Community Detection in Complex Networks Using Collaborative Evolutionary Algorithms

Anca Gog[1], D. Dumitrescu[2], and Béat Hirsbrunner[1]

[1] University of Fribourg, Department of Computer Science,
Bd. de Pérolles 90, 1700 Fribourg, Switzerland,
{anca.gog, beat.hirsbrunner}@unifr.ch

[2] Babes-Bolyai University, Department of Computer Science,
Kogalniceanu 1, 400084 Cluj-Napoca, Romania,
ddumitr@cs.ubbcluj.ro

Abstract. Scientific researchers from computer science, communication and as well from sociology and epidemiology reveal a strong interest in the study of networks. One important feature studied in complex network is the community structure. A new evolutionary technique for community detection in complex networks is proposed in this paper. The new algorithm is based on an information sharing mechanism between the individuals of a population. A real-world network is considered for numerical experiments.

1 Introduction

The study of complex networks intensively preoccupied the scientific community in the recent years. Examples of complex networks in nature and society include metabolic networks, the immune system, the brain, the human social networks, the Internet and the World Wide Web. A complex system is characterized by the lack of central control and the fact that individual components are simple compared to the collective behavior. The study of real-world networks revealed features as degree distribution, average distance between vertices, network transitivity [2], [11].

Another property that concerned scientific researchers is the community structure. A community in a network is a group of nodes densely connected but sparsely connected with the nodes belonging to other communities. The importance of community detection emerges from its many applications. For example, in social and biological networks it could help studying interactions between groups of people or animals, better understanding metabolic networks; this problem also arises in parallel computing, and the list could continue.

Many techniques have been proposed for identifying community structure in complex networks. In this paper, a new collaborative evolutionary algorithm for community detection is proposed. This algorithm is based on the collaboration between individuals that exchange information in order to accelerate the search process. The experimental results prove the efficiency of the proposed technique.

F. Almeida e Costa et al. (Eds.): ECAL 2007, LNAI 4648, pp. 886–894, 2007.

2 Existing Methods for Community Detection

An exhaustive description of all existing methods for community detection in complex networks is beyond the scope of this paper. Yet, an overview of some of the most known algorithms is presented in what follows.

Hierarchical (agglomerative and divisive) clustering [9] aims at discovering natural divisions of networks into groups based on metrics of similarity of strength of connection between vertices. Girvan and Newman [4] proposed a divisive algorithm that uses the edge betweenness as a weight measure of the edges. Radicchi et al. [8] proposed a similar technique but they used a new metric, edge-clustering coefficient whose computation time is less than the betweenness centrality. The resistor network approach proposed by Newman and Girvan in [7] has been improved by Huberman and Wu in [6]. In [7] is also proposed an algorithm for community detection using random walks. Balakrishnan and Deo proposed a new technique based on bibliometric similarity between all pairs of vertices in the graph [1]. Community detection using extremal optimization [3] uses the network modularity proposed in [7].

The drawback of these techniques is the computational complexity that makes them unsuitable for very large networks. Indeed, finding an exact solution for the community detection is believed to be an NP-complete problem and therefore difficult to solve.

Evolutionary computation provides promising algorithms for solving NP-hard problems. They provide good (acceptable) solutions for the problem in a reduced amount of time. Regarding evolutionary techniques applied for detecting community structure in complex networks, there is only one approach made by Tasgin and Bingol [10], as far as we know.

3 Collaborative Evolutionary Algorithms

Standard evolutionary algorithms are characterized by the lack of communication between individuals. Indeed, whether is the subject of selection, recombination or mutation, one individual does not know anything about the other individuals in the population or about the individuals that have contributed to its existence. In the proposed technique, the population imitates a social system where individuals communicate and share information.

One individual (or chromosome) encodes a potential solution of the problem and is composed by a set of elements called *genes*. Each gene can take multiple values called *alleles*. In the proposed collaborative evolutionary algorithm an individual has extra information that enables the sharing mechanism. On one hand, each individual knows the best potential solution already obtained in the search process (*GlobalOpt*). On the other hand, each individual is endowed with memory reverberated in the fact that it knows the value of its best ancestor (*LineOpt*). The ancestors represent all individuals that have existed in one of the previous generations and have contributed to the creation of the current individual: its parents, the parents of its parents, and so on. If within a single ancestral line there are multiple ancestors with the same best fitness values, the closest ancestor is chosen. If within the two ancestral lines of an individual the best individuals have identical fitness, one of them is randomly chosen. In the initial population, the *LineOpt* of each individual is the individual itself.

Both *GlobalOpt* and *LineOpt* will guide the search process in the form of passing relevant genetic material to the individuals. It is intended to show how this information can affect the search process for a problem whose solutions are encoded by discrete variables. This extra information that each individual has will affect the way selection is performed and the way recombination between individuals takes place.

3.1 Encoding and Population Model

A fixed size population is considered, the population size being a parameter of the algorithm. A potential solution of the problem (a chromosome) is a string of constant length $\{x_1, x_2, \ldots, x_n\}$ where n represents the number of nodes in the network and x_i represents the identifier of the cluster to which the node i belongs, $1 \leq i \leq n, 1 \leq x_i \leq n$.

Besides that, each individual retains the value of its best related individual and the value of the best individual obtained so far in the search process.

The initial population is randomly generated. Furthermore, a number of nodes are randomly selected and the neighbors of each node receive the same cluster identifier as the selected node. The neighbors of a node are considered to be all the nodes connected by edges with the current one.

3.2 Fitness Function

The potential solutions are evaluated by means of a real-valued function $f : X \rightarrow \mathbb{R}$, where X denotes the search space of the problem. The fitness of a chromosome tells how good a certain distribution of the nodes into clusters is, how well are detected the communities existing in the network.

In order to quantify the strength of a particular division of the network, the measure of quality proposed in [7] is considered. The *modularity* measure is defined by:

$$Q = \sum_i (e_{ii} - a_i^2),$$

where i is the index of the communities, e_{ii} is the proportion of edges that connect vertices in the community i and a_i is the proportion of edges with at least one node in the community i. The fitness function is to be maximized.

3.3 Collaborative Tournament Selection

The n individuals within the population $P(t)$ are grouped by their *LineOpt*. The clusters $A_1, \ldots, A_k, k \leq n$ are formed according to the rules:

(i) the clusters $A_1, \ldots, A_k, k \leq n$ represent a partition of *P(t)*:

$$(a) A_i \neq \phi, 1 \leq i \leq k,$$

$$(b) \bigcup_{i=1}^{k} A_i = P(t),$$

$$(c) A_i \cap A_j = \phi, 1 \leq i \leq k, 1 \leq j \leq k, i \neq j.$$

(ii) all the individuals that belong to the cluster $A_i (1 \leq i \leq k)$, have the same *LineOpt*:

$$\mathit{LineOpt}(x^u) = \mathit{LineOpt}(x^v),$$

$$\forall x^u, x^v \in A_i, 1 \leq i \leq k.$$

(iii) every two different clusters $A_i, A_j (1 \leq i \leq k, 1 \leq j \leq k, i \neq j)$ have a different *LineOpt*:

$$\mathit{LineOpt}(x^u) \neq \mathit{LineOpt}(x^v),$$

$$\forall x^u \in A_i, x^v \in A_j, 1 \leq i \leq k, 1 \leq j \leq k, i \neq j.$$

In order to preserve the exploration of the search space, two individuals are selected for being recombined only if they belong to different clusters. The aim is to recombine genetic material from individuals that are not genetically related, in order to expand the search to regions that have not been yet explored.

Once we have randomly chosen the two different clusters that will provide individuals for recombination, we choose an individual from each cluster according to a tournament scheme.

Let $A_i, 1 \leq i \leq k$, be one of the two clusters selected for recombination. A group of q $(1 \leq q \leq |A_i|)$ individuals is randomly chosen from the cluster. The sample members are chosen so as to induce a great diversity within the tournament group. The number q is the tournament size and can be different for each cluster, if we take into account the size of the cluster. For all numerical experiments presented in Section 4, the tournament size is set to:

$$q = \left\lceil \frac{|A_i|}{2} \right\rceil,$$

where for a real value k, we denote by $\lceil k \rceil$ the superior integer part of k,

$$k \leq \lceil k \rceil \leq k + 1.$$

This way the tournament size is proportional to the size of the cluster. The fittest individual from the tournament (or sample) group will be chosen as one of the two parents that will be recombined. The second parent is chosen from the second cluster in a similar way.

3.4 Collaborative Recombination Operator

Recombination operator performs an information exchange between chromosomes. This way the offspring obtained after the recombination of the parents will keep genetic information of both parents. Recombination helps the progress of the search by exploring the search space.

A variant of the collaborative recombination operator for permutation based encoding has been proposed in [5]. The recombination operator uses the information encoded by the *GlobalOpt* and the *LineOpt* of each individual. This way it is not only transferring to the offspring genetic information from the parents but from the best ancestor and from the best global as well.

The main idea of this operator is that if a certain individual's *LineOpt* contains genetic material that can also be retrieved in the *GlobalOpt*, than that genetic material is considered as being good for the search process. For our specific problem, relevant genetic material refers to the fact that a certain node belongs to a certain cluster. One important feature of the proposed recombination is the control of the amount of relevant genetic information transferred from the *GlobalOpt* and *LineOpt* to the offspring. The control is made by taking into account the number of the current generation relative to the total number of generations and the number of common genes of *GlobalOpt* and *LineOpt*. The aim is to increase the diversity in the first stages of the algorithm and to become more goal-oriented in the final stages, by keeping in the configuration of the offspring more relevant genetic information from *LineOpt* and *GlobalOpt*.

In order to increase the population diversity during all stages of the search process, another characteristic of the collaborative recombination is that a randomly chosen sequence of one parent is always kept in the offspring.

3.5 Mutation Operator

The mutation operator is not affected by the *GlobalOpt* and *LineOpt* - as its main feature is to introduce diversity into the population of candidate solutions and to reintroduce lost genetic material into the population. Mutation remains responsible for exploring new promising regions of the search space and not to exploit those which already have been discovered.

By taking into account the architecture of the chromosome, mutation has the following features: one gene of the chromosome is randomly selected and assigned with a randomly chosen cluster identifier. Also, all the neighbors of the selected node will receive the same cluster identifier.

Mutation takes place with a certain probability, which is given as parameter of the algorithm.

3.6 Merging Operator

A new merging operator is considered in order to accelerate the search process. The merging operator is applied with a certain probability for each individual. Two genes having different values (e.g. two nodes belonging to different clusters) are randomly chosen. If the individual obtained by combining the two clusters has a better fitness than the original individual, than all the genes that have one of the two cluster identifiers will receive the same value.

3.7 Selection for Replacement and Survival

The best offspring obtained after recombination (possible subject to mutation and/or merging) is kept in the next generation. The elitism ensures that the fitness of the best solution in the population does not deteriorate as the generation advances. The algorithm ends after a certain number of generations that did not improve the best solution obtained so far.

4 Experimental Results

The Zachary's karate club network [12] is considered to test the efficiency of the proposed algorithm. The parameters of the algorithm are written in Table 1.

Table 1. Parameters of the Collaborative Evolutionary Algorithm

Population size	Number of nodes randomly chosen when initializing the population (see subsection 3.1)	Tournament size (see subsection 3.3)	Mutation rate	Number of generation after which the algorithm ends if no improvement is brought
100	$\frac{n}{10}$	$q = \frac{\lvert A_i \rvert}{2}$, $1 \leq i \leq k$	0.05	1000

This is a classic network used for social network analysis. It has 34 nodes and 78 edges that represent the social interactions between the members of a karate club at an American university, observed by Wayne Zachary in the early 1970s. Due to a dispute between the club's administrator and the principal teacher, the club split in two, forming one cluster around the teacher and one cluster around the administrator.

Therefore, the real community structure is formed by two clusters around the node 1 (the administrator) and node 33 (the teacher). Thought, if we understand the communities as subsets of vertices within which vertex-vertex connections are dense but between which connections are less dense, the real division of the karate club into two communities does not represent, by all means, the best modularity of the network. Indeed, a better modularity is obtained if there is a third community that contains only the node 10. This is the solution obtained by the proposed collaborative evolutionary algorithm in 80% of its 25 runs and is depicted in Figure 1. An even better modularity is obtained in 20% of the 25

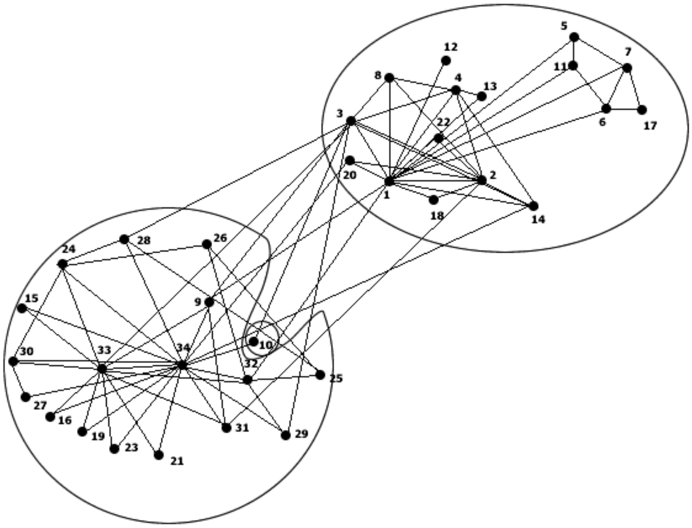

Fig. 1. Solution obtained by the proposed collaborative evolutionary algorithm in 80% of its 25 runs

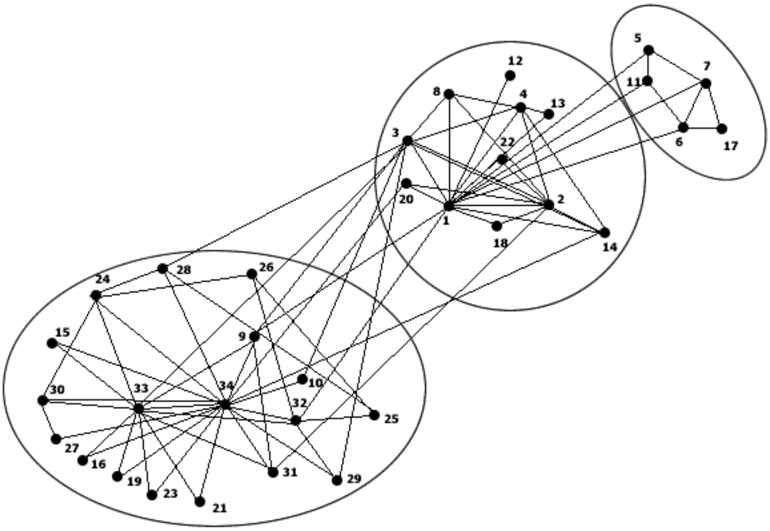

Fig. 2. Solution obtained by the proposed collaborative evolutionary algorithm in 20% of its 25 runs

runs which put the node 10 in the real cluster to which it belongs but create a third cluster that contains the nodes 5, 6, 7, 11 and 17 (see Figure 2).

Compared to the other evolutionary approach of community detection [10] which uses the same modularity as the fitness of the individuals, our proposed collaborative algorithm performs much better, as [10] reports a 97%-100% detection of only two clusters for the karate club network. Even if these two clusters represent the real communities formed in this network, they do not represent the best modularity, a much better modularity being detected by our proposed algorithm.

Moreover, the collaborative evolutionary algorithm finds the solution after about 1050 generations. By taking into account the fact that the algorithm ends after 1000 generations that did not improve the solution found so far (see Table 1), it means that the solution is detected after about 50 generations.

5 Conclusion

A new evolutionary technique for community detection in complex networks has been proposed. The collaborative evolutionary algorithm introduces into the population of individuals knowledge about the best solution obtained so far and about the best related chromosome of each individual. This extra knowledge affects the way recombination and selection are performed. A new merging operator is also proposed. The Zachary's karate club network is considered for numerical experiments. Results show an improvement of the solution obtained by the other evolutionary approach existing for community detection [10].

Acknowledgments. This work was supported by a Scholarship 2006-2007 of the University of Fribourg.

References

1. Balakrishnan, H., Deo, N.: Discovering Communities in Complex Networks. In: Proceedings of the ACM Southeast Regional Conference, pp. 280–285. ACM Press, New York (2006)
2. Barabasi, A.-L.: Linked: The New Science of Networks. Perseus, New York (2002)
3. Duch, J., Arenas, A.: Community Detection in Complex Networks using Extremal Optimization. Physical Review E 72, 027104 -1 (2005)
4. Girvan, M., Newman, M.E.J.: Community Structure in Social and Biological Networks. Proceedings of the National Academy of Sciences of the USA 99, 7821–7826 (2002)
5. Gog, A., Dumitrescu, D.: Adaptive Search in Evolutionary Combinatorial Optimization. In: Proceedings of the International Conference of Bio-Inspired Computing – Theory and Applications (BIC-TA), Wuhan, China, pp. 123–130 (2006)
6. Huberman, B.A., Wu, F.: Finding Communities in Linear Time: a Physics Approach. The European Physics Journal B 38, 331–338 (2004)
7. Newman, M.E.J., Girvan, M.: Finding and Evaluating Community Structure in Networks. Physical Review E 69, 026113-1 (2004)

8. Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D.: Defining and Identifying Communities in Networks. Proceedings of National Academy of Science in USA 101, 2658–2663 (2004)
9. Scott, J.: Social Network Analysis: A Handbook. Sage Publication, London (2000)
10. Tasgin, M., Bingol, H.: Community Detection in Complex Networks using Genetic Algorithm. cond-mat/0604419 (2006)
11. Watts, D.: Six degrees: The Science of a Connected Age. Gardner's Books, New York (2003)
12. Zachary, W.W.: An Information Flow Model for Conflict and Fission in Small Groups. Journal of Anthropological Research 33, 452–473 (1977)