BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 06/06 pp245–256
Volume 1, Number 3, September 2018
DOI: 10.26599/BDMA.2018.9020023

A Multi-granularity Decomposition Mechanism of Complex Tasks Based on Density Peaks

Ziling Pang, Guoyin Wang*, and Jie Yang

Abstract: There are many algorithms for solving complex problems in supervised manner. However, unsupervised tasks are more common in real scenarios. Inspired by the idea of granular computing and the characteristics of human cognitive process, this paper proposes a complex tasks decomposition mechanism based on Density Peaks Clustering (DPC) to address complex tasks with an unsupervised process, which simulates the multi-granular observation and analysis of human being. Firstly, the DPC algorithm is modified to nullify its essential defects such as the difficulty of locating correct clustering centers and classifying them accurately. Then, the improved DPC algorithm is used to construct the initial decomposition solving space with multi-granularity theory. We also define subtask centers set and the granulation rules to guide the multi-granularity decomposing procedure. These rules are further used to decompose the solving space from coarse granules to the optimal fine granules with a convergent and automated process. Furthermore, comprehensive experiments are presented to verify the applicability and veracity of our proposed method in community-detection tasks with several benchmark complex social networks. The results show that our method outperforms other four state-of-the-art approaches.

Key words: multi-granularity; task decomposition; density peaks; complex network

1 Introduction

Intelligent mining and analysis of multi-dimensional mass data is a significant direction for curbing the challenges faced in data mining, such as the complexity of problems, uncertainty of information, and dynamic change of data. These approaches have been much explored in various fields of complex tasks decomposition.

To solve complex problems and tasks, it is necessary to seek answers by considering the combination of several different granularity spaces. Hobbs[1] presented a framework of granular theory, which was a simplified approach to handle complex problems. The theory of quotient space provided a method to describe different granule spaces; it focuses on the conversion between different granule spaces and optimization of granule space[2]. In Ref. [3], Professor Jang proposed a fuzzy inference system based on adaptive network, ANFIS. The system achieved fuzzy reasoning using five neuron layers. Wang and Shi[4] proposed a multi-layer three/multi-valued logic neural network, TMLNN, which explicitly represented the logical knowledge in the neural network and decomposed complex tasks to a certain extent. Deep learning was based on deep neural network model[5], which includes several hidden layers to abstract characteristics of fine grains and coarse grains. In Ref. [5], problems and tasks were figured out by high-level abstract features. Therefore, it is a key research issue to study multi-level and multi-granularity of complex tasks. Complex tasks decomposition means decomposing complex tasks into several smaller subtasks for easy solving. Many classical classification algorithms solve smaller

• Ziling Pang, Guoyin Wang, and Jie Yang are with the Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Post and Telecommunication, Chongqing 400060, China. E-mail: woshi@pangziling.com; wanggy@ieee.org; 530966074@qq.com.

∗ To whom correspondence should be addressed.

Manuscript received: 2017-09-08; accepted: 2018-03-30

subtasks effectively, and by merging these solutions, the original complex tasks can be solved precisely and quickly.

The core challenge of tasks decomposition is in selecting the right method to decompose a large complex task into several small and simple subtasks. There are three decomposition methods. One of them is decomposition by domain experts, whereby experts first decompose the tasks into a series of subtasks before machine learning. The neural network structure based on this method has been used to solve the truck backer problem[6] and in remote sensing information processing[7]. Another method is decomposition by category, whereby the problem is decomposed into a series of sub-problems before learning. It is based on the inherent relationship among the categories in the training data. For instance, OVA (One-Versus-All) mainly deals with static data sets. Perdisci applied it to discover the anomaly of computer network[8]. Reference [9] employs Adaboost, a two-class classifier, to get good results in handwriting recognition. The third method is automatic decomposition, which when learning the original problem, utilizes a classifier to divide all samples into two parts, learning correctly and incorrectly, and then retrains the classifier to recursively learn the incorrect samples before decomposing the original task into a series of subtasks[10].

However, in the event of complex data in real world, it is difficult to achieve good results using the above decomposition methods. For example, decomposition by domain experts depends on sufficient prior knowledge, which is difficult to measure and acquire. For decomposition by category, when using OVA and OVO (One-Versus-One) strategies, the former often causes a serious unbalanced sub-data, which does not reduce the size of sub-samples, while the latter inevitably produces large and complex calculations when the data set has plenty classes. For automatic decomposition, fragments are produced in the decomposing process, namely sub-problem of noise, which negatively impacts the correct decomposition of a task. Moreover, in real world, we mostly deal with non-labeled data, and unsupervised learning is critical.

The purpose of this paper is to process some complex problems in real world with the proposed decomposition mechanism based on multi-granularity and density peaks theories. Our contribution consists of two stages:

(1) Global tasks leading tree: Based on the idea of leading tree proposed in Ref. [11], we generate links between task nodes and map the whole nodes to construct a global tasks leading tree. This global tasks leading tree is considered as a primal solving space in which coarse grain layer contains global concepts information.

(2) Multi-granularity task solving space: Based on the first stage, we refine the primal coarse grain layer. By selecting redundant center subtasks, measuring the similarity of subtasks set, and defining granulation rules, we generate several multi-granularity task solving spaces, and then, according to practical complex issue, we proceed with granularity optimization to obtain the best layer to solve the corresponding problem.

The rest of this paper is organized as follows: Section 2 describes some related works about research on density peaks and the leading tree; Section 3 defines the proposed method in details; Section 4 shows the experiment results to prove our mechanism, and we conclude in Section 5.

2 Related Work

2.1 Clustering by fast search and find of density peaks

In 2014, Alex Rodriguez and Alessandro Laio published an article titled “Clustering by fast search and find of density peaks[12]” in *Science*. It provided a novel and efficient clustering algorithm that requires few parameters and does not involve iteration. The density peak algorithm has attracted widespread attention in academic circles and has been applied in many fields, including remote sensing image analysis[13], age estimation in image characters[14], fundamental matrix estimation in computer vision[15], chemical analysis[16], text discovery[17], social network[18], and image classification[19].

The idea of DPC algorithm is based on two assumptions: First, each cluster center has relatively high local density; Second, each cluster center keeps a relatively large distance from another point with a higher local density. Next, we note a brief introduction of the algorithm. According to the definition of local density, the local density of each node is computed as Eq. (1) and Eq. (2).

$$\rho_i = \sum_{j}^{I} \chi\left(d_{ij} - d_c\right) \tag{1}$$

For small data sets, local density calculated by Eq. (1) is prone to large statistical. Hence, Gaussian kernel is adopted to improve the calculation of local density, which is defined as Eq. (2).

$$\rho_i = \sum_j^I \exp\left(\frac{d_{ij}^2}{d_c^2}\right) \tag{2}$$

where ρ_i is the local density of node i, d_{ij} is the relative distance between node i and node j, d_c is the cutoff distance. The minimal distance between a node and the higher local density nodes is calculated by Eq. (3).

$$\delta_i = \begin{cases} \min\left\{d_{ij}\right\}, & j \in I_S^i,\ I_S^i \neq \varnothing; \\ \max\left\{d_{ij}\right\}, & j \in I_S^i,\ I_S^i = \varnothing \end{cases} \tag{3}$$

where $I_S^i = \left\{k \in I_S \mid \rho_k > \rho_i\right\}$.

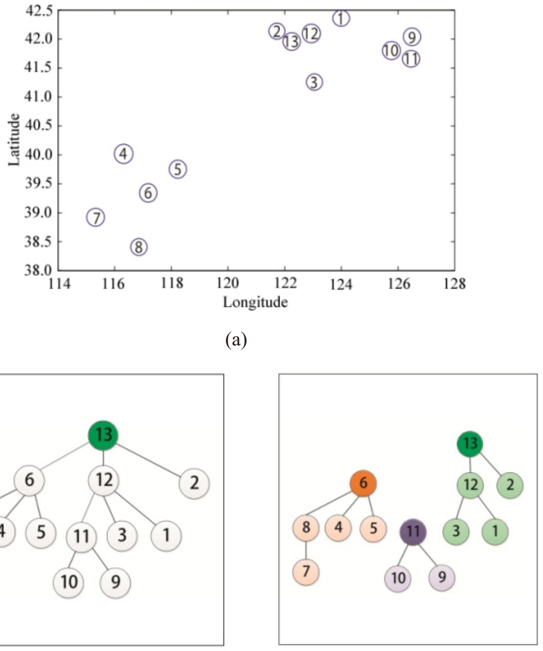
However, the DPC algorithm cannot avoid several vital problems, such as the difficulty to locate correct clustering centers and accurately classify them, and a high time-consumption. For the defects of DPC, scholars carry out heated research from different perspectives. Xu and Wang[11] proposed a way to generate a global concept problem map by the intermediate results of DPC; they named the map "leading tree" which could improve the operating time when classifying halo data. Based on the leading tree concept, Xu et al.[20,21] adopted linear regression method to select clustering centers to improve the problem of center selection difficulty, and proposed a hierarchical clustering algorithm to process LSHD data sets. In DPC, another problem is selection threshold *dc*. In Ref. [22], authors combined physical theories with DPC according to different data features. They used data field to adaptively generate applicable threshold *dc* and got more objective results. In Refs. [23, 24], peak density based on the KNN algorithm could explore and efficiently discover the densities of peak samples, and the corresponding experiments proved that these methods had achieved outstanding results. Furthermore, preprocessed data sets by PCA[24] resulted in good performance of DPC in low-dimensional and high-dimensional data sets. In Ref. [25], the geodesic distance used to calculate the distance matrix made DPC more suitable for high-dimensional data clustering. In Ref. [26], DPC was improved by chameleon hierarchical clustering algorithm. It allows DPC divide the data set into several subcategories, which offsets its weakness. Through custom connectivity and closeness measure formulas sub-classes were evaluated for similarity and the most similar ones were merged until there was none to merge.

2.2 Leading tree in DPC

The concept of leading tree was first proposed by Xu and Wang[11]. Careful investigation reveals that the intermediate result *Nneigh* in DPC essentially represents a tree, where each node, except the root, is led by its parent to join the same cluster. *Nneigh* is the index of the nearest neighbor with larger ρ for data points. By assigning the non-center data points into the parent center, we can get a lot of small families. By splitting the small families into branches, we can get the clusters in the form of subtrees. We show an example in Ref. [11] to improve understanding of the leading tree process. The example is to cluster 13 2D points, named DS, as shown in Fig. 1a.

First, compute the intermediate result *Nneigh* of DS with DPC, as shown in Table 1. The leading tree of DS is shown in Fig. 1b. According to the greatest $\rho \times \delta$ value, points 13, 11, and 6 have been selected as centers, and the leading tree of DS is split into 3 subtrees (shown in Fig. 1c), which corresponds to the clusters CL in Table 1.

Additionally, the leading tree connects the nodes with correlative parent-child links, which is an excellent way to map the global nodes. However, the leading tree can



(a)

(b)

(c)

Fig. 1 (a) Dataset DS which contains the latitude and longitude of 13 cities; (b) Leading tree of DS; and (c) Clustering result of DS.

Table 1 The intermediate results of DS.

i	$Nneigh_i$	CL
1	12	1
2	13	1
3	12	1
4	12	1
5	6	2
6	13	2
7	8	2
8	6	2
9	11	3
10	11	3
11	12	3
12	13	1
13	0	1

trace back to the original DPC algorithm, so the defect of DPC algorithm still exists. Fortunately, by consulting the leading tree, we improve the problem to a certain extent.

3 Proposed Method

In this paper, we simulate the characteristics of the multi-granule observation and analysis of human being, consider the information granulation method based on unsupervised clustering learning, construct the complex tasks decomposition mechanism based on multi-granularity and clustering method, and then we demonstrate the efficiency of our model in complex social networks. First, we must note that since there are countless and diverse kinds of complex tasks, our proposed method aims to explore the properties of discrete tasks. Our method consists of two stages: global tasks leading tree and multi-granularity task solving spaces.

3.1 Global tasks leading tree

As mentioned above, we first utilize leading tree idea based on DPC algorithm to construct a global tasks leading tree, which connects all nodes in the coarse grain layer. Additionally, the leading tree algorithm inherits the demerits derived from DPC, which leads to Domino effect, whereby if an incorrect center is chosen, classification error would occur among the whole neighbors with local densities less than the one. Fortunately, our proposed method minimizes the problem and has achieved high accuracy in experiments.

Since we prove our method in complex social networks and the discovery of latent communities, we show the process of detecting Dolphin network whose statement details are in the experimental section. Figure 2 displays the real social structure of Dolphin network. Before presenting the process, we improve the original DPC algorithm because it is used to cluster two-dimensional discrete points; it is improper to cluster the social network nodes whose edges represent link relationships.

In Eq. (2), the relative distance d_{ij} between node i and node j is calculated by Euclidean metric. Euclidean metric represents a two-dimensional distance of horizontal and vertical axes. In social complex networks, there are no distinct two-dimensional distances because the nodes indicate social members and the members connect to each other with certain relations. Therefore, we adapt the topology of social network to improve the computing of distance between node i and node j, which is defined as Eq. (4).

$$d_{ij} = \frac{\sqrt{\left|\Gamma(i)\right| \times \left|\Gamma(j)\right|}}{\Gamma(i) \bigcap \Gamma(j)} \tag{4}$$

where $\Gamma(i)$ and $\Gamma(j)$ represent neighbor sets of i and j, respectively, and neighbors are defined by edges between nodes. If i and j are not accessible to each other, their distance cost would be infinite. If they are accessible, but have no other common neighbor node, their distance cost would be 1. Additionally, if they are accessible and possess several mutual neighbor nodes, their distance would be less than 1 since they have more ways to contact each other. We computed the local density by Eq. (2) and Eq. (3) and the relative

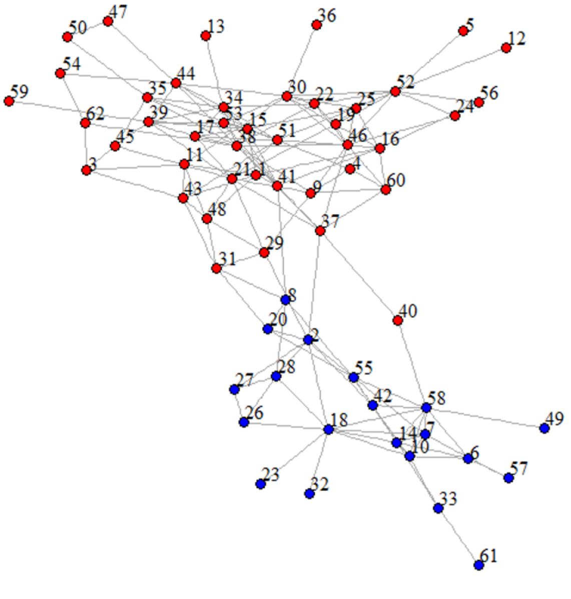


Fig. 2 The real social structure of Dolphin network.

distance of higher density node by Eq. (4). Then we selected the cutoff distance around 1% to 2%. We choose $d_c = 2\%$. Thereafter, we took note of each node i and their distance to higher local density node $Nneigh_i$. The node i will be the child and higher local density nodes $Nneigh_i$ will play their parents. The child node when linked to the parent node would construct a branch of leading tree structure. We recorded the meaningful intermediate results of DPC, as shown in Table 2, and presented the global tasks leading tree, as shown in Fig. 3.

Subsequently, we made some statements about the Table 2 and Fig. 3. In Table 2, for instance, when node $i = 1$, $\rho_i = 2.21$, and $Nneigh_i = 48$, this means that the local density of node $i = 1$ is 2.21 and the parent node of node $i = 1$ is node 48. In Fig. 3, the first statement is the root node 46, which is calculated by multiplying ρ by δ. If the node whose value $ord\ \gamma = \rho \times \delta$ is maximum, it is a likely core of the social network and regarded as the root of the global tasks leading tree. The leading tree is constituted by nodes set and their parents set. Afterwards, we generate the global tasks leading tree of Dolphin network and form the initial solving space in the coarsest grain layer.

Table 2 The intermediate results of DPC in Dolphin network.

i	δ_i	ρ_i	$Nneigh_i$
1	2.21	0.75	48
2	2.42	0.79	55
3	1.08	0.81	11
4	0.92	0.79	60
⋯	⋯	⋯	⋯
31	1.76	0.73	48
32	0.20	1.00	18
⋯	⋯	⋯	⋯
62	0.60	11.00	38

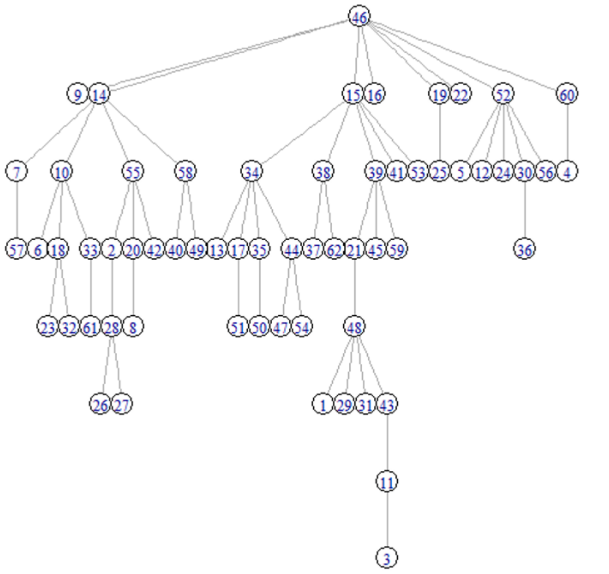


Fig. 3 The global task leading tree of Dolphin network.

3.2 Multi-granularity task solving space

In this stage, to find the optimal task solving space, we will refine the initial solving space of coarsest grain by the theory of multi-granularity. If we solve a problem in a fine grain layer, we can gain more efficiency and consume less time. Based on first stage, we utilize similarity of subtasks to partition the initial coarse grain layer into several fine grain layers, and then, estimate the optimal task solving layer through the rules of optimizing of solving layers. Additionally, the steps of our method, the MrGDM (Multi-granularity decomposition mechanism of complex tasks based on density peaks), are described in Algorithm 1. Our proposed method contains three aspects which are arranged in the following sequence.

3.2.1 Initial subtask centers set

Firstly, for initial subtask centers set, it consists of the latent core nodes which can organize other non-core nodes to be a small subtask entirety. Besides, what calls for special attention is that the initial subtask centers set is not the final subtask centers set. For DPC algorithm, in the decision graph, there is

Algorithm 1 MrGDM

Input: distance matrix D, cut off distance dc, number of redundant centers N, granulating threshold $thres$

Output: multi-granularity task solving space

```
use distance matrix $D$ and $dc$ to compute $\rho$ by Eq. (3), $\delta$ by Eq. (4);
construct leading tree according to $N$, $N$ comes from $\rho$ and $\delta$;
compute $ord\ \gamma = \rho \times \delta$, $CT = ord\gamma(N)$;
// granulating procedure
while $similarity < thres$ do
    for $i = 1$ to $sizeof(CT)$ do
        $SC(i)=find(Nn == CT(i))$;
        $similarity(i) = Similarity(SC(i), T - SC(i))$;
        if $similarity(i) < thres$ and max $|similarity(i) - thres|$ then
            $FCT(j) = i$;
            trigger granulation rule;
            remove $FCT(j)$ from $CT$;
return multi-granularity task solving space
```

some remarkable information for choosing the subtask centers set. For nodes set T, calculating the value $ord\ \gamma = \rho \times \delta$ and descending γ_i, which is shown in Table 3. The nodes more closer to $ord\ \gamma_i$ would more likely to be the centers, because they possess the criterion of higher local density and relatively far distance according to DPC algorithm. Considering the problem of omitting any center, we select advisable and redundant center nodes from $ord\ \gamma_i$, as the initial subtask centers set CT.

3.2.2 Similarity measure of subtasks

After choosing advisable and redundant center tasks, the center tasks with their oriented subtask sets can be clearly observed from the global tasks leading tree. The oriented subtask nodes set SC can be obtained by traversing the nodes of subtask centers set in the global tasks leading tree.

$$SC(i) = \{t \in T | Nneigh(t) == CT(i),\ i \in (1, sizeof(CT))\} \tag{5}$$

We assume that the subtask nodes set can split from the global task leading tree. Computing the similarity between SC and $T - SC$, if their similarity is less than the user-specified threshold $thres$ which is set according to the actual situation, it means SC is not close or interconnected with the initial tasks T. And it illustrates that SC satisfies the condition to be an autocephalous subtask set. Therefore, by splitting the global tasks leading tree, several small and isolated subtrees would be generated. On the basis of splitting process, the solving space of the coarse granular layer will be segregated and form a solving space with fine granular layer which is composed by a series of polybasic and independent subtask sets. If the small subtask entireties have been processed altogether, the original complex task would be answered by synthesizing the answers of subtask entireties.

There are many algorithms to measure the similarity between objects, such as Euclidean metric, Cosine, and Jaccard in vector space model. We select the similarity measure method in Ref. [27], which is based on graph partitioning theory and fully considers relative interconnectivity and closeness, thereby manifesting great results in clustering.

Thus, the relative interconnectivity RI between SC and $T - SC$ is

$$RI(SC,\ T - SC) = \frac{2 \times |EC\ (SC,\ T - SC)\ |}{|EC\ (SC)\ | + |EC(T - SC)|} \tag{6}$$

where $EC(SC, T - SC)$ denotes the absolute interconnectivity between SC and $T - SC$ and represents the total weights of the edges that straddle the nodes in SC and $T - SC$. $EC(SC)$ and $EC(T - SC)$ represent the sum weight of the edges in SC and $T - SC$, respectively.

The relative closeness RC between SC and $T - SC$ is

$$RC\ (SC, T{-}SC) = \frac{\bar{S}EC\ (SC, T - SC)}{X_1 \bar{S}EC(SC) + X_2 \bar{S}EC(T{-}SC)} \tag{7}$$

and $X_1 = \frac{|SC|}{|SC| + |T - SC|}, X_2 = \frac{|T - SC|}{|SC| + |T - SC|}$. where RC is the normalization of RI . $\bar{S}EC(SC, T - SC)$ is the average weight of the edges that connect nodes between SC and $T - SC$. $\bar{S}EC(SC)$ and $\bar{S}EC(T - SC)$ denote the average weights of the edges that pertain to the min-cut bisector of SC and $T - SC$, respectively. Terms $|SC|$ and $|T - SC|$ are the numbers of nodes in SC and $T - SC$, respectively.

The similarity of SC and $T - SC$ is measured by

$$Similarity\ (SC, T - SC) = RI(SC, T - SC) \times RC(SC, T - SC)^{\alpha} \tag{8}$$

where the user-specified parameter α is to control the relative importance between RI and RC. If $\alpha = 1$, it means that relative interconnectivity and closeness have the same importance.

3.2.3 Optimizing grain layer of solving spaces

By calculating the similarities of task subsets generated by CT, we set a user-specific threshold $thres$ to control the progress of separating the initial coarse granular layer into fine granular layers. Here, we formalize some notions and rules for optimizing grain layer of solving spaces.

Definition 1 (*Final tasks center set*) Given a task T, a redundant center task set denoted by CT, $SC(i)$ is the subtask set of T. For each node of CT to calculate the $Similarity(i) = Similarity(SC(i), T - SC(i))$, select the nodes where $Similarity(i) < thres$, and then, pop up the node which occupies maximum $|Similarity(i) - thres|$ to belong to the final tasks center set FCT. When there is no node in CT that satisfies the condition $Similarity < thres$, the process of popping up center node is to terminate.

Definition 2 (*Granulating rule*) Within Definition 1, for each granulating operation, if there is a node popped up, granulate from on top coarse grain layer to under fine grain layer; granulation is to be terminated when

Table 3 The redundant centers *CT* of Dolphin network.

i	δ_i	ρ_i	$Nneigh_i$	$ord\ \gamma_i$
1	2.21	0.75	48	46
2	2.42	0.79	55	15
3	1.08	0.81	11	14
4	0.92	0.79	60	38
⋯	⋯	⋯	⋯	34
31	1.76	0.73	48	52
32	0.20	1	18	⋯
⋯	⋯	⋯	⋯	⋯
62	0.61	1	38	61

center popping terminates.

In Definition 1, the paramount idea is that for each granulation the most likely center of subtask will be popped up, as conforms to the ideas of priority of majority. For every processing of popping, it is the processing of task decomposition; thus, we can obtain an accurate center of subtask. Via the method in Definition 2, we get a precise fine grain layer of solving space, since the terminal optimal decomposition of task has been captured. Our method can pop up the potential subtask centers and automatically generate the optimal fine granularity task solving space where we can decompose the complex task into undemanding subtask sets.

In Table 3, $ord\ \gamma$ descends from $\gamma = \rho \times \delta$. As we mentioned in Algorithm 1, we select the first to sixth nodes in $ord\ \gamma$ as advisable and redundant centers, so $CT = \{46, 15, 14, 38, 34, 52\}$. According to Definition 1 and Definition 2, we explore the final optimal task solving space of Dolphin social network. We set $thres = 0.5$, because if the *Similarity* between two communities is less than 0.5, it means that they are less connected to each other, so we consider that segregating the two communities is meaningful and ponderable.

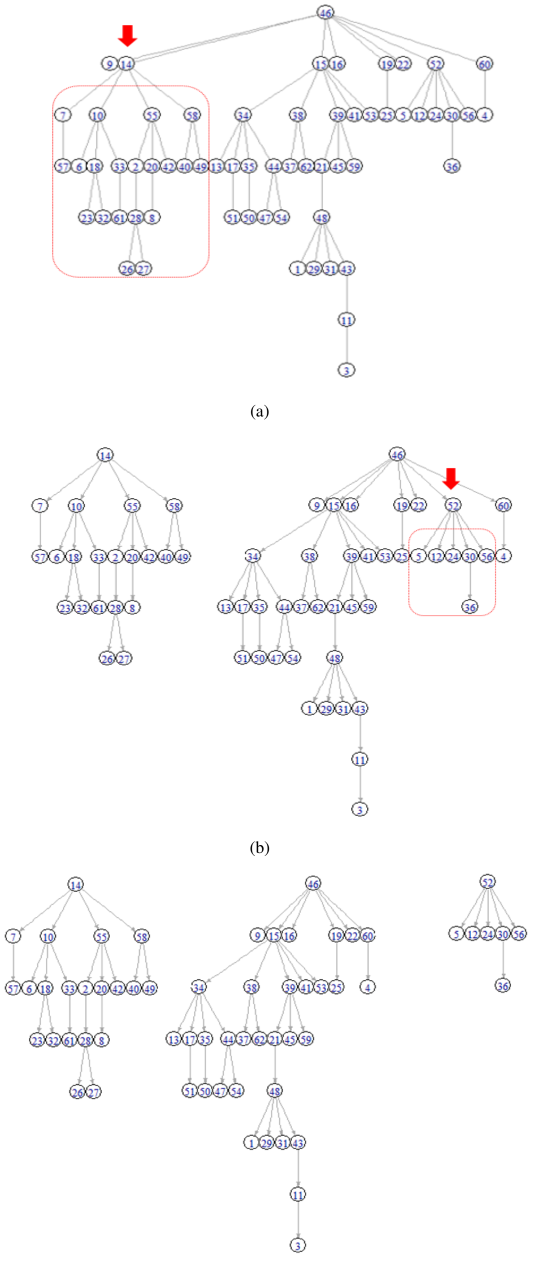
According to Definition 1, for the first round shown in Table 4, node 14 is the most likely center of a community. Node 14 and its child nodes should be split from the global leading tree, and then, the popped node 14 should be removed from redundant CT to FCT.

Table 4 Popping process for the first member of final center task set.

i	Similarity (i)
46	—
15	0.436
14	0.129
38	0.291
34	0.290
52	0.290

Next, consulting with Definition 2, we can granulate the initial coarse granular layer to a particular fine granular layer, which is shown in Figs. 4a and 4b. After the first popping, the terminal condition has not been triggered, so we continue executing the popping process until the terminal condition takes effect.

In the second process shown in Table 5, we can see



(a)

(b)

(c)

Fig. 4 (a) Global leading tree structure of Dolphin network; (b) Progress of the first granulation, where node 14 is one of the community center, so the tree splits two sub trees, and they are constituted as a fine granularity task solving layer; (c) Granulating process triggered by center node 52, and they form a fine granular solving layer with three subtask sets.

Table 5 Popping process for the second member of final center task set.

i	Similarity (*i*)
46	—
15	0.559
38	0.500
34	0.529
52	0.317
⋮	⋮

that the node 52 is popped up as the second latent community center, and an operation like first popping process is repeated until the terminal condition occurs. The granulating progress is presented in Figs. 4b and 4c.

In the third process shown in Table 6, the terminal condition has been triggered automatically, because there is no node whose similarity is less than threshold.

In the Dolphin network, we select the second solving space to reveal the communities detected by our method. Compared to the real structure of Dolphin network, the results obtained by method MrGDM can ultimately conform to real communities, this proves that the MrGDM is practicable and meaningful, and can provide decent validity simultaneously. Figure 5 shows MrGDM results.

4 Experiment

4.1 Compared methods

Table 7 briefly introduces the description of four state-of-the-art algorithms (such as ENBC[28], Local-T[29], CDERS[30], and LICOD[31]) for detecting community and describes the core concepts of each method.

4.2 Data sets

Dolphin network Dolphin social network is commonly used in testing a method for detecting community. Lusseau et al.[32] observed and recorded the contacts of a bottlenose Dolphin group in the New Zealand Doubtful Sound Fjord for 7 years. The group consists of two families and 159 edges of 62 nodes. Edges between the nodes represent frequent contact between two dolphins. The large family includes 42 members, while the small one contains only 20 members.

Table 6 Popping process for the third member of final center task set.

i	Similarity (*i*)
46	—
15	0.559
38	0.500
34	0.529
⋮	⋮

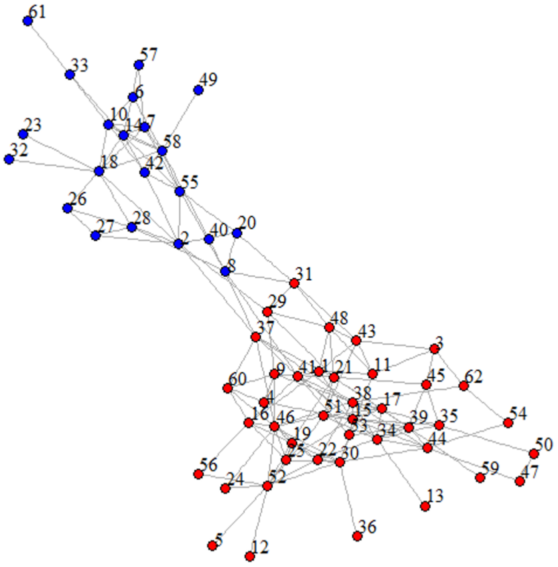


Fig. 5 The structure of Dolphin network by MrGDM.

Zarchary's karate club network Zachary social network is a complex network whose background is American University Karate Club in 1970s. Zachary[33] spent 3 years (1970–1972) to clarify the real structure of the club. The network consists of 34 nodes and 78 edges, where each node represents a member of the club. Links between the nodes represent two members often appear together in some occasions out of the club activities (such as karate training, club meeting, etc.), that is to say, out of the club, they can be called the friends. During the investigation, due to the dispute between the club director John A and the coach Mr. Hi, the club split into two small clubs. John A convened number 18 members, while Mr. Hi possessed the rest members.

NCAA college-football network The social network of the National Collegiate Athletic Association (NCAA) College-football[34] consists of the conferences of the football teams in American colleges, which contains 115 college football teams, 613 edges, and 12 conferences. The links between teams described that they played a football with each other in 2000s season. Additionally, the teams conducted more games in group phase than that in area phase; therefore, there were more links between the teams of a conference.

Table 7 Compared methods and their description.

Method	Description
ENBC[28]	By utilizing personal views and redefined community structure, this algorithm presents a notion of common interest in the relationship of social network. The method proposes two key measures reachability and isolability, where the reachability evaluates the ability of each node to reach out members of community and the isolability accounts for the ability of any community to isolate itself from the rest of the network.
Local-T[29]	By considering the number of internal and external triads, the main idea of this method is the T metric, which computes the relative quality of a community. The authors propose an intuitive statistical method based on the T metric, which can identify outlier and hub nodes within each discovered community.
CDERS[30]	The authors utilize an expanding ring search starting from the individual of interest and treat it as the seed node, and then according to their definition of a community, they iteratively contain the nodes at increasing number of hops from the seed user. If there is no further nodes can be appended, the iterative process is terminated. Furthermore, the social communities are organized by the list of added nodes.
LICOD[31]	This method adopts a leader-follower approach, whereby the leader nodes create social communities in which local communities can be calculated. The nodes whose degree is higher percentage compared to their neighbors would be selected as leader nodes. And then the leaders with a certain percent of common neighbors are considered a community. By computing the shortest distance of every node to the leader and considering the decision of neighbors, each node can be added to advisable community.

LFR datasets For exploring the effect of network configuration parameters on detecting community algorithm, this artificial dataset is proposed by Lancichinetti et al.[35] For the situation of our method, we generate two LFR benchmark graphs with a software package that can be downloaded from the website `http://santo.fortunato.goolepages.com`.

4.3 Evaluation

In this paper, to better evaluate the performance of the proposed method in comparison with other approaches, well-known measures are employed, such as NMI, F-measure, and Modularity measure (defined in Refs. [34, 36, 37], respectively). Additionally, index C in Table 8 includes the number of communities computed by these compared algorithms.

4.4 Comparison of results

Here, we compare the proposed approach, MrGDM, with other state-of-the-art methods such as CDERS, Local-T, LICOD, and ENBC by the selected datasets. The performance and efficiency of these algorithms have been presented in Table 8.

Firstly, in Karate club network, for the NMI and F-Score measures, ENBC and Local-T have the same

Table 8 Experimental applications of the compared methods for the selected datasets.

Dataset	Method	NMI	F-Score	Modularity (rank)	C
Karate				0.358	2
	ENBC	0.837	0.939	0.358	2
	Local-T	0.837	0.939	0.372	2
	LICOD	0.677	0.882	0.372	2
	CDERS	0.649	0.832	0.312	2
	MrGDM	1(1)	1(1)	0.371 (2)	2
Dolphin				0.360	2
	ENBC	0.530	0.671	0.491	3
	Local-T	0.434	0.605	0.510	4
	LICOD	0.442	0.471	0.494	7
	CDERS	0.627	0.890	0.374	2
	MrGDM	0.889 (1)	0.970 (1)	0.379 (2)	2
Football				0.551	12
	ENBC	0.906	0.810	0.592	10
	Local-T	0.864	0.722	0.545	10
	LICOD	0.900	0.850	0.603	11
	CDERS	0.762	0.644	0.634	12
	MrGDM	0.890 (3)	0.829 (2)	0.555 (1)	12
LFR1				0.784	15
	ENBC	0.992	0.992	0.781	15
	Local-T	0.992	0.992	0.781	15
	LICOD	0.965	0.922	0.777	13
	CDERS	0.922	0.867	0.634	15
	MrGDM	0.970 (3)	1 (1)	0.857 (4)	15
LFR2				0.859	30
	ENBC	0.998	0.997	0.858	30
	Local-T	0.777	0.504	0.706	19
	LICOD	0.997	0.995	0.853	31
	CDERS	0.942	0.913	0.780	30
	MrGDM	0.959 (3)	0.957 (3)	0.837 (3)	31

maximum when compared. Besides, in Table 8, we can see that the Modularity of MrGDM is 0.371 which is the second close to real Modularity 0.358 of social network, and MrGDM performs perfectly in NMI and F-Score. Secondly, as we can see, CDERS had the maximum NMI and F-Score and a great modularity in Dolphin network. For MrGDM, it has the better performance in the first two measures than that of CDERS with tiny difference in Modularity. Subsequently, in American football team network, MrGDM has a better performance regarding Modularity. However it fails to achieve desirable values for NMI and F-Score. LICOD and ENBC fail to detect either one or two of the communities. Finally, for the two LFR networks, MrGDM cannot keep prominent performance in these artificial datasets compared to the performance in real social networks, which is because the community detecting algorithm by MrGDM is based on the topology of real networks which considers the true meanings of links. Furthermore, the hierarchical notions of communities are well demonstrated naturally by multi-granularity theory. In artificial networks, it fails to consider the real hierarchical notions of communities since they are simple maps which only set up the number of nodes, edges, and degrees but ignore the true meanings of links. Thus, by assessing comprehensively, the community detecting method based on MrGDM generally performs better for the selected datasets in comparison to other approaches.

4.5 Complexity analysis

In the section, we present a comparative analysis of the complexity of computing MrGDM algorithm. The merits of MrGDM, which inherit from the original DPC algorithm, include its lucid simplicity, great accuracy, and strong adaptability in decomposing different complex. In the first stage of our method, we certified the distance measure of different complex tasks. Choosing an appropriate distance measure according the real structure of tasks is crucial for optimizing final optimal solving space. Selecting different distance matrixes leads to different calculating complexity. The original DPC algorithm chooses Euclidean matrix to compute the distance of two dimension data, which brings $O\left(n^2\right)$ time complexity, while, in social network, for instance, in the community detecting of Dolphin network, the distance matrix is evaluated by its topology structure and the time complexity approximately is $O\left(n^2\right)$, where n is the total nodes of social network. In the granulating stage, it generates the initial global tasks leading tree by one traverse of node set *Nneigh*, and costs $O(n)$ time complexity. Granulating the initial coarse solving space to fine grain solving spaces approximately costs $O\left(st\right)$, where s stands for the number of initial advisable and redundant initial center tasks and t is the cycle index of automatic algorithm termination. Therefore, the complexity of the whole algorithm is $O\left(n^2\right) + O(n) + O\left(st\right)$ approximately and it actually depends on the complexity of distance metric.

The complexities of other exiting algorithms are summarized in Table 9. Local-T, LICOD, and LeadF algorithms are more complex and far less accurate in identifying communities than MrGDM. SCAN is averagely complex, but is inaccurate in identifying communities. For ENBC, communities are identified very accurately, but it is time-consuming. Our algorithm, MrGDM, detects communities as accurately as ENBC, making these two algorithms the most accurate, which is their main strength. Moreover, MrGDM possesses a unique advantage over other algorithms in that it is hierarchical and comprehensible for every intermediate process of detecting communities, and as such, it is one of the best options for seeking accurate hierarchical communities. However, each algorithm has its own advantages and disadvantages.

Granular computing theory combined with MrGDM method improves DPC algorithm and explores a decomposition mechanism based on simulation of human cognitive process and clustering technique. It offers a new perspective to detect communities in social

Table 9 Summary of complexity of community detection algorithms.

Algorithm	Complexity	Remark
ENBC[28]	$O(n^2)$	Accurate communities, but still bound by cost
Local-T[29]	$O(n^3)$	Identifies outliers, but for local communities
CDERS[30]	$O(n^2)$	Accurate communities in small network
LICOD[31]	$O(n^3)$	Smaller and very inaccurate communities
SCAN[38]	$O(m)$	Identifies outliers, but inaccurate communities
LeadF[39]	$O(nm)$	Smaller communities in dense network
MrGDM	$O(n^2)$	Accurate communities, but for dense network

network, and it can quickly and accurately find stable communities with hierarchical structure.

5 Conclusion and Future Work

This paper proposes a complex task decomposition mechanism based on the GrC (Granular Computing) and DPC, which simulates the characteristics of human multi-granule thinking. We examine the applicability and veracity of our proposed method in complex social networks, where it performed better than several state-of-the-art approaches. Additionally, it seems almost inevitable that MrGDM has a lot of applications where it would yield desirable results. Furthermore, for future study, we will investigate further on the decomposition mechanism based on MrGDM and apply it to more complex tasks and investigate the time-consumption.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 61572091) and Chongqing Postgraduate Scientific Research and Innovation Project (No. CYB16106).

References

[1] J. R. Hobbs, Granularity, in *Readings in Qualitative Reasoning about Physical Systems*, D. S. Weld and J. De Kleer, eds. Elsevier, 1990, pp. 542–545.

[2] L. Zhang and B. Zhang, *Theory and Application of Problem Solving*. Beijing, China: Tsinghua University Press, 1990.

[3] J. S. R. Jang, ANFIS: Adaptive-network-based fuzzy inference system, *IEEE Trans. Syst., Man, and Cybern.*, vol. 23, no. 3, pp. 665–685, 1993.

[4] G. Y. Wang and H. B. Shi, TMLNN: Triple-valued or multiple-valued logic neural network, *IEEE Trans. Neural Netw.*, vol. 9, no. 6, pp. 1099–1117, 1998.

[5] L. Y. Zhu, G. W. Yuan, and Q. Du, An efficient explicit/implicit domain decomposition method for convection-diffusion equations, *Numer. Methods Partial Differ. Equ.*, vol. 26, no. 4, pp. 852–873, 2010.

[6] R. E. Jenkins and B. P. Yuhas, A simplified neural network solution through problem decomposition: The case of the truck backer-upper, *IEEE Trans. Neural Netw.*, vol. 4, no. 4, pp. 718–720, 1993.

[7] S. Thiria, C. Mejia, F. Badran, and M. Crépon, Multimodular architecture for remote sensing operations, in *Proc. 4^{th} Int. Conf. Neural Information Processing Systems*, Denver, CO, USA, 1991, pp. 675–682.

[8] R. Rifkin and A. Klautau, In defense of one-vs-all classification, *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, 2004.

[9] R. Perdisci, G. F. Gu, and W. Lee, Using an ensemble of one-class SVM classifiers to harden payload-based anomaly detection systems, in *Proc. 6^{th} Int. Conf. Data Mining*, Hong Kong, China, 2006, pp. 488–498.

[10] L. C. Cobo, C. L. Jr. Isbell, and A. L. Thomaz, Automatic task decomposition and state abstraction from demonstration, in *Proc. 11^{th} Int. Conf. Autonomous Agents and Multiagent Systems*, Valencia, Spain, 2012, pp. 483–490.

[11] J. Xu and G. Y. Wang, Leading tree in DPCLUS and its impact on building hierarchies, arXiv preprint arXiv: 1506.03879, 2015.

[12] A. Rodriguez and A. Laio, Clustering by fast search and find of density peaks, *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[13] K. Sun, X. R. Geng, and L. Y. Ji, Exemplar component analysis: A fast band selection method for hyperspectral imagery, *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 5, pp. 998–1002, 2015.

[14] Y. W. Chen, D. H. Lai, H. Qi, J. L. Wang, and J. X. Du, A new method to estimate ages of facial image for large database, *Multimed. Tools Appl.,* vol. 75, no. 5, pp. 2877–2895, 2016.

[15] H. Wu and Y. Wan, Clustering assisted fundamental matrix estimation, arXiv preprint arXiv: 1504.03409, 2015.

[16] K. M. Dean, L. M. Davis, J. L. Lubbeck, P. Manna, P. Friis, A. E. Palmer, and R. Jimenez, High-speed multiparameter photophysical analyses of fluorophore libraries, *Anal. Chem.*, vol. 87, no. 10, pp. 5026–5030, 2015.

[17] A. P. Zeng and Y. P. Huang, A text classification algorithm based on Rocchio and hierarchical clustering, in *Proc. 7^{th} Int. Conf. Advanced Intelligent Computing*, Zhengzhou, China, 2011, pp. 432–439.

[18] M. M. Wang, W. L. Zuo, and Y. Wang, An improved density peaks-based clustering method for social circle discovery in social networks, *Neurocomputing*, vol. 179, pp. 219–227, 2016.

[19] W. He and M. D. Xing, Classification method for POLSAR images based on find of density peak, (in Chinese), *Syst. Eng. Electron.*, vol. 38, no. 1, pp. 60–63, 2016.

[20] J. Xu, G. Y. Wang, and W. H. Deng, DenPEHC: Density peak based efficient hierarchical clustering, *Inf. Sci.*, vol. 373, pp. 200–218, 2016.

[21] G. Y. Wang, J. Xu, Q. H. Zhang, and Y. C Liu, Multi-granularity intelligent information processing, in *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, Y. Y. Yao, Q. H. Hu, H. Yu, and J. W. Grzymala-Busse, eds. Springer, 2015.

[22] S. L. Wang, D. K. Wang, C. Y. Li, Y. Li, and G. Y. Ding, Clustering by fast search and find of density peaks with data field, *Chin. J. Electron.*, vol. 25, no. 3, pp. 397–402, 2016.

[23] T. Qiu, K. F. Yang, C. Y. Li, and Y. J. Li, A physically inspired clustering algorithm: To evolve like particles, arXiv preprint arXiv: 1412.5902, 2014.

[24] M. L. Du, S. F. Ding, and H. J. Jia, Study on density peaks clustering based on k-nearest neighbors and principal component analysis, *Knowl.-Based Syst.*, vol. 99, pp. 135–145, 2016.

[25] X. H. Xu, Y. S. Ju, Y. L. Liang, and P. He, Manifold density peaks clustering algorithm, in *Proc. 3^{rd} Int. Conf. Advanced Cloud and Big Data*, Yangzhou, China, 2015, pp. 311–318.

[26] W. K. Zhang and J. Li, Extended fast search clustering algorithm: Widely density clusters, no density peaks, arXiv preprint arXiv: 1505.05610, 2015.

[27] G. Karypis, E. H. Han, and V. Kumar, Chameleon: Hierarchical clustering using dynamic modeling, *Computer*, vol. 32, no. 8, pp. 68–75, 1999.

[28] A. Biswas and B. Biswas, Investigating community structure in perspective of ego network, *Expert Syst. Appl.*, vol. 42, no. 20, pp. 6913–6934, 2015.

[29] J. Fagnan, O. Zaïane, and D. Barbosa, Using triads to identify local community structure in social networks, in *Proc. 2014 IEEE/ACM Int. Conf. Advances in Social Networks Analysis and Mining*, Beijing, China, 2014, pp. 108–112.

[30] K. H. Lim and A. Datta, A seed-centric community detection algorithm based on an expanding ring search, in *Proc. 1^{st} Australasian Web Conf.*, Adelaide, South Australia, 2013.

[31] Z. Yakoubi and R. Kanawati, LICOD: A leader-driven algorithm for community detection in complex networks, *Vietnam J. Comput. Sci.*, vol. 1, no. 4, pp. 241–256, 2014.

[32] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations, *Behav. Ecol. Sociobiol.*, vol. 54, no. 4, pp. 396–405, 2003.

[33] W. W. Zachary, An information flow model for conflict and fission in small groups, *J. Anthropol. Res.*, vol. 33, no. 4, pp. 452–473, 2015.

[34] M. Girvan and M. E. J. Newman, Community structure in social and biological networks, *Proc. Natl. Acad. Sci. USA*, vol. 99, no. 12, pp. 7821–7826, 2002.

[35] A. Lancichinetti, S. Fortunato, and F. Radicchi, Benchmark graphs for testing community detection algorithms, *Phys. Rev. E*, vol. 78, no. 4, 2008.

[36] N. X. Vinh, J. Epps, and J. Bailey, Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance, *J. Mach. Learn. Res.*, vol. 11, pp. 2837–2854, 2010.

[37] M. A. Tabarzad and A. Hamzeh, A heuristic local community detection method (HLCD), *Appl. Intell.*, vol. 46, no. 1, pp. 62–78, 2017.

[38] X. W. Xu, N. Yuruk, Z. D. Feng, and T. A. J. Schweiger, SCAN: A structural clustering algorithm for networks, in *Proc. 13^{th} ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, San Jose, CA, USA, 2007, pp. 824–833.

[39] D. Shah and T. Zaman, Community detection in networks: The leader-follower algorithm, arXiv preprint arXiv: 1011.0774 , 2010.

Ziling Pang is studying for a master's degree in Chongqing University of Posts and Telecommunications. She received the bachelor degree from Chongqing University of Posts and Telecommunication, China, in 2015. Her current research interests include granular computing, knowledge technology, and cognitive computing.

Guoyin Wang is a professor and PhD supervisor in Chongqing Key Laboratory of Computational Intelligence at Chongqing University of Posts and Telecommunications. He received the BS, MS, and PhD degrees from Xi'an Jiaotong University, China, in 1992, 1994, and 1996, respectively. His research interests include data mining, machine learning, rough set, granular computing, soft computing, cognitive computing, etc.

Jie Yang is a PhD candidate in Chongqing Key Laboratory of Computational Intelligence at Chongqing University of Posts and Telecommunications. Currently, he is also an associate professor with Zunyi Normal University. He received the BS and MS degrees from Hunan University, China, in 2009 and 2012, respectively. His current research interests include cloud model, granular computing, machine learning, and knowledge discovery.