected for multiple times and each vertex can join multiple coalitions.

$CoaSet^0$, $CoaSet^1$, …, $CoaSet^k$ reveal the hierarchical communities of the graph. The value of k represents the number of levels, $CoaSet^k$ implies the community structure at k-th level, and the number of coalitions in $CoaSet^k$ means the number of communities at k-th level. Because the agglomerative process is carried out automatically, the number and size of the communities are obtained automatically rather than specified in advance.

The time complexity of the *COFOGA* algorithm is $O(|V|\log|V|)$ at worst case. Note that, $|V|-1$ iterations are an upper bound and the algorithm will terminate as soon as a pair of coalitions would not be merged. It is possible that the algorithm ends before the *grand coalition* forms.

5 Experiments and results

In this section, extensive experiments have been undertaken for assessment:

(1) the effectiveness of the *COFOGA* algorithm in real networks. Two well-known real networks are used to examine if the *COFOGA* algorithm can correctly identify the overlapping communities and the hierarchical structure of communities;

(2) the effectiveness of the *COFOGA* algorithm in benchmark networks. These benchmark networks are produced under different assumptions containing different community information, such as different vertices, different connections, or different overlapping vertices. Because the real community information is known, we use the normalized mutual information (NMI) as the quantitative evaluation metric. These benchmark networks are also used to assess the efficiency of the *COFOGA* algorithm under different conditions.

(3) whether the *COFOGA* algorithm is limited by the resolution limit. To this end, we create two synthetic networks made of cliques (complete graphs). We want to find out whether the *COFOGA* algorithm can integrate the smaller cliques into the larger group.

(4) comparisons have been made with other algorithms in which non-cooperative game theory has been applied.

5.1 Assessing the effectiveness of the *COFOGA* algorithm in real networks

We first apply the *COFOGA* algorithm in the *Zachary's Karate Network* (Zachary 1977) and the *Lusseau's Dolphin Network* (Lusseau 2003), two well-known real networks used to test community detection algorithms. The *Zachary's Karate Network* consists of 34 vertices and 79 links, and the *Lusseau's Dolphin Network* consists of 62 vertices and 159 links. Figure 7 (a) and (b) presents the communities

detected by *LocalEquilibrium* (an algorithm that applies non-cooperative game theory) (Chen et al. 2010) and coalitions detected by the *COFOGA* at the first level (i.e. $k = 1$) in the *Zachary's Karate Network*, and Figure 8 (a) and (b) presents the communities detected by *LocalEquilibrium* (Chen et al. 2010) and coalitions detected by the *COFOGA* at the first level (i.e. $k = 1$) in the *Lusseau's Dolphin Network*. Similar to the community structures detected by *LocalEquilibrium,* the community structures detected by *COFOGA* are refinements of the community structures discovered in Newman and Girvan's work (2004), in which two networks are divided into two components (the two components in the *Zachary's Karate Network* correspond to the upper overlapping communities and the two lower communities in Figure 7(b), and the two components in the *Lusseau's Dolphin Network* correspond to the three upper overlapping communities and the three lower communities in Figure 8(b)). However, the number of communities and the overlapping vertices discovered by the *COFOGA* are different from those discovered by *LocalEquilibrium*, for example, in the *Zachary's Karate Network*, *LocalEquilibrium* discovered five communities and three overlapping vertices (vertices 1, 33 and 34), while the *COFOGA* discovers three communities and only one overlapping vertex (vertex 10, which has two links connecting to vertices in different communities). In addition, there may be more than one overlapping vertex between two communities in the structure detected by the *COFOGA*, (e.g. vertices 22, 3, 21 and 51 in the *Lusseau's Dolphin Network*).

Figure 9 shows the *stable community structures* of the *Zachary's Karate Network* and the *Lusseau's Dolphin Network* detected by the *COFOGA*. These *stable community structures* are similar to the community structures discovered in Newman and Girvan's work (2004).

This experiment indicates that the *COFOGA* algorithm is able to discover overlapping and hierarchical communities, which by visual inspection provide meaningful information about the community structures and can be used in further investigation of community interconnections.

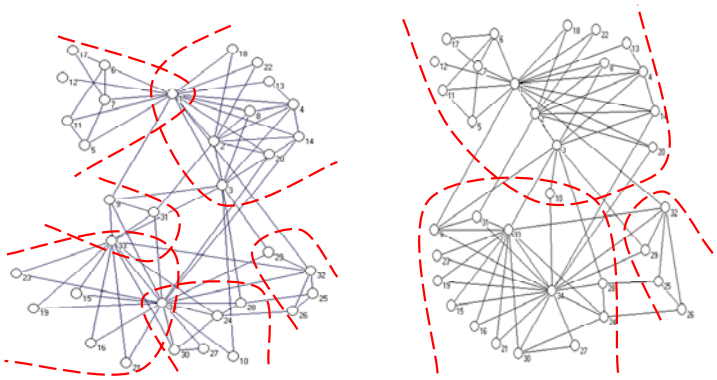

Figure 7. The community structures for the *Zachary's karate Network*. (a) The structure detected by *LocalEquilibrium* (Chen et al. 2010); (b) The structure detected by the *COFOGA* at the first level (*CoaSet*^{1}, $\alpha = 1/\sqrt{79}$, $\beta = 1$).

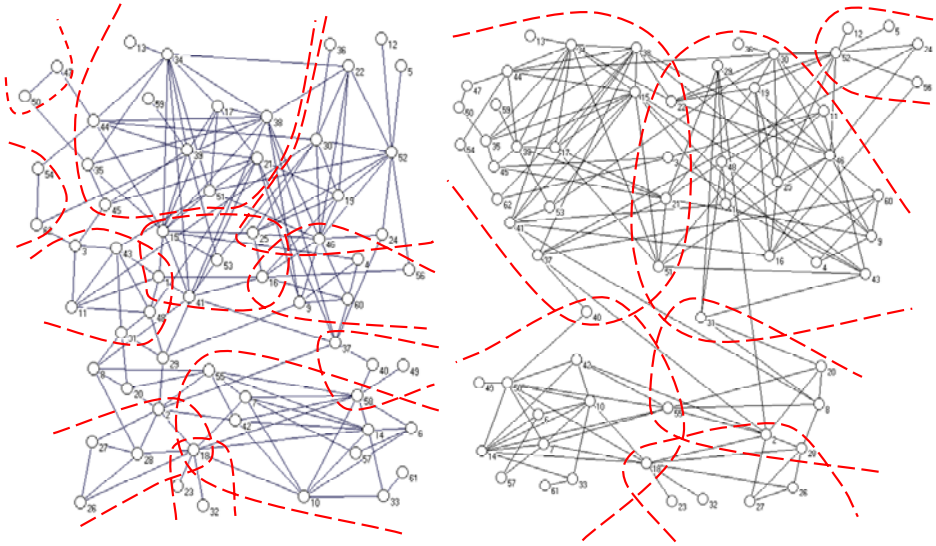

Figure 8. The community structures of the *Lusseau's Dolphin Network*. (a). The structure detected by *LocalEquilibrium* (Chen et al. 2010); (b). The structure detected by the *COFOGA* at the first level (*CoaSet*^{1}, $\alpha = 1/\sqrt{159}$, $\beta = 1$).

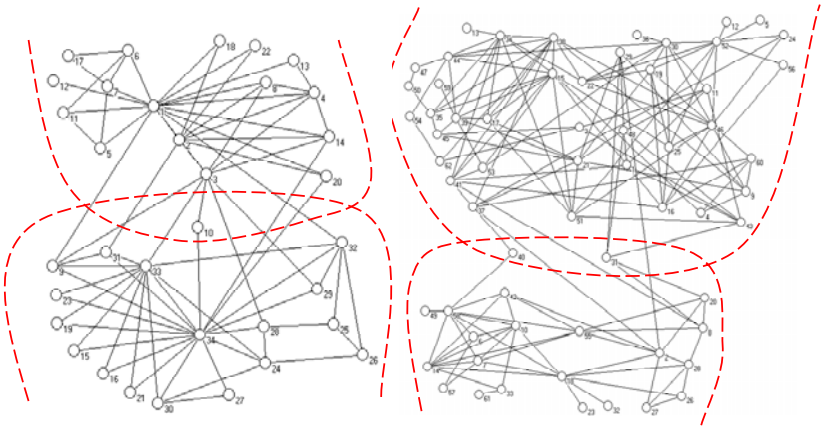

Figure 9. The stable community structures of the *Zachary's Karate Network* and the

Lusseau's Dolphin Network detected by *COFOGA.* (a). The stable community structure of the *Zachary's Karate Network* (*CoaSet*^{2}, $\alpha = 0.25$, $\beta = 1$); (b). The stable community structure of the *Lusseau's Dolphin Network* (*CoaSet*^{2}, $\alpha = 0.25$, $\beta = 1$)

5.2 Assessing the effectiveness and efficiency of the *COFOGA* algorithm in benchmark networks

We first produce a benchmark network with overlapping vertices by using Lancichinetti and Fortunato's method (2009) under following parameters: the number of vertices N =128, the average degree k=10, the maximum degree *maxk*=30, the mixing parameter, i.e. the portion of crossing edges *mu*=0.1, the minus exponent for the degree sequence *t1*=2, the minus exponent for the community size distribution *t2*=1, the minimum for the community sizes *minc*=10, the maximum for the community sizes *maxc*=30, the number of overlapping vertices *on*=10, the number of memberships of the overlapping vertices *om*=2. The benchmark structure and community structures detected by *LocalEquilibrium* and *COFOGA* are shown in Figure 10 and Table 1. In Table 1, the shaded vertices are overlapping vertices.

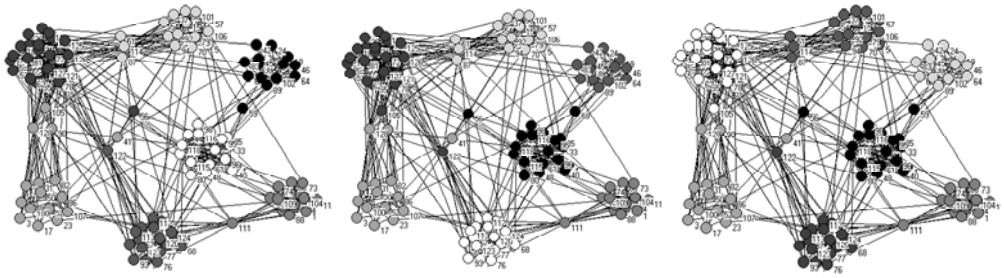

Figure 10. The community structures of the benchmark network with 128 vertices. (a) The benchmark network; (b) The community structure detected by *LocalEquilibrium*; (c) The stable community structure detected by the *COFOGA* ($\alpha = 1/\sqrt{1366}$ at the first level, $\alpha = 0.25$ at the higher level, $\beta = 0.1$).

Table 1 The structures of benchmark network and the communities detected by *LocalEquilibrium* and *COFOGA* (The vertices with shade are overlapping vertices)

No.	Benchmark	*stra-game*	*coop-game*
1	1 11 14 21 22 25 26 47 63 66 69 73 74 84 88 99 104 109 111	1 11 14 21 22 25 26 47 63 66 69 73 74 84 88 99 104 109 111	1 11 14 21 22 25 26 47 63 66 69 73 74 84 88 99 104 109 111
2	2 3 17 23 31 39 41 50 52 82 86 90 100 107 128	2 3 17 23 31 39 41 50 52 82 86 90 100 105 107 128	2 3 17 23 31 39 41 50 52 82 86 90 100 105 107 128
3	4 33 35 40 43 44 48 56 59 61 75 80 91 95 96 98 110 115 116 118	4 33 35 40 43 44 48 56 59 61 75 80 91 95 96 98 110 115 116 118	4 33 35 40 43 44 48 56 59 61 75 80 91 95 96 98 110 115 116 118
4	6 9 19 24 32 38 42 45 46 59 64 83 89 102	6 9 19 24 32 38 42 45 46 59 64 83 89 102	6 9 19 24 32 38 42 45 46 59 64 83 89 102
5	5 15 16 37 41 51 54 57 58 70 79 87 92 94 101 106 114	5 15 16 37 41 51 54 57 58 70 79 87 92 94 101 106 114	5 15 16 37 41 51 54 57 58 70 79 87 92 94 101 106 114
6	8 10 12 18 29 34 51 53 55 56 62 65 71 72 78 81 87 90 103 105 114 119 121 122 125 126 127 128	8 10 12 18 29 34 51 53 55 56 62 65 71 72 78 81 87 103 105 114 119 121 122 125 126 127 128	8 10 12 18 29 34 51 53 55 56 62 65 71 72 78 81 87 90 103 105 114 119 121 122 125 126 127 128
7	7 13 20 27 28 30 36 49 60 67 68 76 77 85 93 97 108 111 112 113 117 120 122 123 124	7 13 20 27 28 30 36 49 60 67 68 76 77 85 93 97 108 111 112 113 117 120 122 123 124	7 13 20 27 28 30 36 49 60 67 68 76 77 85 93 97 108 111 112 113 117 120 122 123 124

From Table 1, we can see that both *LocalEquilibrium* and *COFOGA* identify the number of communities and the memberships of vertices (except for vertex 150 and 90) correctly. Vertex 150 is not an overlapping vertex in the benchmark structure, but both *LocalEquilibrium* and the *COFOGA* judge that vertex 105 is an overlapping vertex. From Figure 10, we can see that vertex 105 has many links connected to different groups, so the judgment of *LocalEquilibrium* and the *COFOGA* is reasonable. Vertex 90 is an overlapping vertex in the benchmark structure and the *COFOGA* identifies it correctly, but *LocalEquilibrium* does not identify it as an overlapping vertex. From Figure 10, we can see that vertex 90 has also many links connected to different groups, so it is an overlapping vertex.

Next we produce a series of benchmark networks with overlapping vertices under different parameters and use the *normalized mutual information* (*NMI*) (Danon et al. 2005; Lancichinetti et al. 2009) between the detected community structure and the underlying ground truth as the evaluation metric (Lancichinetti et al. 2009). Figure 11 presents the *NMI* values between the community structures detected by *LocalEquilibrium*/*COFOGA* and the benchmark community structures under different fractions of overlapping vertices. Figure 12 compares the running times of *LocalEquilibrium* and *COFOGA* for detecting community structures on the produced

benchmark networks. The *x*-axis represents the portion of vertices that belong to multiple communities. Figure 13 presents the *NMI* values between the community structures detected by the *COFOGA* and the benchmark community structures under different β. The *x*-axis represents the value of β. The networks used to produce Figures 11, 12 and 13 (a)~(d) consist of 1,000 vertices, whereas those of Figure 11, 12 and 13 (e)~(h) consist of 5,000 vertices. The community sizes in Figure 11, 12 and 13 (a), (b), (e) and (f) range between *minc*=10 and *maxc*=50, and the community sizes in Figure 11, 12 and 13 (c), (d), (g) and (h) range between *minc*=20 and *maxc*=100. The mixing parameter *mu*=0.1 for Figures 11, 12 and 13 (a), (c), (e) and (g), and *mu*=0.3 for Figures 11, 12 and 13 (b), (d), (f) and (h). The other parameters are *t1*=2, *t2*=1, *k*=20, *maxk*=50 and *om*=2.

From Figure 11, we can see that both the *COFOGA* and *LocalEquilibrium* perform very well when the portion of crossing edges *mu*=0.1, with NMI being above 85 per cent, and the *COFOGA* outperforms *LocalEquilibrium* when the portion of overlapping vertices is small. For *mu*=0.3, the *COFOGA* outperforms *LocalEquilibrium* no matter the number of vertices *N*=1,000 or *N*=5,000.

From Figure 12, we can see that the *COFOGA* is much faster than *LocalEquilibrium* over all instances: the longest running time of the *COFOGA* is 11 and 67 seconds for *N*=1,000 and *N*=5,000 respectively, while the shortest running time of *LocalEquilibrium* is 185 and 203 seconds for *N*=1,000 and *N*=5,000 respectively. Moreover, the running time of *LocalEquilibrium* increases greatly with the number of vertices *N*, the portion of crossing edges *mu*, and the fraction of overlapping vertices, for example, the running time of *LocalEquilibrium* is 4,495 seconds for *N*=5,000, *mu*=0.3 and where half the vertices belong to multiple communities. However, the running time of the *COFOGA* is more stable than *LocalEquilibrium.*

From Figure 13, we can see that different β result in different *NMI* for each network. The values of β corresponding to the maximal *NMI* in different networks

are often different. How to select a suitable β for a network is a future direction we will pursue.

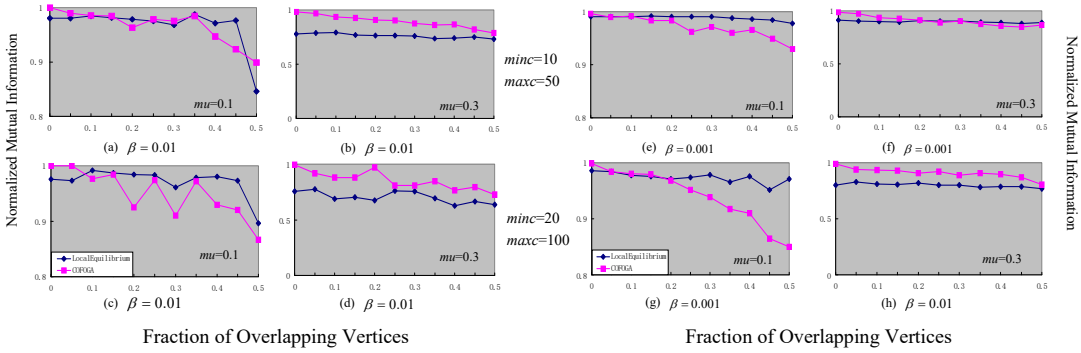

Figure 11. The *NMI* values between the community structures detected by *LocalEquilibrium*/*COFOGA* and the real community structures under different fractions of overlapping vertices, (a)~(d) consist of 1,000 vertices, (e)~(h) consist of 5,000 vertices. The minimum degree and maximum degree of the network are 20 and 50 respectively. $\alpha = 1/\sqrt{|E|}$ at the first lever and $\alpha = 0.25$ at the other levels.

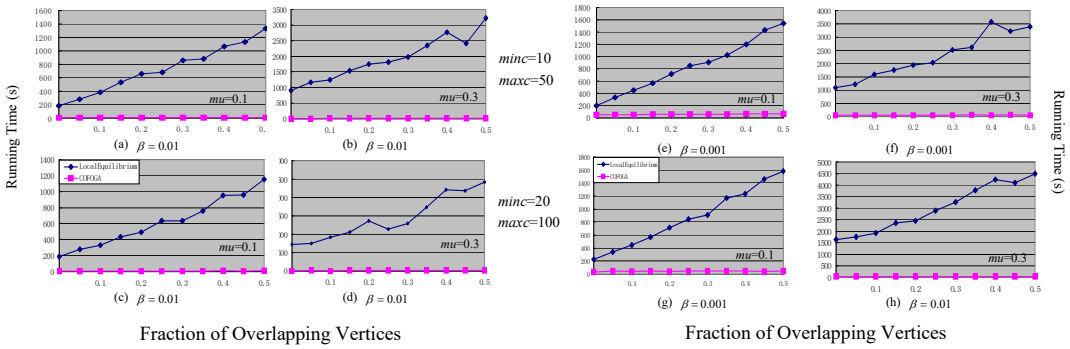

Figure 12. The running times of *LocalEquilibrium* and *COFOGA* for detecting community structures on the benchmark networks under different fractions of overlapping vertices, (a)~(d) consist of 1,000 vertices, (e)~(h) consist of 5,000 vertices. The minimum degree and maximum degree of the network are 20 and 50 respectively. $\alpha = 1/\sqrt{|E|}$ at the first lever and $\alpha = 0.25$ at the other levels.

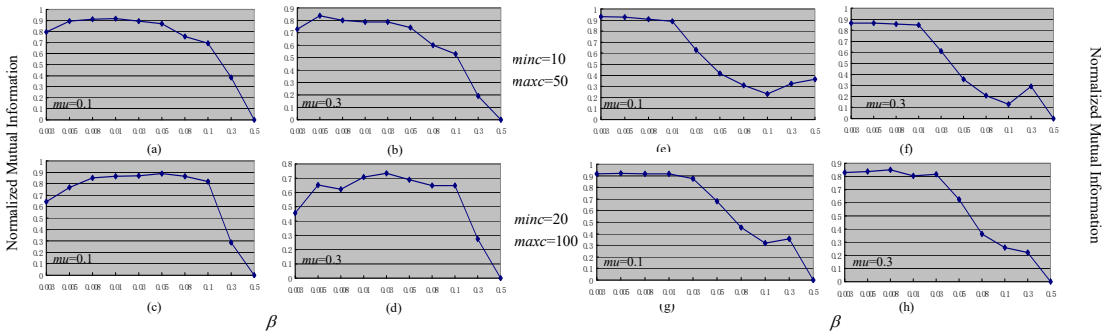

Figure 13. The *NMI* values between the community structures detected by the *COFOGA* and the real community structures under different β, (a)~(d) consist of 1,000 vertices, (e)~(h) consist of 5,000 vertices. The minimum degree and maximum degree of the network are 20 and 50 respectively. The fraction of overlapping vertices is 0.5. $\alpha = 1/\sqrt{|E|}$ at the first lever and $\alpha = 0.25$ at the other levels.

The experimental results on the benchmark networks indicate that the *COFOGA* algorithm achieves results of high quality in terms of the NMI measures, and is much faster than the *LocalEquilibrium* algorithm.

5.3 Assessing the resolution limit

The first synthetic network (*SNC1*) is made of 30 identical cliques, which are complete graphs with five vertices connected by single links. The community structure of *SNC1* detected by *LocalEquilibrium* is shown in Figure 14 (a). Figure 14 (b) shows the community structures of *SNC1* detected by *COFOGA* at the first level, in which different colours represent different communities, and vertices with the same colour belong to the same community.

The second synthetic network (*SNC2*) is made of four cliques. Two of which are complete graphs with 30 vertices, and the other two are complete graphs with 5 vertices. The community structure of *SNC2* detected by *LocalEquilibrium* is shown in Figure 15 (a), Figure 15 (b) shows the community structure of *SNC2* detected by *COFOGA* at the first level, in which different grey levels represent different communities, and vertices with the same grey level belong to the same community.

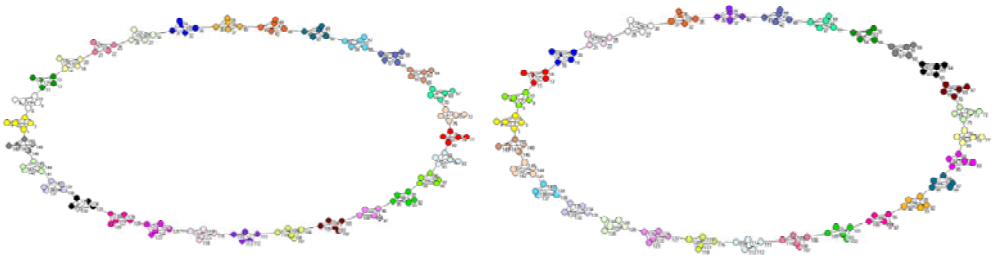

Figure 14. The community structures of *SNC1*. (a). The community structure of *SNC1* detected by *LocalEquilibrium*; (b). The community structure of *SNC1* detected by the *COFOGA* at the first level (*CoaSet*^{1}, $\alpha = 1/\sqrt{330}$, $\beta = 0.1$).

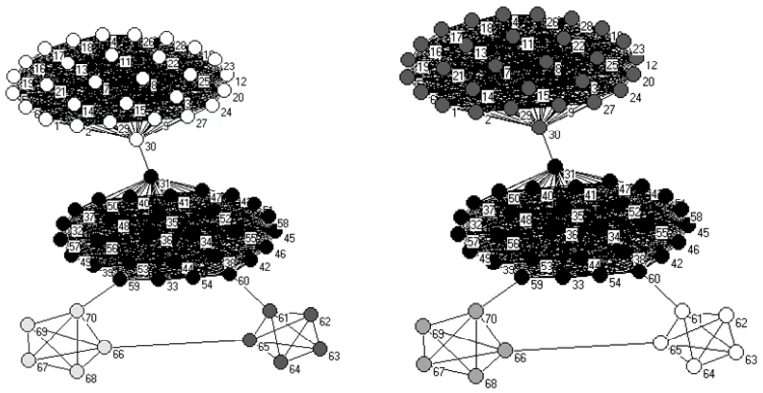

Figure 15. The community structures of *SNC2* .(a). The community structure of *SNC2* detected by *LocalEquilibrium*, The community structure of *SNC2* detected by the *COFOGA* at the first level (*CoaSet*^{1}, $\alpha = 1/\sqrt{1784}$, $\beta = 0.3$).

Figures 14 and 15 indicate that both *LocalEquilibrium* and the *COFOGA* identify community structures correctly, i.e. both of them are not limited by the resolution limit of Newman and Girvan's *modularity.*

6 Conclusions

In this paper, community detection in a social network is modelled as a coalition formation game in which individuals cooperate with each other to improve a group's utilities. This matches well with the fact that a community in fact is an interactive phenomenon amongst multiple individuals, thus the proposed approach in this paper is able to detect communities more rationally, where overlapping and hierarchical communities can be identified. Because the number of coalitions is fewer than the number of individuals, the game amongst coalitions would require less computation

than the game amongst all individuals, thus our approach, which is based on the coalition formation game, is more efficient than the approaches that are based on non-cooperative game theory. Meanwhile, our approach avoids the pre-requests for the number and the size of communities. In addition to discovering groups of related individuals in social networks, our approach can also be applied to other purposes, such as to detect sets of web pages dealing with the same topic, or biochemical pathways in metabolic networks.

In this study, we detect the community structure that maximizes the total utility by the coalition formation process in this paper, but we do not consider the evolution of this structure, i.e. the change of the community structure when one or more players joins or leaves the game.

In our utility function, α and β are two important parameters. In this study, we let $\alpha = 1/\sqrt{|E|}$ at the first level, and let $\alpha = 0.25$ at the other levels. So our experimental results are only influenced by β. The experiment results show that community structures detected by the *COFOGA* are sensitive to β. As part of our future work, we consider the design of a method to find appropriate α and β automatically, or to design a more appropriate utility function.

In the future, we will further explore the properties of the coalition formation game, especially of the utility function, for tracing the evolution of the community structure and reducing the centralized complexity, and we will make efforts to reduce the computational complexity and investigate the distributed approach for forming coalitions, which has a distinct advantage for dealing with large scale networks.

References

Alvari, H., Hashemi, S. and Hamzeh, A. (2011) Detecting overlapping communities in social networks by game theory and structural equivalence concept. *Artificial Intelligence and Computational Intelligence, Lecture Notes in Computer Science*, 7003, 620–630.

Ahn, Y. Y., Bagrow, J. P. and Lehmann S. (2010) Link communities reveal multi-scale complexity in networks. *Nature*, 466(7307), 761–764.

Bagrow, J. P. (2012) Communities and bottlenecks: trees and treelike networks have high modularity. *Physical Review E* 85, 066118.

Ball, B., Karrer, B. and Newman, M. E. J. (2011) An efficient and principled method for detecting communities in networks. *Physical Review E* 84.

Blondel, V. D., Guillaume, J. L., Lambiotte, R. and Lefebvre E. (2008) Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experim*ent, 2008, P10008.

Bo, Y., Di, J., Liu, J. and Liu, D. (2013) Hierarchical community detection with applications to real-world network analysis. *Data & Knowledge Engineering*, 83, 20–38.

Brandes, U., Delling, D., Gaertler, M., Görke, R., Hoefer, M., Nikoloski, Z. and Wagner, D. (2008) On modularity clustering. *IEEE Transaction on Knowledge and Data Engineering*, 20(2), 172–188.

Chen, W., Liu, Z., Sun, X. and Wang, Y. (2010) A game-theoretic framework to identify overlapping communities in social networks. *Data Mining and Knowledge Discovery*, 21(2), 224–240.

Clauset, A., Moore, C. and Newman, M. (2008) Hierarchical structure and the prediction of missing links in networks. *Nature,* 453(7191), 98–101.

Danon, L., Danone, Díaz-Guilera, A., Duch, J. and Arenas, A. (2005) Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005, P09008.

Fortunato, S. (2010) Community detection in graphs. *Physics Reports*, 486, 75–174.

Fortunato, S. and Barthélemy, M. (2006) Resolution limit in community detection. *Proceedings of the National Academy of Sciences of the United States of America*, 104(1), 36–41.

Francesco, F. and Clara, P. (2014) An evolutionary multiobjective approach for community discovery in dynamic networks. *IEEE Transactions on Knowledge and*

Data Engineering, 26(8), 1838–1852.

Galbrun, E., Gionis, A. and Tatti, N. (2014) Overlapping community detection in labeled graphs. *Journal of Data Mining and Knowledge Discovery*, 2828(5–6), 1586–1610.

Hajibagheri, A., Alvari, H., Hamzeh, A. and Hashemi, A. (2012) Social networks community detection using the Shapley value. *16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISwww.lw20.comP)*, Shiraz, Iran, 2–3 May, 222–227.

Krishnamurthy, B. and Wang, J. (2000) On network-aware clustering of web clients. *Computer Communication Review*, 30 (4), 97–110.

Lancichinetti, A. and Fortunato, S. (2009) Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1), 016118.

Lancichinetti, A., Fortunato, S. and Kertesz, J. (2009) Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11, 033015.

Li, G. P., Pan, Z. S., Xiao, B. and Huang, L. W. (2014a) Community discovery and importance analysis in social network. *Intelligent Data Analysis*, 18(3), 495–510.

Li, X. T., Ng, M., K. and Ye, Y. M. (2014b) MultiComm: finding community structure in multi-dimensional networks. *IEEE Transactions on Knowledge and Data Engineering*, 26(4), 929–941.

Liu, P., Raahemi, B. and Benyoucef, M. (2010) Knowledge sharing in dynamic virtual enterprises: a socio-technological perspective. *Knowledge-Based Systems,* 24(3), 427–443.

Lung, R. L., Gog, A. and Chira, C. (2012) A game theoretic approach to community detection in social networks. *Nature Inspired Cooperative Strategies for Optimization* (NICSO 2011), Studies in Computational Intelligence, 387, 121–131.

Lusseau, D. (2003) The emergent properties of a dolphin social network. *Proceedings of the Royal Society B: Biological Sciences*, 270, S186-S188.

Nash, J. F. (1951) Non-cooperative games. *Annals of Mathematics*, 54(2),

286–295.

Newman, M. E. J. and Girvan, M. (2004) Finding and evaluating community structure in networks. *Physical Review E* 69, 026113.

Palla, G., Derenyi, I., Farkas, I. and Vicsek, T. (2005) Uncovering the overlapping community structures of complex networks in nature and society. *Nature*, 435, 814–818.

Papadopoulous, S., Kompatsiaris, Y., Vakali, A. and Spyridonos, P. (2012) Community detection in social media. *Data Mining and Knowledge Discovery*, 24(3), 515–554.

Saad, W., Han, Z., Debbah, M., Hjørungnes, A. and Basar, T. (2009) Coalitional game theory for communication networks: a tutorial. *IEEE Signal Processing Magazine*, 26(5), 77–97.

Sales-Pardo, M., Guimerà, R., Moreira, A. A. and Amaral, L.A.N. (2007) Extracting the hierarchical organization of complex systems. *Proceedings of the National Academy of Sciences of the United States of America*, 104(39), 15224–15229.

Sarason, S. B. (1974) The psychological sense of community: *Prospects for a Community Psychology*. Jossey-Bass, San Francisco.

Shen, H. W., Cheng, X. Q., Cai, K. and Hu, M. B. (2009) Detect overlapping and hierarchical community structure in networks. *Physica A,* 338, 1706–1712.

Wu, F., Huberman, B., Adamic, L. and Tyler, J. (2004) Information flow in social groups. *Physica A: Statistical Mechanics and its Applications*, 337 (1–2), 327–335.

Wu, P. and Li, S. K. (2011) Social network analysis layout algorithm under ontology model. *Journal of Software*, 6(7), 1321–1328.

Wu, Y. B., Jin, R.M., Li, J., and Zhang, X. (2015) Robust local community detection: on free rider effect and its elimination. *Proceedings of the VLDB Endowment*, 8(7), 798–809.

Xin, Y., Yang, J. and Xie, Z. Q. (2015a) A semantic overlapping community detection algorithm based on field sampling. *Expert Systems with Applications*, 42

(2015), 366–375.

Xin, Y., Yang, J., Xie, Z. Q., and Zhang, J. P. (2015b) An overlapping semantic community detection algorithm base on the ARTs multiple sampling models. *Expert Systems with Applications*, 42 (2015), 3420–3432.

Yuan, W., Guan, D., Lee, Y.-K., Lee, S. and Hur, S.J. (2010) Improved trust-aware recommender system using small-worldness of trust networks, *Knowledge-Based Systems,* 23(3), 232–238.

Zacharias, G. L., MacMillan, J., Hemel, S. B. V. editors. (2008) Behavioral modeling and simulation: from individuals to societies. *National Academies Press*, Washington, DC.

Zachary, W. W. (1977) An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33, 452–473.

Zlotkin, G. and Rosenschein, J. (1994) Coalition cryptography and stability mechanisms for coalition formation in task oriented domains. *Proceedings of The Twelfth National Conference on Artificial Intelligence*, Seattle, Washington, 1–4 August, pp.432–437. AAAI Press, Menlo Park, California.

Zhao, Z., Feng, S., Wang, Q., Huang, J., Williams, G. and Fan, J. (2012) Topic oriented community detection through social objects and link analysis in social networks. *Knowledge-Based Systems,* 26, 164–173.

Zhou, L., Cheng, C., Lü, K. and Chen H. (2013a) Using coalitional games to detect communities in social networks. *Proceedings of the 14st International Conference on Web-Age Information Management (WAIM2013)*, 14–16 June, 326–331, Springer-Verlag, LNCS 7923.

Zhou, L., Lü, K., Cheng C. and Chen, H. (2013b) A game theory based approach for community detection in social networks. *Proceedings of the 29th British National Conference on Database (BNCOD2013),* 8–10 July, 268–281, Springer-Verlag, LNCS 7968.

Zhou, L. and Lü, K. (2014) Detecting communities with different sizes for social network analysis. *Computer Journal*, Oxford University Press., doi:10.1093/comjnl/bxu087.

Zhou, L., Yang, P., Lü, K., Wang, L. and Chen, H. (2015a) A fast approach for detecting overlapping communities in social networks based on game theory. S. Maneth (ed.): *Proceedings of the 31th British National Conference on Database (BICOD2015),* July 6–8, Edinburgh, Scotland, 1–12, Springer International Publishing Switzerland, LNCS 9147.

Zhou, L., Yang, P., Lü, K., Zhang, Z. and Chen, H. (2015b) A coalition formation game theory-based approach for detecting communities in multi-relational networks. Li, J. and Sun, Y. (eds.): *Proceedings of the 16st International Conference on Web-Age Information Management (WAIM2015)*, 8–10 June, 30–41, Springer International Publishing Switzerland, LNCS 9098.