Game Theoretic Clustering for Finding Strong Communities

Chao Zhao, Ali Al-Bashabsheh and Chung Chan

***Abstract*—We consider the problem of detecting meaningful communities by proposing a model in convex game in game theory and a measure for community strength. The community strength has meaningful interpretation and we call the communities found by the proposed method strong communities. We propose the dual of the objective function by which we characterize and solve the strong communities. The strong communities with respect to different strength values form a hierarchy and can be represented by a dendrogram, and strict theoretic proofs are given. For computing all the strong communities, submodular function minimization is used, and in certain cases, an augmented graph can be constructed and then we can use max-flow min-cut algorithm as an alternative. We also give simple examples to show the structure of the strong communities.**

I. Introduction

In this work, we consider the community detection problem. Community detection is a fundamental problem in various fields such as biological study and social network analysis. The definition of community may vary based on the problem and the objective, while the definition given in [2–4] are thought to be widely accepted. A general idea of the definition for a community is a group of individuals that has stronger connection among the group compared with the connections outside the group.

For conducting community detection, real world problems are usually converted to graphs with nodes representing individuals and edges representing relations. A large amount of community detection methods are developed according to various principles and objective functions, and there are surveys of community detection[5–11].

Game theory is one of the techniques that has been applied in community detection [9, 12–14], including the application for finding disjoint, overlapping and hierarchical communities.

As a systematic framework, game theory models and studies the decisions and outcomes of the players in the game[15, 16]. Generally, game theory can be classified into two categories, non-cooperative game theory and cooperative game theory. Non-cooperative game theory considers the competition between individual players, and emphasizes the strategies and payoffs of the individual players. Cooperative game theory considers the cooperation between players, and deal with the problem of allocating payoff to players according to the worth of the coalitions formed. There are non-transferable utility cooperative game, where the payoff for a player inside a coalition can't be transferred to another player in the same coalition, and transferable utility cooperative game where payoffs are considered to be able to be transferred among players in the same coalition. The core, kernel, nucleolus, Shapley value, egalitarian, etc., are important solution concepts in cooperative game[17, 18].

In this work, we propose a strength measure which is derived from cooperative game theory and has meaningful interpretation for a set of players in the game to be a strong community. The objective function is based on convex game, and its dual formula are proposed and used in analyzing the properties and characterization of the hierarchical strong communities, and the algorithms for optimizing the problem are given. General weighted digraphs are considered here, and can extend to unweighted graphs or undirected graphs.

This paper is organized as follows. In Section II, we derive the objective function which is founded on convex game in cooperative game theory, and give the definitions for community strength and strong communities. In Section III, first we give the dual form of the initial objective function, which provides the basis for the following analysis. Then we elaborate the hierarchical properties of the strong communities and show that they can be represented by a dendrogram. This motivates the computation methods for solving the problem. In Section IV, we describe how to solve the problem by submodular function minimization, and in certain cases, how to use max-flow min-cut algorithm as an alternative method. In Section V, we give examples and discuss the dendrogram and strong communities. In Section VI we conclude the work. Some of the proofs are put in [1, Appendix].

II. Problem Formulation

Given a graph $G = (V, E)$, where V is the set of nodes, and E is the set of edges. The weight of the edge $e(i, j)$ from node $\{i\}$ to node $\{j\}$ is denoted as $a_{i,j}$ and the adjacency matrix is denoted as

$$\boldsymbol{A} := [a_{ij}] \in \mathbb{R}_+^{|V|\times|V|}.$$

Here we consider both digraphs and undirected graphs, weighted and unweighted graphs here. For edges in undirected graph, each edge is regarded as bi-directed edge with the same weight for each direction. For unweighted graph, we set $a_{i,j} = 1, \forall e(i, j) \in E$. Additionally, at current stage it is assumed there is no self-loops, i.e., $a_{i,i} = 0, \forall i \in V$. The notation $|C|$ means the cardinality of a set C, and only the graphs with $|V| > 1$ are considered by default.

For convenience, we use

$$w(B, C) := \sum_{i\in B}\sum_{j\in C} a_{ij} \quad \text{for } B, C \subseteq V.$$

Then for a community $C \subseteq V$, the total weights of internal edges is $w(C, C) = \sum_{i,j\in C} a_{ij}$.

C. Chan (corresponding author, email: chung.chan@cityu.edu.hk), and C. Zhao are with the Department of Computer Science, City University of Hong Kong.

A. Al-Bashabsheh is with the Big Data and Brain Computing (BDBC) center at Beihang University, Beijing, China.

By regarding the nodes in V as players, and defining relevant function over the nodes and edges, we can formulate a game theoretical community detection problem with strength measure as follows.

Consider a cooperative game[16] characterized by (V, g), where

– V is a finite set of players with $|V| \geq 2$,
– $g : 2^V \to \mathbb{R}$ is a set function called the characteristic function, where $g(C)$ is the *worth* of the coalition $C \subseteq V$, assuming players in C cooperate to form such coalition.

Denote the *payoff* allocation for the players as a vector

$$r_V = (r_1, r_2, ..., r_{|V|}) \in \mathbb{R}^{|V|},$$

with r_i being the i-th element in r_V as the payoff allocated for i-th player.

For a coalition $C \subseteq V$, when the payoff vector is r_V, the payoff allocation for players in the coalition C is denoted as $r_C \in \mathbb{R}^{|C|}$, i.e., r_C is a payoff allocation vector only for players in C, and the values of the elements are the corresponding ones in r_V. The total payoff in the coalition C is denoted as $r(C) := \sum_{i \in C} r_i$.

Furthermore, when g is a supermodular function, the game is called convex game[16]. In this case, for $\forall B, C \subseteq V$,

$$g(B) + g(C) \leq g(B \cup C) + g(B \cap C). \tag{1}$$

Or equivalently, for $\forall B \subseteq C \subseteq V, i \in V \setminus B$,

$$g(B \cup \{i\}) - g(B) \leq g(C \cup \{i\}) - g(C), \tag{2}$$

where both sides are the increases in worth when a player i is added to a coalition. Formula (2) means that the increase in worth when a player adds to a coalition is equal or larger for a larger superset coalition, i.e., the marginal worth is non-diminishing for convex games.

For simplicity, g is thought to be normalized, i.e., $g(\emptyset) = 0$.

As for the payoff allocation, the transferable utility is considered here, i.e., the payoffs can be transferred between players in the same coalition. The *core*[19] is one of the important solution concepts in cooperative game, which is about the feasible allocation of payoffs to players.

The core of a game (V, g) is defined as[16]:

$$\begin{aligned} \text{Core}(V, g) := \{r_V \in \mathbb{R}^{|V|} \,|r(V) = g(V),& \\ r(C) \geq g(C), \forall C \subseteq V\}.& \end{aligned}$$

In the definition of the core, $r(V) \leq g(V)$ means the payoff allocation exactly splits the total worth of the grand coalition V. The inequality $r(C) \geq g(C)$ says that the total payoff for coalition C is equal to or larger than the worth of coalition C, which means when $C \subsetneq V$, players in C will have a better total payoff by cooperating in the grand coalition V than the worth of coalition C, and hence won't deviate from the grand coalition V to favor a smaller coalition C.

The definition of the core can also be extended for a subset of players in V. For $C \subseteq V, C \neq \emptyset$, the core of the subgame (C, g) as

$$\begin{aligned} \text{Core}(C, g) := \{r_C \in \mathbb{R}^{|C|} \,|r(C) = g(C),& \\ r(B) \geq g(B), \forall B \subseteq C\}.& \end{aligned}$$

With the above model and setting, we can detect communities in V by regarding the feasible and advantageous coalitions that will form under some constraints in the convex game as communities, and analyze the the property of a collection of nodes if they form a community $C \subsetneq V$ by ways of analyzing the coalition C in the convex game.

The notion of *community* in graph G and the notion of *coalition* in the convex game (V, g) are indeed equivalent and interchangeable in the sense of referring to a subset C of V, and hence we will not specially distinguish between the use of *community* and *coalition* in the following.

Our question of interest is, ***we want to identify strong communities based on the convex games by using the following measure of community strength.***

Definition 1 For $C \subseteq V : |C| > 1$, define

$$\sigma(C) := \min_{\substack{B \subsetneq C \\ B \neq \emptyset}} \max_{r_C \in \text{Core}(C,g)} \frac{r(B)}{|B|}, \tag{3}$$

then $\sigma(C)$ is referred to as the ***strength*** of community C. □

The interpretation of $\sigma(C)$ as the community strength is intuitive. Suppose the total payoff $r(C)$ for coalition C is fixed, and players in the set $B \subsetneq C$ are in need of being allocated more payoff, and hence players in $C \setminus B$ will support B by transferring some payoff to payers in B. Then the inner maximization is the largest average payoff that can be allocated to B without making the coalition C being unfavorable, i.e., still with $r_C \in \text{Core}(C, g)$. The minimization over B gives the worst case scenario of such support.

Our goal is to identify strong communities defined using σ as follows:

Definition 2 For any threshold α, $C \subseteq V : |C| > 1$, define the collection of *strong communities* in V as

$$\mathcal{C}_\alpha(V) := \text{maximal}\{C \subseteq V \mid |C| > 1, \sigma(C) > \alpha\}, \tag{4}$$

where maximal $\mathcal{F}$ denotes the inclusionwise maximal subsets in the collection $\mathcal{F}$. □

III. Main results

A. Characterization of community strength

The community strength defined in Definition 1 takes a simpler form for convex game as shown in Theorem 1.

Theorem 1 *For any* $C \subseteq V : |C| > 1$*,*

$$\sigma(C) = \min_{\substack{B \subsetneq C: \\ |B| \neq \emptyset}} \frac{g(C) - g(B)}{|C \setminus B|}, \tag{5}$$

Furthermore, the collection of optimal solutions to (3) *is given by*

$$\{C \setminus B | B \in \mathcal{B}(C)\}, \tag{6}$$

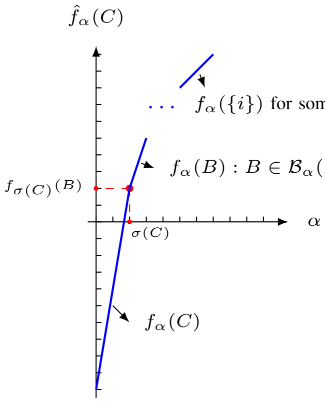

Fig. 1: Sketch curve of $\hat{f}_\alpha(C)$ defined in (7a)

where $\mathcal{B}(C)$ is the set of optimal solutions to the minimization in (5). □

Define for $\alpha \in \mathbb{R}$ and $C \subseteq V : |C| > 1$ that

$$\hat{f}_\alpha(C) := \min_{\substack{B \subsetneq C:\\ |B| \neq \emptyset}} f_\alpha(B), \quad \text{where} \tag{7a}$$

$$f_\alpha(B) := \alpha|B| - g(B), \quad \text{for } B \subseteq V. \tag{7b}$$

The collection of optimal solutions to the minimization in (7a) is denoted as $\mathcal{B}_\alpha(C)$ for α in the following.

It follows that the curve of $\hat{f}_\alpha(C)$ against $\alpha \in \mathbb{R}$ is piecewise linear, non-decreasing and convex in α. With $C \subseteq V : |C| > 1$, the curve must have at least one turning point, as illustrated in Fig. 1.

The following result shows that $\sigma(C)$ can be obtained from the curve. It will be useful in proving the subsequent results regarding the representation and computation of the strong communities defined in Definition 2.

Theorem 2 *For the curve $\hat{f}_\alpha(C)$ against $\alpha \in \mathbb{R}$:*

1) $\sigma(C)$ is the α-coordinate of the first turning point. More precisely,

$$\min_{\substack{B \subsetneq C:\\ |B| \neq \emptyset}} f_\alpha(B) < f_\alpha(C) \iff \sigma(C) < \alpha, \tag{8a}$$

$$\min_{\substack{B \subsetneq C:\\ |B| \neq \emptyset}} f_\alpha(B) = f_\alpha(C) \iff \sigma(C) = \alpha, \tag{8b}$$

$$\min_{\substack{B \subsetneq C:\\ |B| \neq \emptyset}} f_\alpha(B) > f_\alpha(C) \iff \sigma(C) > \alpha. \tag{8c}$$

2) The collection $\mathcal{B}_\alpha(C)$ of optimal solution to (7a) satisfies

$$\mathcal{B}_\alpha(C) \not\ni C, \quad \text{for } \alpha > \sigma(C), \tag{9a}$$

$$\mathcal{B}_\alpha(C) = \mathcal{B}(C) \cup \{C\}, \quad \text{for } \alpha = \sigma(C), \tag{9b}$$

$$\mathcal{B}_\alpha(C) = \{C\}, \quad \text{for } \alpha < \sigma(C). \tag{9c}$$

The result is also illustrated in the Fig. 1, where $\sigma(C)$ and the optimal solutions $\mathcal{B}(C)$ is associated with the first turning point of $\hat{f}_\alpha(C)$.

B. Representation of strong communities

The strong communities defined in Definition 2 form a hierarchy structure, and can be represented by a dendrogram.

Lemma 1 *For all $C_1, C_2 \subseteq V : C_1 \cap C_2 \neq \emptyset$,*

$$\sigma(C_1 \cup C_2) \geq \min\{\sigma(C_1), \sigma(C_2)\}. \tag{10}$$

Lemma 1 is the basis for Theorem 3. Similar formula with Lemma 1 also appeared in the info-clustering framework[20] for a different measure of strength. However, the problem we consider here is different.

Theorem 3 *For any $C_1 \in \mathcal{C}_{\alpha_1}(V), C_2 \in \mathcal{C}_{\alpha_2}(V)$ where $\alpha_1 \leq \alpha_2$, we have*

$$C_1 \supseteq C_2, \text{ or } C_1 \cap C_2 = \emptyset.$$

Furthermore,

$$\text{if } C_1 \supsetneq C_2, \text{ then } \alpha_1 < \alpha_2.$$

PROOF Suppose to the contrary that $C_1 \not\supseteq C_2$ and $C_1 \cap C_2 \neq \emptyset$. By Lemma 1,

$$\sigma(C_1 \cup C_2) \geq \sigma(C_1) > \alpha_1.$$

This contradicts $C_1 \in \mathcal{C}_{\alpha_1}(V)$, in particular, the inclusionwise maximality of C_1, since $C_1 \cup C_2 \supsetneq C_1$, where the strict inclusion is because $C_1 \not\supseteq C_2$. ■

The family $\bigcup_{\alpha \in \mathbb{R}} \mathcal{C}_\alpha(V)$ is said to be laminar and can be shown to contain at most $|V| - 1$ elements. More precisely, we will show that the family of communities together with their levels of strength can be represented by the following dendrogram with σ means the cophenetic similarity.

Definition 3 The dendrogram for the set of communities is defined as follows:

1) Every $C \in \bigcup_{\alpha \in \mathbb{R}} \mathcal{C}_\alpha(V)$ is an internal node annotated with the value $\sigma(C)$;
2) Every singleton $\{i\}$ for $i \in V$ is a leaf node (annotated with the value $+\infty$);
3) The parent of each node $B \subseteq V$ is defined as the (unique) inclusionwise minimal

$$p(B) := \text{minimal}\{C \in \mathcal{C}_\alpha(V) | B \subsetneq C, \alpha \in \mathbb{R}\}. \tag{11}$$

As illustrated in Fig. 2, the dendrogram forms a tree because each node (except the root node V) has a unique parent node. □

Proposition 1 *For every $B \subsetneq V : B \neq \emptyset$, the inclusionwise minimum element $p(B)$ exists.* □

Using the following result, we can show that the set of children for each node $C \in \bigcup_{\alpha \in \mathbb{R}} \mathcal{C}_\alpha(V)$ is

$$\mathcal{C}_{\sigma(C)}(C) \cup \{\{i\} \mid i \in V \setminus \mathcal{C}_{\sigma(C)}(C)\},$$

which is also illustrated in the figure.

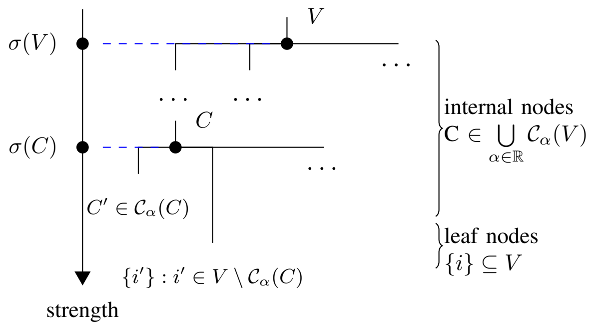

Fig. 2: Dendrogram of the communities

Proposition 2 *For any nodes $C \in \bigcup_{\alpha\in\mathbb{R}} \mathcal{C}_\alpha(V)$ of the dendrogram,*

$$p(B) = C \iff B \in \mathcal{C}_{\sigma(C)}(C), \tag{12}$$

which implies $\sigma(B) > \sigma(C)$. □

We defined the community strength in (3) by modeling the problem based on convex game in game theory, and give the alternative form of community strength in (5), and showed that the community strength and the solutions to the minimization of (5) is related with the first turning point of the curve defined by (7a) against the parameter α. We also showed that the collection of strong communities defined in (4) form a hierarchy and can be represented by a dendrogram. These motivates the methods for computing strong communities, as described in the following section.

IV. Computation of Strong Communities

In this secion, we will show how to compute all the strong communities.

The following result allows $\mathcal{C}_\alpha(V)$, the collection of all the strong communities, to be computed recursively.

Theorem 4 *For $\alpha \in \mathbb{R}$,*

$$\mathcal{C}_\alpha(V) = \mathcal{B}^*_\alpha(V) \cup \mathcal{C}_\alpha(U), \text{ where} \tag{13a}$$

$$\mathcal{B}^*_\alpha(V) := \{B \in \text{ minimal } \mathcal{B}_\alpha(V) \mid |B| > 1\}, \tag{13b}$$

$$U := V \setminus \bigcup \mathcal{B}^*_\alpha(V), \tag{13c}$$

and the recursive procedure finishes when $|U| \leq 1$ or U do not change between two recursions. Here, $\mathcal{B}_\alpha(V)$ is the set of optimal solutions to (7a) for the case $C = V$ in (7a). □

The following proposition is the basis of Theorem 4.

Proposition 3 *For any $B \in \mathcal{B}_\alpha(V)$, $\forall C \subseteq V$,*

$$B \cap C \neq \emptyset \text{ and } C \nsubseteq B \Rightarrow \sigma(C) \leq \alpha, \tag{14}$$

or the contrapositive

$$\sigma(C) > \alpha \Rightarrow B \cap C = \emptyset \text{ or } C \subseteq B. \tag{15}$$

Proof Suppose to the contrary that there exists $C \subseteq V$ s.t. $\sigma(C) > \alpha$, $B \cap C = \emptyset$ and $C \nsubseteq B$. Consider an inclusionwise maximal C. By Lemma 1 we have

$$\begin{aligned}\sigma(C \cup B) &\geq \min\{\sigma(C), \sigma(B)\}\\ &> \alpha,\end{aligned}$$

and so by maximality of C we have $C \supsetneq B$. Then we have

$$\begin{aligned}f_\alpha(C) &\overset{(a)}{<} \min_{\substack{T \subsetneq C:\\ T \neq \emptyset}} f_\alpha(T)\\ &\overset{(b)}{\leq} f_\alpha(B),\end{aligned}$$

where (a) is by (8) in Theorem 2, and (b) is by $C \supsetneq B$ as obtained above with the assumption. However, this contradicts the optimality of $B \in \mathcal{B}_\alpha(V)$. ■

According to Theorem 4, for computing strong communities, first we compute $\mathcal{B}^*_\alpha(V)$, then we get U by (13c), and continue to compute for $\mathcal{C}_\alpha(U)$ recursively until U contains at most one node, or do not change between two recursions. We will show that we can use submodular function minimization to solve the problem, and when $g(B)$ is a function of linear combination of internal and incoming / outcoming edges in B, we can also use max-flow min-cut algorithm on an augmented digraph.

A. Using submodular function minimization

For computing $\mathcal{B}_\alpha(V)$ and $\mathcal{B}^*_\alpha(V)$ in (13b) by submodular function minimization, we can write $\hat{f}_\alpha(V)$ according to (7a) and (7b) as

$$\hat{f}_\alpha(V) = \min_{\substack{B \subseteq V:\\ |B| \neq \emptyset}} f_\alpha(B), \quad \text{where} \tag{16a}$$

$$f_\alpha(B) = \alpha|B| - g(B). \tag{16b}$$

Since $g(B)$ is defined to be a supermodular function, $f_\alpha(B)$ is clearly a submodular function, and hence $\hat{f}_\alpha(V)$ is a submodular function minimization problem. What's more, by the properties of parametric submodular function minimization with α as a parameter, we can obtain all the critical α's and the corresponding solutions to the minimization in (16a), from results given by the submodular minimization methods like minimum norm base method[21]. The critical α's mean the strength values of the strong communities found, and they are turning point in curve of $\hat{f}_\alpha(C)$ in (7a).

The first run of submodular minimization only enable us to obtain $\mathcal{B}_\alpha(V)$, which usually correspond to a part of the dendrogram of the strong communities with respect to α. We need continue to update U by (13c) and apply submodular function minimization to solve $\mathcal{C}_\alpha(U)$ recursively until U contains at most one node, or do not change between two recursions. Besides applying submodular minimization recursively, we can conduct submodular minimization parallel by optimizing over all the sets containing the node in computation, and organize the hierarchical strong communities at last.

B. Using max-flow min-cut algorithm

When $g(B)$ is a function of linear combination of internal and incoming or outgoing edges of $B \subseteq V$ as follows (outgoing edges is not considered here, while it's trivial to also consider the outgoing edges):

$$g(B) = \beta \cdot w(B, B) - (1 - \beta) \cdot w(V \setminus B, B), \tag{17}$$

where $\beta \in [0, 1]$ manages the trade off between the internal edges and the incoming edges from outside of B into B. The function $g(B)$ defined in this way is supermodular[22].

The $\mathcal{B}^*_\alpha(V)$ defined in (13b) can be calculated as follows by rewriting the minimization of $\hat{f}_\alpha(V)$ as [22]

$$\hat{f}_\alpha(V) = \min_{j \in V} \hat{f}^{(j)}_\alpha(V), \text{ where} \tag{18a}$$

$$\hat{f}^{(j)}_\alpha(V) := \min_{\substack{B \subseteq V: \\ j \in B}} f_\alpha(B), \tag{18b}$$

$$f_\alpha(B) := \alpha|B| - g(B), \tag{18c}$$

and $\mathcal{B}^{(j)}_\alpha(V)$ denotes the set of optimal solutions. It follows that

$$\mathcal{B}^{(j)}_\alpha(V) = \{B \in \mathcal{B}^{(j)}_\alpha(V) \mid \hat{f}_\alpha(V) = \hat{f}^{(j)}_\alpha(V)\}, \text{and so} \tag{19}$$

$$\begin{aligned}\mathcal{B}^*_\alpha(V) = \{B \in \text{ minimal } \mathcal{B}_\alpha(V) \mid |B| > 1,\\ \mathcal{B}_\alpha(V) = \bigcup_{j \in V} \mathcal{B}^{(j)}_\alpha(V), \hat{f}_\alpha(V) = \hat{f}^{(j)}_\alpha(V)\}.\end{aligned} \tag{20}$$

When we want to compute $\mathcal{C}_\alpha(V)$ for a given α, we can first construct the augmented digraph introduced in [22] and then apply max-flow min-cut algorithm and method [23–26] for solving $\mathcal{B}^{(j)}_\alpha(V)$ for each node $j \in V$, followed by obtaining $\mathcal{B}^*_\alpha(V)$. Then, we need continue to recursively update U by (13c) and solve for $\mathcal{C}_\alpha(U)$, with the nodes in $V \setminus U$ being added to the source when constructing the augmented digraph for running max-flow min-cut algorithm, until U contains at most one node or do not change between two recursions, to get $\mathcal{C}_\alpha(V)$ finally.

When we want to compute the strong communities for all the α, we can first find all the critical α's with the method in [22] in solving $\mathcal{C}_\alpha(V)$, and then for each critical α, following the above procedures of computing $\mathcal{C}_\alpha(V)$ for a given α, to find all the strong communities in V corresponding to the current α. In practice, parallel computation is available.

V. Discussions

we choose the characteristic function g with the form given in (17) but with different β for experiments.

To illustrate the dendrogram of strong communities found by our method, the simple digraph in Fig. 3a is used as an example, and the result for the cases $\beta = 1$ and $\beta = 0.5$ are shown in Fig. 3b and Fig. 3c, respectively. The example of the calculation procedures based on Theorem 4 for the case $\beta = 1$ is put in [1, Appendix] due to space limit.

We can obtain the collection of strong communities $\mathcal{C}_\alpha(V)$ in (4) for α from the dendrogram. For example, the strong communities for $\alpha = \frac{5}{2}$ is,

$$\mathcal{C}_{\frac{5}{2}}(V) = \{\{0, 1\}, \{2, 3\}\}.$$

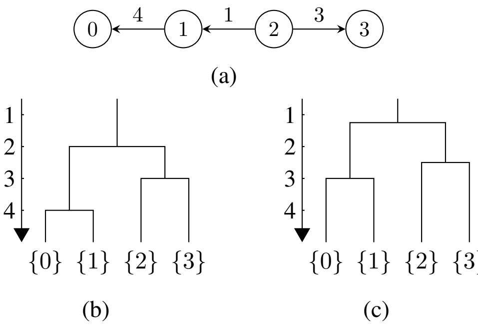

Fig. 3: A simple digraph and the dendrogram when g is defined by (17) with different β. (a) The digraph; (b) The dendrogram when $\beta = 1$; (b) The dendrogram when $\beta = \frac{1}{2}$.

The parameter β in (17) is a balancing factor between internal and incoming edges, and when $\beta = 1$, it can be used for the problem of finding densest subgraphs.

In [27], another kind of augmented graph is constructed, and an algorithm is given for quickly increasing α value based on current community that already been found and then conducting max-flow min-cut. We want to point out that, although the method there is similar with solving $\mathcal{B}_\alpha(V)$ here, the algorithm there for calculating the next critical α, as the author also said, need more calculation steps if one want to get more solutions for intermediate α's. In another word, not all critical α's are found, while our method calculate all the critical α's and the solutions directly. Additionally, our method is beyond just finding $\mathcal{B}_\alpha(V)$, and we considered more general digraphs, and can be generalized to undirected graph directly.

The strong communities are defined from game theory, where the strength can be interpreted as the support inside the community that can be shared to a part of individuals in need in the same community. Application in real world problems is promising, such as finding small groups of advertisers and key words in sponsored auction, where the community strength mean the average money inside the groups[27, 28].

VI. Conclusions

We established the community strength notion with meaningful interpretation based on convex game in cooperative game theory, and propose the dual objective function which is useful in analyzing the properties of strong communities. We showed that the hierarchical properties of the strong communities and can be represented by a dendrogram with theoretical derivations. The optimization of the objective function can be solved by submodular function minimization, and when the supermodular characteristic function in the convex game is a function given by linear combination of total internal edge weights and total incoming edge weights, max-flow min-cut method is an alternative for computing strong communities. We provide examples to illustrate the strong communities and compared with related work, and explained the superior consideration and ideas of our method.

References

[1] C. Zhao, A. Al-Bashabsheh, and C. Chan, "Game Theoretic Clustering for Finding Strong Communities," Feb. 2022. [Online]. Available: https://github.com/ao-hk/game_clustering

[2] G. W. Flake, S. Lawrence, C. L. Giles *et al.*, "Efficient identification of web communities," in *KDD*, vol. 2000, 2000, pp. 150–160.

[3] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying communities in networks," *Proceedings of the national academy of sciences*, vol. 101, no. 9, pp. 2658–2663, 2004.

[4] M. E. Newman, "Fast algorithm for detecting community structure in networks," *Physical review E*, vol. 69, no. 6, p. 066133, 2004.

[5] M. A. Javed, M. S. Younis, S. Latif, J. Qadir, and A. Baig, "Community detection in networks: A multidisciplinary review," *Journal of Network and Computer Applications*, vol. 108, pp. 87–111, 2018.

[6] S. R. Chintalapudi and M. K. Prasad, "A survey on community detection algorithms in large scale real world networks," in *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*. IEEE, 2015, pp. 1323–1327.

[7] Q. Cai, L. Ma, M. Gong, and D. Tian, "A survey on network community detection based on evolutionary computation," *International Journal of Bio-Inspired Computation*, vol. 8, no. 2, pp. 84–98, 2016.

[8] C. Pizzuti, "Evolutionary computation for community detection in networks: a review," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 3, pp. 464–483, 2017.

[9] A. Jonnalagadda and L. Kuppusamy, "A cooperative game framework for detecting overlapping communities in social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 491, pp. 498–515, 2018.

[10] X. Su, S. Xue, F. Liu, J. Wu, J. Yang, C. Zhou, W. Hu, C. Paris, S. Nepal, D. Jin, Q. Z. Sheng, and P. S. Yu, "A comprehensive survey on community detection with deep learning," *arXiv preprint arXiv:2105.12584*, 2021.

[11] F. Liu, S. Xue, J. Wu, C. Zhou, W. Hu, C. Paris, S. Nepal, J. Yang, and P. S. Yu, "Deep learning for community detection: Progress, challenges and opportunities," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 2020, pp. 4981–4987.

[12] S. Athey, E. Calvano, and S. Jha, "A theory of community formation and social hierarchy," 2016.

[13] R. P. Gilles, *The cooperative game theory of networks and hierarchies*. Springer Science & Business Media, 2010, vol. 44.

[14] X. Zhou, X. Zhao, Y. Liu, and G. Sun, "A game theoretic algorithm to detect overlapping community structure in networks," *Physics Letters A*, vol. 382, no. 13, pp. 872–879, 2018.

[15] O. Morgenstern and J. Von Neumann, *Theory of games and economic behavior*. Princeton university press, 1953.

[16] G. Chalkiadakis, E. Elkind, and M. Wooldridge, "Computational aspects of cooperative game theory," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 5, no. 6, pp. 1–168, 2011.

[17] R. B. Myerson, *Game theory: analysis of conflict*. Harvard university press, 1997.

[18] A. Jonnalagadda and L. Kuppusamy, "A survey on game theoretic models for community detection in social networks," *Social Network Analysis and Mining*, vol. 6, no. 1, p. 83, 2016.

[19] B. M. Roger, "Game theory: analysis of conflict," *The President and Fellows of Harvard College, USA*, 1991.

[20] C. Chan, A. Al-Bashabsheh, Q. Zhou, T. Kaced, and T. Liu, "Info-clustering: A mathematical theory for data clustering," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 2, no. 1, pp. 64–91, 2016.

[21] S. Fujishige, *Submodular functions and optimization*. Elsevier, 2005, vol. 58.

[22] C. Chan, A. Al-Bashabsheh, C. Zhao *et al.*, "Finding better web communities in digraphs via max-flow min-cut," in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 410–414.

[23] A. V. Goldberg, S. Hed, H. Kaplan, R. E. Tarjan, and R. F. Werneck, "Maximum flows by incremental breadth-first search," in *European Symposium on Algorithms*. Springer, 2011, pp. 457–468.

[24] A. V. Goldberg, S. Hed, H. Kaplan, P. Kohli, R. E. Tarjan, and R. F. Werneck, "Faster and more dynamic maximum flow by incremental breadth-first search," in *Algorithms-ESA 2015*. Springer, 2015, pp. 619–630.

[25] V. Kolmogorov, "A faster algorithm for computing the principal sequence of partitions of a graph," *Algorithmica*, vol. 56, no. 4, pp. 394–412, 2010.

[26] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan, "A fast parametric maximum flow algorithm and applications," *SIAM Journal on Computing*, vol. 18, no. 1, pp. 30–55, 1989.

[27] K. J. Lang and R. Andersen, "Finding dense and isolated submarkets in a sponsored search spending graph," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 2007, pp. 613–622.

[28] J. Auerbach, J. Galenson, and M. Sundararajan, "An empirical analysis of return on investment maximization in sponsored search auctions," in *Proceedings of the 2nd International Workshop on Data Mining and Audience Intelligence for Advertising*, 2008, pp. 1–9.

VII. Appendix

Proof of Theorem 1

Proof For $\forall B \subsetneq C, B \neq \emptyset$, by the relationship between $r(\cdot)$ and $g(\cdot)$ when $r_C \in \text{Core}(C, g)$, we have

$$r(B) = r(C) - r(C \setminus B) \leq g(C) - g(C \setminus B),$$

where the equality is achieved when $r(C \setminus B) = g(C \setminus B)$. Hence,

$$\max_{r_C \in \text{Core}(C,g)} \frac{r(B)}{|B|} \leq \frac{g(C) - g(C \setminus B)}{|B|} \tag{21}$$

Next we will show that $\exists r_C \in \text{Core}(C, g)$ s.t. $r(C \setminus B) = g(C \setminus B)$, so that the equality is achieved in (21).

To ease the notation, denote

$$T := C \setminus B. \tag{22}$$

With T, (21) becomes

$$\max_{r_C \in \text{Core}(C,g)} \frac{r(B)}{|B|} \leq \frac{g(C) - g(T)}{|C \setminus T|} \tag{23}$$

For the convex subgame (T, g), the core of (T, g) exists by the property of convex game. Suppose $r_T \in \text{Core}(T, g)$, which means

$$r(T) = g(T), \text{and} \tag{24}$$
$$r(T') \geq g(T'), \forall T' \subseteq T. \tag{25}$$

Then we show the steps to construct a vector r_C from r_T with $r_C \in \text{Core}(C, g)$.

Step 1: Select an i, with

$$i \in \left\{ \arg\max_{j: j \in C \setminus T} g(T \cup \{j\}) - g(\{j\}) \right\}. \tag{26}$$

Step 2: Assign

$$r_i = g(T \cup \{i\}) - g(\{i\}), \tag{27}$$

and construct $r_{T \cup \{i\}}$ by values from r_T and r_i.

For $\forall T' \subseteq T$, by supmodularity of g,

$$g(T' \cup \{i\}) - g(T') \leq g(T \cup \{i\}) - g(T). \tag{28}$$

Then,

$$\begin{aligned} r(T' \cup \{i\}) &= r(\{i\}) + r(T') && (29a) \\ &= g(T \cup \{i\}) - g(\{i\}) + r(T') && (29b) \\ &\geq g(T \cup \{i\}) - g(\{i\}) + g(T') && (29c) \\ &\geq g(T' \cup i), && (29d) \end{aligned}$$

where (27) and (28) are used to obtain the inequality.

Combine (29), (24) and (25), it indicates that the constructed

$$r_{T \cup \{i\}} \in \text{Core}(T \cup \{i\}, g). \tag{30}$$

Update T by $T \cup \{i\}$, and continue the Step 1 and Step 2 above, and finally we will have a constructed

$$r_C \in \text{Core}(C, g), \tag{31}$$

with $r(T) = g(T)$ preserved.

This means, for $\forall B \subsetneq C, B \neq \emptyset$, the formula (21), or its equivalent formula (23), can achieve equality with the constructed r_C obtained in (31) .

Therefore, the minimization over the non-empty proper subset of C for the right hand side of (23), will lead to a value equal to the minimization over the non-empty proper subset of left hand side of (23), which is the community strength for C defined in (3). And based on above proof, we know that, if a non-empty set B^* is a solution for the minimization in (3), the set $T^* := C \setminus B^*$ will be a solution for the minimization in (5).

Hence Theorem 1 is established.

There are other ways which may seem simpler to prove the duality between (3) and (5), such as by using the strong duality and weak duality in the linear programming, while we think the above proof helps the understanding of community strength better. ■

Proof of Theorem 2.

Proof We first show the second property in Theorem 2, which will imply the first property.

Denote the sign of a number x by

$$sgn(x) := \begin{cases} 1, & x > 0 \\ -1, & x < 0 \\ 0, & x = 0. \end{cases} \tag{32}$$

We have

$$\begin{aligned} &sgn(\sigma(C) - \alpha) \\ &\quad = sgn(\min_{\substack{B \subsetneq C: \\ |B| \neq \emptyset}} \frac{g(C) - g(B)}{|C \setminus B|}) && (33a) \\ &\quad = sgn(\min_{\substack{B \subsetneq C: \\ |B| \neq \emptyset}} g(C) - g(B) - \alpha|C \setminus B|) && (33b) \\ &\quad = sgn(\min_{\substack{B \subsetneq C: \\ |B| \neq \emptyset}} f_\alpha(B) - f_\alpha(C)), && (33c) \end{aligned}$$

where

- (33a) is by (5);
- (33b) is because $|C \setminus B| > 0$ and $B \neq C$ and $sgn(x) = sgn(ax)$ if $a > 0$;
- (33c) is by the definition of f_α in (8).

Note also that the sets of optimal solutions to the minimization / maximization in each step are the same. Hence,

$$\begin{aligned} \sigma(C) < \alpha &\iff sgn(\sigma(C) - \alpha) < 0 && (34a) \\ &\iff sgn(\min_{\substack{B \subsetneq C: \\ |B| \neq \emptyset}} f_\alpha(B) - f_\alpha(C)) < 0 && (34b) \\ &\iff \min_{\substack{B \subsetneq C: \\ |B| \neq \emptyset}} f_\alpha(B) < f_\alpha(C) && (34c) \\ &\iff C \notin \mathcal{B}_\alpha(C), && (34d) \end{aligned}$$

where

- (34a) is by (32);

- (34b) is by (33c);
- (34c) is by (32);
- (34d) is by definition of $\mathcal{B}_\alpha$.

(34c) and (34d) implies (8a) and (9a), respectively.

Similarly, with the inequalities "<" replaced by ">", and simply by definition of $\mathcal{B}_\alpha$, we have

$$\begin{aligned}\sigma(C) > \alpha &\iff \min_{\substack{B\subsetneq C:\\ |B|\neq\emptyset}} f_\alpha(B) > f_\alpha(C)\\ &\iff \{C\} = \mathcal{B}_\alpha(C),\end{aligned}$$

which implies (8c) and (9c).

With the inequalities replaced by equalities, we have

$$\sigma(C) = \alpha \iff \min_{\substack{B\subsetneq C:\\ |B|\neq\emptyset}} f_\alpha(B) = f_\alpha(C),$$

which implies (8b) and (9b). This completes the proof. ■

Proof of Proposition 1

PROOF Consider $B \subsetneq V : B \neq \emptyset$. An inclusionwise minimal element in the set in its parent $p(B)$ exists since V is in the set. Suppose to the contrary that there can be multiple minimal elements, say $C_1 \in \mathcal{C}_\alpha(V)$ and $C_2 \in \mathcal{C}_\alpha(V)$ with $\alpha_1 \leq \alpha_2$ w.l.o.g.

By Theorem 3, since $C_1 \cap C_2 \supseteq B \neq \emptyset$, we have $C_2 \subseteq C_1$, contradicting the minimality of C_1 as desired. Hence $p(B)$ exists and is unique. ■

Proof of Proposition 2

PROOF To prove the "if" case, consider $B \in \mathcal{C}_{\sigma(C)}(C)$, which implies by definition of $\mathcal{C}_\alpha$ in (2) that $\sigma(B) > \sigma(C)$ and $B \subsetneq C$, and so C satisfies the condition in (11). To show minimality, note that the maximality of $B \in \mathcal{C}_{\sigma(C)}(C)$ by definition of $\mathcal{C}_\alpha$ in (2) indicates that,

$$\forall C' \subseteq C : B \subsetneq C', \sigma(C') \leq \sigma(C).$$

Hence, $p(B) = C$.

Consider the reverse case, i.e., $p(B) = C$ and any $\alpha' \in \mathbb{R}$ s.t. $B \in \mathcal{C}_{\alpha'}(V)$. Then

$$\sigma(B) > \alpha' \geq \sigma(C),$$

where the last inequality is by Theorem 3 because $C \notin \mathcal{C}_{\alpha'}(V)$ as $C \supsetneq B$. To show $B \in \mathcal{C}_{\sigma(C)(C)}$, it suffices to show maximality of B in $\mathcal{C}_{\sigma(C)}(C)$, i.e.,

$$\forall C' \subsetneq C : C' \supsetneq B, \sigma(C') \leq \sigma(C).$$

Suppose to the contrary that there exists

$$C' \subsetneq C : C' \supsetneq B, \sigma(C') > \sigma(C).$$

Then, there must also exist

$$C'' \in \mathcal{C}_{\sigma(C)}(C) : C'' \supseteq C'.$$

By Theorem 3, since $C'' \cap C \supseteq B \neq \emptyset$, we must have $C'' \subsetneq C$, contradicting the minimality of C by definition of parent in (11) as desired.

Hence, Proposition 2 is established. ■

Proof of Theorem 4.

PROOF First we show that

$$\mathcal{B}^*_\alpha(V) \subseteq \mathcal{B}_\alpha(V). \tag{35}$$

Consider any $B \in \mathcal{B}^*_\alpha(V)$. By minimality that $B \in$ minimal$\mathcal{B}_\alpha(C)$,

$$f_\alpha(B) > \min_{\substack{T\subsetneq B\\ T\neq\emptyset}} f_\alpha(T),$$

which implies $\sigma(B) > \alpha$ by (8) since $|B| > 1$. Then by Proposition 3, B is maximal among $\{B' \mid B' \subseteq V, \sigma(B') > \alpha\}$, which means $B \in \mathcal{C}_\alpha(V)$.

It remains to show

$$\mathcal{C}_\alpha(V) \setminus \mathcal{B}^*_\alpha(V) = \mathcal{C}_\alpha(V). \tag{36}$$

First, we show that any element of l.h.s of (36) is in r.h.s of (36).

Consider any element C of l.h.s of (36), i.e., $C \in \mathcal{C}_\alpha(V)$ and $C \notin \mathcal{B}_\alpha(V)$. Since $C \in \mathcal{B}_\alpha(V)$, we have $\sigma(C) > \alpha$. For all $B \in \mathcal{B}^*_\alpha(V)$, since $B \in \mathcal{C}_\alpha(V)$ by (35) and $C \neq B$, we have $C \nsubseteq B$ by the maximality of $C \in \mathcal{C}_\alpha(V)$ in (4). By Proposition 3, $B \cap C = \emptyset$ and so

$$C \subseteq V \setminus \bigcup \mathcal{B}_\alpha(V) = U$$

by definition of U, and then $C \in \mathcal{C}_\alpha(U)$.

Next, we show that any element of l.h.s of (36) is in r.h.s of (36).

Consider any element C of the r.h.s. of (36). Since $C \in \mathcal{C}_\alpha(U)$, we have $\alpha(C) > \alpha$. For any $C' \subseteq V : C' \supsetneq C$, on one hand, we have $B \cap C' \neq \emptyset$ for some $B \in \mathcal{B}^*_\alpha(V)$; on another hand, we also have $C' \nsubseteq B$, since $C' \setminus B \supseteq C \setminus B = C$ which is a non-empty subset of $U \subseteq V \setminus B$. By Proposition 3, we have $\sigma(C') \leq \alpha$. Hence, C is maximal among subsets of $\{C'' | C'' \subseteq V, \sigma(C'') > \alpha\}$, which implies $C \in \mathcal{C}_\alpha(V)$. Therefore $C \subseteq \mathcal{C}_\alpha(V) \setminus \mathcal{B}^*_\alpha(V)$ otherwise C will not in l.h.s of (36), contradicting the case of C we consider here.

Hence, (36) is shown.

The combination of (35) and (36) establishes Theorem 4. ■

Calculation procedure for Fig. 3 when $\beta = 1$.

As an example, we show how to obtain the dendrogram in Fig. 3b and then obtain the strong communities below.

Following the recursive procedure in Theorem 4, to solve $\mathcal{C}_\alpha(V)$, first seek to solve $\mathcal{B}^*_\alpha(V)$, then for each critical α value at turning point, solve for $\mathcal{C}_\alpha(U)$ with similar procedure, where U is the complement set of V given by (13c). Hence, the procedure to solve $\mathcal{B}^*_\alpha(V)$ is representative, which can be done by following (18b) to (20).

When $j=0$ is the sink node,

$$\begin{aligned}
f_\alpha(\{0\}) &= \alpha \cdot |(\{0\})| - g(\{0\}) = \alpha \\
f_\alpha(\{0,1\}) &= \alpha \cdot |(\{0,1\})| - g(\{0,1\}) = 2\alpha - 4 \\
f_\alpha(\{0,1,2\}) &= 3\alpha - 5 \\
f_\alpha(\{0,1,2,3\}) &= 4\alpha - 8,
\end{aligned}$$

and $f_\alpha(\{0,1,3\}), f_\alpha(\{0,2\}), f_\alpha(\{0,2,3\}), f_\alpha(\{0,3\})$ can be written out in similar way and we omit them here. Then drawing f_α against α, the lowest curve formed is $\hat{f}_\alpha^{(0)}(V)$, and the corresponding solution $\mathcal{B}_\alpha^{(0)}(V)$ can be obtained.

$$\hat{f}_\alpha^{(0)}(V) = \begin{cases} 4\alpha - 8, & \alpha < 2 \\ 2\alpha - 4, & 2 \leq \alpha < 4 \\ \alpha, & \alpha \geq 4 \end{cases}$$

and correspondingly,

$$\mathcal{B}_\alpha^{(0)}(V) = \begin{cases} \{0,1,2,3\}, & \alpha < 2 \\ \{0,1\}, & 2 \leq \alpha < 4 \\ \{0\}, & \alpha \geq 4 \end{cases}$$

Then, for all $j \in V$, calculate $\hat{f}_\alpha^{(j)}(V)$ and $\mathcal{B}_\alpha^{(j)}(V)$, and finally obtain $\hat{f}_\alpha(V)$ by (16a) and $\mathcal{B}_\alpha^*(V)$ by (20).

The result is

$$\mathcal{B}_\alpha^*(V) = \begin{cases} \{0,1,2,3\}, & \alpha < 2 \\ \{0,1\}, & 2 \leq \alpha < 4 \end{cases}$$

It's easy to check that when $\alpha \geq 4$, the U in (13c) won't change between two runs, and hence can stop. For $2 \leq \alpha < 4$, we need continue to calculate $\mathcal{C}_\alpha(U)$ with $U = \{2,3\}$, and with similar calculation procedure, we can obtain for $2 \leq \alpha < 4$,

$$\mathcal{B}_\alpha^*(U) = \{2,3\}, 2 \leq \alpha < \frac{5}{2}$$

Calculation finished.