A Novel Genetic Algorithm for Overlapping Community Detection

Yanan Cai, Chuan Shi, Yuxiao Dong, Qing Ke, and Bin Wu

Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia
Beijing University of Posts and Telecommunications, Beijing, China
{caiyanan,shichuan,dongyuxiao,keqing,wubin}@bupt.edu.cn

Abstract. There is a surge of community detection on complex network analysis in recent years, since communities often play special roles in the network systems. However, many community structures are overlapping in real word. For example, a professor collaborates with researchers in different fields. In this paper, we propose a novel algorithm to discover overlapping communities. Different from conventional algorithms based on node clustering, our algorithm is based on edge clustering. Since edges usually represent unique relations among nodes, edge clustering will discover groups of edges that have the same characteristics. Thus nodes naturally belong to multiple communities. The proposed algorithm apply a novel genetic algorithm to cluster on edges. A scalable encoding schema is designed and the number of communities can be automatically determined. Experiments on both artificial networks and real networks validate the effectiveness and efficiency of the algorithm.

Keywords: community detection, overlapping community, genetic algorithm, link community.

1 Introduction

Nowadays, community detection, as an effective way to reveal the relationship between structure and function of networks, has drawn lots of attention and been well developed. To do so, networks are modeled as graphs, where nodes represent objects and edges represent the interactions among them. Community detection divides a network into groups of nodes, where nodes are densely connected inside, while sparsely connected outside. However, in real world, objects often have multiple roles. For example, a professor collaborates with researchers in different fields, a person has his family group as well as friends group at the same time, etc. All of these objects represent the interaction between communities and then play an important role in the stability of the network. In community detection, these objects should be divided into multiple groups, which is known as overlapping. Overlapping community detection still remains a challenge in community detection.

Until now, lots of overlapping communities have been proposed, which can be roughly divided into two classes, node-based and link-based overlapping community detection algorithms. The node-based overlapping community detection

J. Tang et al. (Eds.): ADMA 2011, Part I, LNAI 7120, pp. 97–108, 2011.

algorithms [1,2,3,4,5,6,7,9], classify nodes of the network directly. The link-based algorithms cluster the edges of network, and map the final link communities to node communities by simply gather nodes incident to all edges within each link communities [8]. All of these algorithms contribute to overlapping community detection, however, they still have disadvantages. For example, the coverage of CPM [2] largely depends on the feature of the network , etc.

In this paper, we propose a genetic algorithm to detect overlapping community with link clustering, which is named Genetic algorithm for overlapping Community Detection (GaoCD). The algorithm first finds the link communities by optimizing objective function partition density D [8], and then map the link communities to node communities based on a novel genotype representation method. The number of the communities found by GaoCD can be automatically determined, without any prior information. Experiments on both artificial networks and real networks are designed to validate the algorithm. Experiment on artificial networks shows that GaoCD work well on networks with typical overlapping structure. Experiments on real networks compare GaoCD with ABL [8] and GA-NET+ [9]. It is shown that GaoCD always achieves higher partition density D and finds denser communities.

The paper is organized as follows. In the next section, we introduce the related works. Section 3 describes the new genetic algorithm we propose, including framework, objective function, genetic representation, and operators. The experimental results are illustrated in Section 4. Finally, Section 5 concludes the paper.

2 Related Work

Many algorithms have been developed to detect overlapping communities in complex networks, such as CPM [2], CONGA[5], GA-Net+ [9], etc. Among them, CPM is the most famous and widely used. However, CPM has a strict community definition and is not flexible enough for real network. When the network is too dense, CPM finds giant clique communities, however, when the network is too sparse, it finds no cliques at all. And thus, the coverage of CPM largely depends on the feature of the network, providing no global prospective for the whole network.

GA-Net+ [9], proposed by Pizzuti, first adopts genetic algorithm to detect overlapping communities. It proposes a method to transfer node graph to line graph, in which nodes present edges of the node graph, while edges present adjacent relationships of edges of node graph. The line graph is then used as the input of the genetic algorithm, and in each generation, the line graph is transferred to node graph to evaluate the fitness. After selection, the graph is transferred again for the next iteration of GA. The transfer between line graph and node graph costs much computation and decreases the effectiveness. GaoCD is also a genetic algorithm, but it clusters edges of the network, with different genotype, objective function and operators. Instead of community score [9], GaoCD adopts partition density D to evaluate the quality of the partition.

What's more, the partition found by GaoCD coverages the whole network and provides a global view of the structure of the network.

Recently, link based methods are proposed to detect overlapping communities. Based on the thought that each edge plays an unique role in the network, Ahn, Bagrow and Lehmann [8] first propose a link-based algorithm, clustering the edges of the network. They define the similarity of edges and an evaluation function for link community, partition density D. The algorithm first calculates the similarity of all edges of the network and assign each edge to its own community. At each step, the method chooses pairs of edges with the largest similarity and merges their respective communities until all edges belong to a single cluster. Then, the history of the clustering process is stored in a dendrogram, and the partition with the largest partition density D is chosen as the final result. As shown is section 4, this algorithm tends to find small communities and can not provide global view of the structure of the network.

There are other algorithms for overlapping community detection,such that the SCP of Kumpula [10], Lancichinetti's algorithm [7], etc. All of them need prior information, or have coverage problem, or suffer of efficiency.

3 The New Genetic Algorithm for Community Detection

In this section, we discuss our algorithm in detail,including the framework of the algorithm, objective function, the crucial genetic representation and operators.

3.1 Framework of the Algorithm

Genetic algorithm , derived from evolutionary biology, is a searching technique to find exact or approximate solutions for optimization problems. The GA algorithms are implemented as computer simulation, in which a population of abstract representations of candidate solutions to an optimization problem evolves towards approximate solutions, based on the production and selection schema. The framework of GaoCD is described in Algorithm 1.

To effectively apply genetic algorithm to solve overlapping community detection problem, we design a new kind of genetic representation, encoding the edges of the network and a specific decoding schema taking the edge feature into consideration. The genetic representation and operation designed effectively reduce the search space and thus improve the searching effectiveness.

3.2 Objective Function

GaoCD is a link-based algorithm, which finds link communities. For this reason, the novel genetic algorithm chooses link community evaluation function, partition density D, as the objective function. Partition density D is raised by Ahn in [8], evaluating the link density within the community, as described in Equation (1).

$$D(c) = \frac{m_c - (n_c - 1)}{\frac{n_c(n_c-1)}{2} - (n_c - 1)} \tag{1}$$

Algorithm 1. Main framework of GaoCD

procedure GAOCD($size, gens, p_c, p_m$)
  //$size$ is the size of the population.
  //$gens$ is the running generation.
  //p_c and p_m are the ratio of crossover and mutation, respectively, with $p_c \in [0, 1]$, $p_m \in [0, 1]$ and $p_c + p_m = 1$.
  $P_t = \Phi$
  for each i in 1 to $size$ **do**
    $g_i = generate\_individual()$
    $evaluate(g_i)$; $P_t = P_t \cup \{g_i\}$
  end for
  for each $gen \leftarrow t$ in 1 to $gens$ **do**
    $i = 0$; $P_{t+1} = \Phi$
    while $i < size$ **do**
      randomly select two individuals from P_t, g_j and g_k, $j, k \in [1, size]$
      generate random value $r \in [0, 1]$
      if $r < p_c$ **then**
        $g_j^{'}, g_k^{'} = crossover(g_j, g_k)$
      else $g_j^{'} = mutate(g_j)$; $g_k^{'} = mutate(g_k)$
      end if
      $i = i + 2$
      $evaluate(g_j^{'})$; $P_{t+1} = P_{t+1} \cup \{g_j^{'}\}$
      $evaluate(g_k^{'})$; $P_{t+1} = P_{t+1} \cup \{g_k^{'}\}$
    end while
    $selection(P_{t+1}, P_t \cup P_{t+1})$
  end for
  return $P[1]$
end procedure

$generate\_individual()$ //initialize an individual according to the genetic representation schema.

$evaluate(g)$ //evaluate the fitness of g individual according to objective function partition density D.

$crossover(g_j, g_k)$ //crossover genetic operator.

$mutate(g_j)$ //mutation genetic operator.

$selection(P_{t+1}, P_t \cup P_{t+1})$ //The selection step of the genetic algorithm. First select $size$ individuals with maximum fitness from $P_t \cup P_{t+1}$,and then fill in P_{t+1} one by one in decreasing order according to fitness value.

Define $P = \{P_1, \cdots, P_C\}$ as a partition of the network's links into C subsets. $m_c = |P_c|$ is the number of links in subset c. $n_c = |U_{e_{ij} \in P_c}\{i, j\}|$ represents the number of nodes incident to links in subset c. D_c refers to the link density of subset c. The partition density D is the average of D_c over all communities, weighted by the fraction of links present in each:

$$D = \frac{2}{M} \sum_c m_c \frac{m_c - (n_c - 1)}{(n_c - 2)(n_c - 1)} \tag{2}$$

As we can see that partition density D only considers the link density within the community, different from the common community definition that a community should be densely intra-connected and sparsely connected with the rest communities. For overlapping communities, this definition make no sense, as Fig. 1 shows. In this figure, there are three obvious communities, all of which are cliques. Because all of the communities are overlapping, with node 0 as common node, each community is densely intra-connected, while not sparsely connected with other communities.

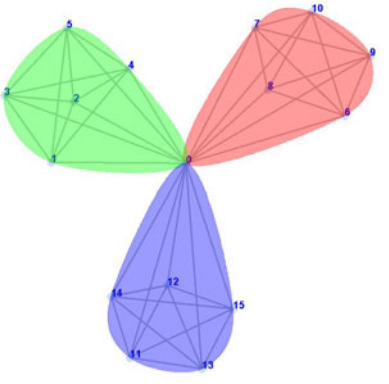


Fig. 1. A classical network containing overlapping communities

3.3 Genetic Representation

In this section, we describe the genetic representation in detail, including the encoding and decoding phase.

Encoding Phase. For those genetic algorithms for community detection which encoding the nodes of the network, we refer to them as node-based, such as GACD [11], GA-Net+ [9]. Different from the node-based genotype, GaoCD encodes the links of the network. In this link-based representation, an individual g of the population consists of m genes $\{g_0, g_1, \cdots, g_i, \cdots, g_{m-1}\}$,where $i \in \{0, \cdots, m-1\}$ is the identifier of edges, m is the number of edges, and each g_i can take one of the adjacent edges of edge i. According to graph theory, two edges are adjacent if they share one node in undirected graph. For example, in Fig. 2 (a), edge 0 has four adjacent edges, 1, 5, 2, 6, and one possible value for g_0 is 1, as shown in Fig. 2 (b).

Decoding Phase. The decoding phase transfers an genotype to partition, which consists of link communities. Gene g_i of the genotype and it's value j is interpreted that edge i and edge j have one node in common, and should be classified to same component. In the decoding phase, all components of edges are

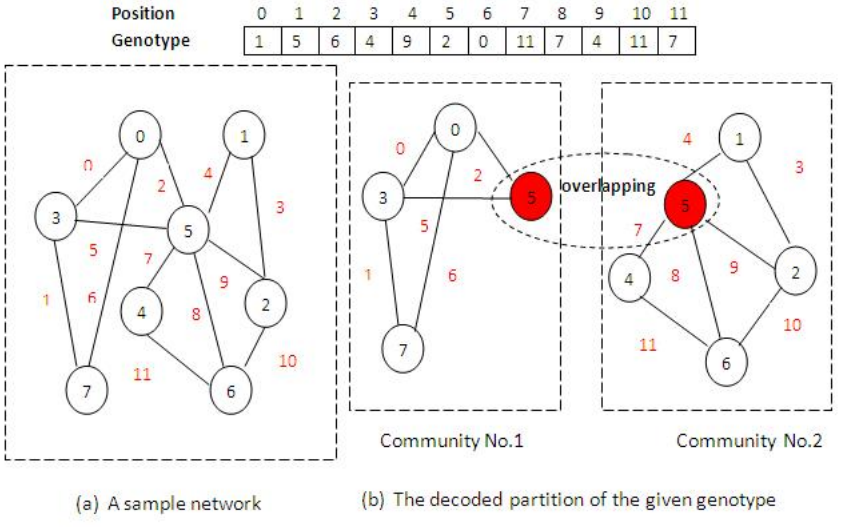


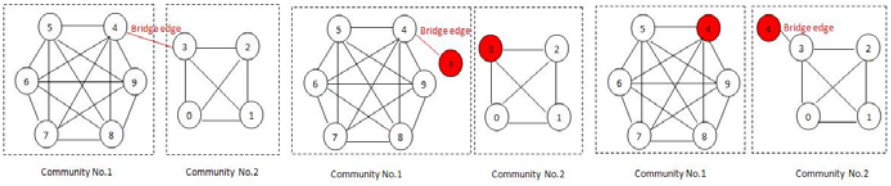
Fig. 2. Illustration of the genetic representation;(a) A simple network for encoding;(b) A possible genotype for the network in (a) and the corresponding decoded partition

found, and all edges within the component constitute a link community. According to link-based algorithm, by Ahn [8], overlapping communities are contained simply by gathering the nodes incident to all edges for each link community. However, it is not suitable for our algorithm, which restrain that each link community contains more than one link for the purpose of full coverage. We raise a fine tuning schema to deal with a special kind of edges, which is called **Bridge Edge**.

Bridge Edge is defined as the edge connecting two communities, as Fig. 3 (a) edge $(3, 4)$ shows. It is obvious that Fig. 3 contains two absolute communities, both of which are cliques. Because the bridge edge must belong to one unique link community, then the bridge edge could to classified to any of the two cliques, as shown in Fig. 3 (b) and (c). By simply gathering the nodes incident to edges of link community to form node community, we obtain overlapping communities shown in Fig. 3 (b) and (c) respectively, which obviously opposite to our purpose. To avoid this problem, we raise fine tuning method.

Fine tuning adjust the node membership of node community obtained by simple mapping schema. It is designed for nodes which have multi membership. The method first finds the list of nodes which have multi membership, and then for each node i in the list, it tests that whether node i contributes to the communities $c_{i1}, c_{i2}, \cdots, c_{in}$ by adding to them, where c_{ij} is community containing node i. Here, we adopt average degree of the community to evaluate the contribution. The definition of average degree is as follows:

$$AD(c) = 2 * \frac{|E(c)|}{|c|} \tag{3}$$



(a) reasonable partition (b) one possible unreasonable partition (c) the other possible unreasonable partition

Fig. 3. Illustration of bridge edge problem(edge of red color represents bridge edge);(a) A sample network containing bridge edge $(3, 4)$; (b) A possible partition found by the simple mapping schema from link community to node community, with node 3 classified to both communities. (c) The other possible partition found by the simple mapping schema from link community to node community, with node 4 classified to both communities.

where c is a community, $E(c)$ is the number of edges in the community, and $|c|$ is the number of nodes of the community.If adding to the community makes $|AD(c)|$ increase, we suppose that the node contributes to the community. If node i contributes to community c_{ij}, then the community stays the same(note that the community contains the node originally), otherwise, the node is removed from the community. If average degrees of all the n communities the node belongs to decrease with the node in, the least decreased community stay same, while others remove the node from the community. In Fig. 3 (b), node 3 is overlapping and contained in community No.1 and community No.2. Because node 3 decreases the $|AD(c)|$ of community No.1, while increases the $|AD(c)|$ of community No.2, then node 3 should be removed from community No.1, which is the most reasonable partition. In Fig. 3 (c), node 4 decreases the $|AD(c)|$ of community No.2, while increases the $|AD(c)|$ of community No.1, and then node 4 would be removed from community No.2.

3.4 Operators

In GaoCD, we assume that the network is a simple undirected connected graph. According to the genetic representation, we further raise the corresponding operators. Both the crossover and mutation ensure that the edge corresponding to gene i is incident to edge i. Suppose that we have two genotypes $g1$ and $g2$, and in the crossover phase, a random value i is generated. If the value of the ith gene of them are j and k respectively, through crossover, the value of ith gene exchanged, which makes the ith gene of genotype $g1$ is k, and the ith gene of genotype $g2$ is j. Because ith edge is incident to both edge j and edge k according to the generation of $g1$ and $g2$, the crossover has no effect on the principle that the edge of gene i should be incident to the ith edge of the network. In the mutation phase, each random value is generated for each parent, please notice that the values may not equal. If the value generated is i for genotype $g1$, and

the ith edge is incident to edge $i_1, \cdots, i_n$, then the ith value of $g1$ is replaced by i_k, $k \in 1, \cdots, n$. Fig. 4 shows the operation of network in Fig. 2 (a).

3.5 Discussion

GA is a search technique for optimization problem. When applied to specific problem, it is essential to design a appropriate genotype, including the encoding schema and decoding schema. The encoding schema we design ensures the algorithm covers the whole network, leaving no isolated nodes. The decoding schema of GaoCD probably deal with the bridge edges of the network, which improves the accuracy. What's more, the encoding schema reduces the search space from $O(E^E)$ to $O(d^E)$, where d is the number of incident edges of each edge, E is the length of genotype, $d << E$. Reducing the search space makes GaoCD search the more accurate solution with less consuming time.

4 Experiments

In order to test the effectiveness of GaoCD, we design experiments on artificial networks and real networks respectively. The experiment on artificial networks evaluate the ability of GaoCD to discover the overlapping nodes on different kinds of networks. The experiment on real networks compares GaoCD with ABL [8] and GA-NET+ [9]. All of the experiments are carried out on a 2.66GHz and 2G RAM Pentium IV computer.

4.1 Experiments on Artificial Networks

We first use artificial networks to test the performance of our algorithm. Fig. 5 includes several artificial networks, each of which represents a type of network structure. Testing all these tiny networks provide a view of the complicated networks with similar structure. In Fig. 5, each color represents one community. We

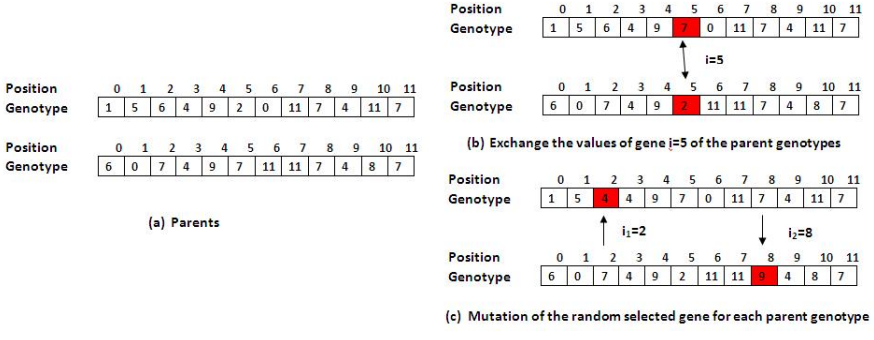
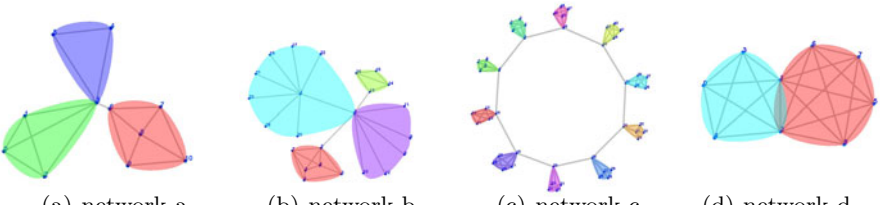


Fig. 4. Illustration of operators. (a) shows two genotypes of the population; (b) is one possible crossover of parent genotype of (a); (c) show a possible mutation for each parent genotype;

can see that network a and network c contain bridge edges which should not be divided into either of the communities in theory, GaoCD successfully distinguish the bridge edges and deal with them properly. Network b is a hierarchical core network, a node could be the core of a community, and at the same time, it belongs to another core community with another node as the core. In network d, GaoCD correctly divided the sharing nodes of the two cliques to both them. Overall, GaoCD correctly finds the overlapping communities for all kinds of networks, and ensures all nodes of the networks are covered in the partition, leaving no absolute nodes.



(a) network a (b) network b (c) network c (d) network d

Fig. 5. Four typical kinds of networks

4.2 Experiments on Real Networks

In this section, we first validate GaoCD on real networks with partition density D as the evaluation function, compared with ABL and GA-NET+. And then, we investigate the partition found by GaoCD by analyzing the community sizes of the partition. At last, an intuitive view is given for the partition found by GaoCD.

As stated by Ahn [8], when overlap is pervasive, each community has many more external than internal connections, the common definition is not suitable. Here we adopt partition density D as the evaluation function, which only considers the link density inside the community. Here, we validate GaoCD on several common data sets, as described in Table 1.

Table 1. Real networks

	karate(N1)	polbooks(N2)	dolphins(N3)	football(N4)	lesmis(N5)
Nodes	34	105	62	115	77
Edges	78	441	159	613	254

As shown in Fig. 6, for all real networks, GaoCD has the highest partition density D, which means that the partition found by GaoCD is denser than ABL and GA-Net+ does. To further investigate the quality of the partition found by GaoCD, we analyze the community size distribution of the communities for all real networks. For communities with size two contains single link, representing bridge edges or isolated nodes, which do not contribute to partition density D. We classify the communities into three classes, the first contains communities

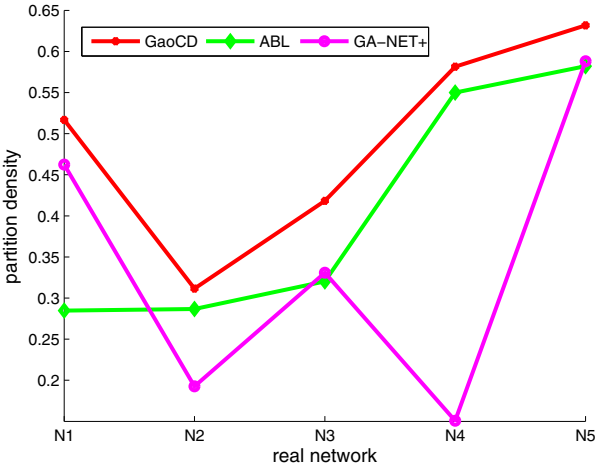


Fig. 6. Comparison of GaoCD, ABL, GA-Net+, relative to partition density D for real networks

with sizes varying from three to five, and second from six to ten, communities larger than ten belong to the third class. Fig. 7 shows the ratio of all three classes over all communities, respectively. It is easy to see that ABL tends to find small communities, with size from three to five. For almost all networks, GaoCD has larger values for ratio in middle and large classes. Overall, ABL tends to find tiny communities, and can not reflect the whole structure of the network. While GaoCD, on the contrast, finds denser communities in all sizes, capturing macrostructure as well as the microstructure of the network.

Fig. 8 show the partitions found by GaoCD. Fig. 8 (a) is the co-purchasing network of books about US politics by the online bookseller Amazon.com. Nodes represent books, and the edges betweens nodes represent frequent co-purchasing of books by the same buyers. It is obvious that the network constitutes of two large communities, with each of them surrounded by small ones. Node 6, node

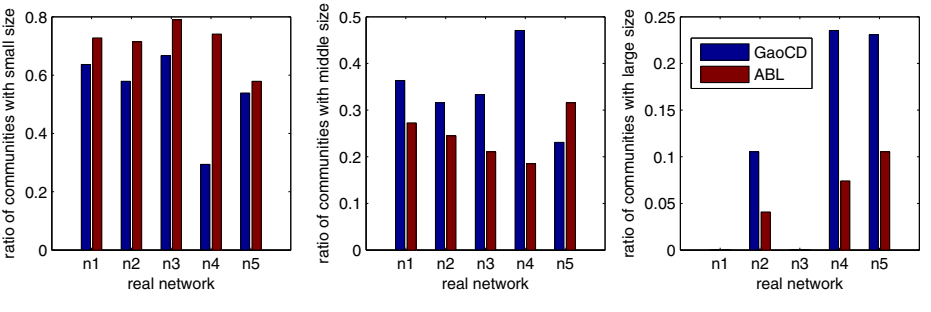
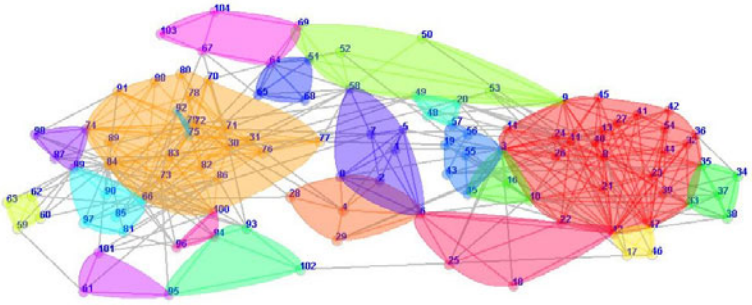
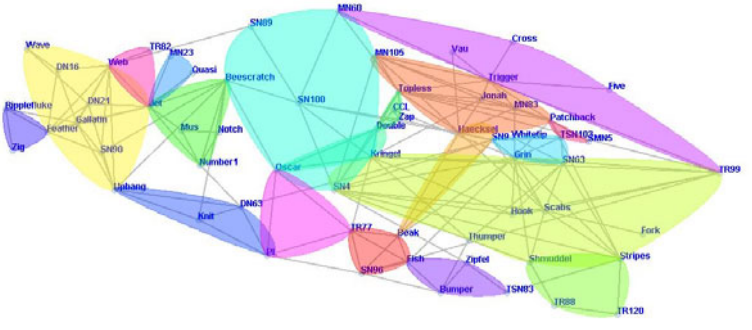


Fig. 7. Comparison of GaoCD, ABL, GA-Net+, relative to partition density D for real networks



(a) polbooks



(b) dolphins

Fig. 8. Partitions found by GaoCD.(a) is for network polbooks, and (b) for network dolphins.

58 and node 3 are nodes with obvious overlapping membership. Fig. 8 (b) is a social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand. The network fell into two parts because of the living of SN100. GaoCD successfully distinguish the special role of SN100, finding SN100 as the core of a community, connecting nodes from other different communities. Removing SN100, the core of the community, makes other nodes of the community unconnected, which then splits the networks.

5 Conclusion

In this paper, we propose a genetic algorithm for overlapping community detection, optimizing partition density D. Different from those node-based overlapping community detection algorithms, GaoCD utilizes the property of the unique role of links and applies a novel genetic algorithm to cluster on links. The genetic representation and the corresponding operators significantly reduce the search space and make the number of the communities determined automatically. Moreover, GaoCD covers all nodes of the networks, no matter the

network is dense or sparse. To validate our algorithm, experiments on artificial networks and real networks are carried out, respectively. Both of them show that GaoCD finds overlapping structure successfully. Compared with ABL and GA-Net+, GaoCD finds denser communities , which reflects the macrostructure as well as the microstructure of the network.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (Grant No.60905025, 90924029, 61074128).

References

1. Pereira, J.B., Enright, A.J., Ouzounis, C.A.: Detection of functional modules from protein interaction networks. Proteins: Structure, Fuctions, and Bioinformatics 54, 49–57 (2004)
2. Palla, G., Derenyi, I., Farkas, I., Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society. Nature 435, 814–818 (2005)
3. Baumes, J., Goldberg, M., Magdon-Ismail, M.: Efficient Identification of Overlapping Communities. In: Kantor, P., Muresan, G., Roberts, F., Zeng, D.D., Wang, F.-Y., Chen, H., Merkle, R.C. (eds.) ISI 2005. LNCS, vol. 3495, pp. 27–36. Springer, Heidelberg (2005)
4. Zhang, S.H., Wang, R.S., Zhang, X.S.: Identification of overlapping community structure in complex networks using fuzzy c-means clustering. Phisica A 374, 483–490 (2007)
5. Gregory, S.: An Algorithm to Find Overlapping Communities Structure in Networks. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) PKDD 2007. LNCS (LNAI), vol. 4702, pp. 91–102. Springer, Heidelberg (2007)
6. Gregory, S.: A fast algorithm to find overlapping communities in networks. In: PKDD, pp. 408–423 (2008)
7. Lancichinetti, A., Fortunato, S., Kertesz, J.: Detecting the overlapping and hierarchical community structure of complex networks (2008), arXiv:0802.1281, physics.soc-ph
8. Ahn, Y.Y., Bagrow, J.P., Lehmann, S.: Link communities reveal multiscale complexity in networks. Nature 466, 761–764 (2010)
9. Pizzuti, C.: Overlapping Community Detection in Complex Networks. ACM (2009)
10. Kumpula, J.M., et al.: Sequential algorithm for fast clique percolation. Phys. Rev. E 78, 026109 (2008)
11. Shi, C., Yan, Z.Y., Wang, Y., Cai, Y.N., Wu, B.: A Genetic Algorithm for Detecting Communities in Large-scale Complex Networks. ACS 13(1), 3–17 (2010)