EXTREMAL OPTIMIZATION AND NETWORK COMMUNITY STRUCTURE

Noémi Gaskó
Department of Computer Science, Babeş-Bolyai University, Cluj-Napoca, Romania
gaskonomi@cs.ubbcluj.ro

Rodica Ioana Lung
Department of Statistics, Forecasts, Mathematics
Babeş-Bolyai University, Cluj-Napoca, Romania
rodica.lung@econ.ubbcluj.ro

Mihai Alexandru Suciu
Department of Computer Science, Babeş-Bolyai University, Cluj-Napoca, Romania
mihai-suciu@cs.ubbcluj.ro

Abstract The network community structure detection problem has been recently approached with several variants of an extremal optimization algorithm. An extremal optimization algorithm is a stochastic local search method that evolves pairs of individuals that can be represented as having several components by randomly replacing components having worst fitnesses. The number of components to be replaced in one iteration influences both the exploitation and exploration capabilities of the method; an efficient method of adjusting this number during the search may significantly influence the quality of results. In this paper we explore the use of several updating mechanisms for this number. Numerical experiments are used evaluate them and also to compare results obtained with those provided by other state-of-art methods.

Keywords: Community structure detection, Extremal optimization.

1. Introduction

The network community structure detection problem has recently attracted a lot of attention from the heuristic community because both its large applicability and challenging nature. A particular challenge associated with this problem arises from the lack of a formal definition for the concept of community, and subsequently that of community structure [6]. Apart from classical definitions that attempt to characterize communities by using various network measures, a relatively recent class of approaches define the community structure as the optimum value of a certain fitness function that is supposed to illustrate the modularity of the network. Alas, it is also accepted that such an ideal function has not yet been proposed; existing attempts can only be validated by means of numerical experiments and, while some functions prove suitable for synthetic benchmarks, most of them fail when tested on real-world networks for which the community structure is not so well-defined. Moreover, the effectiveness of using a certain fitness function depends also on the underlying method used to compute the optimum value.

In this paper we investigate the behavior of an extremal optimization algorithm designed to optimize the modularity function [13], combined with the community fitness [8], when using four different manners of updating the number of nodes to be changed in one iteration: the classic variant of changing only one node [3, 4], the improved τEO [2], and the more recent variants in which this number decreases exponentially [18], or linearly [11].

2. Network Community Structure

The fact that the community structure detection problem can be reformulated as an optimization problem, makes it approachable by stochastic search methods, benefiting from their scalability and adaptability. Given an undirected, unweighted graph $G = (N, E)$, where N is the set of nodes, or vertices, and E is the set of edges/links, a community structure is described intuitively as a partition over the set of nodes such that nodes within each set are more connected to each other than to the other sets in the partition.

While this intuitive definition appears to be easily formalized, by considering either that a community is a group of nodes such that for each one the number of links within the community is greater than the number of links connecting it to the outside (the strong community concept [15]) or even that the total number of links connecting nodes inside the community is greater than the number of links to the outside (the weak community concept [15]), there are many counterexamples of networks with known community structure that do not satisfy either definition. In fact, there does not exist a definition that formalizes the intuitive description above and be accepted as valid for most situations.

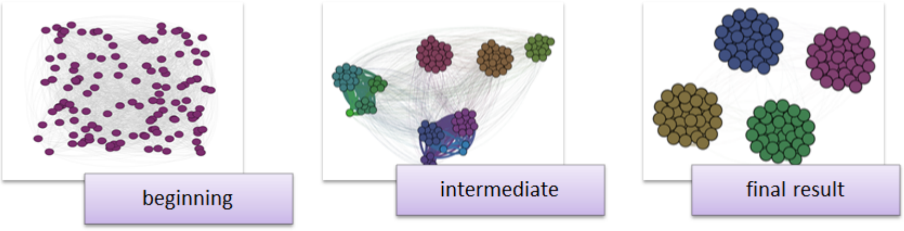

Figure 1: An example of a solution detected by an heuristic on a network with 124 nodes and 4 communities.

In spite of this, or maybe because of it, alternate methods to define the community structure have been proposed. One of the most popular one, from a computational point of view, is to use a function that has as an optimum value the real community structure of the network. Again, while such a function that may be used for all possible networks does not exist, there are some that are more effective and that became popular in approaching this problem in the last years. Examples are the modularity Q [13], the modularity density [10], the community score [14], and the community fitness [8]. These functions, and most of all the modularity and the modularity density, have been widely used and studied in conjunction with various heuristics designed for their direct or indirect optimization, i.e., directly finding their optimum and consider it as the community structure, or only including them in one or more search phases. An example of what is expected from a community structure detection algorithm is depicted in Fig. 1.

The modularity Q of a community structure is defined as

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(C_i, C_j), \tag{1}$$

where the sum runs over all pairs of vertices i and j, A is the adjacency matrix, m the total number of links in the network, k_i the degree of node i, C_i the community of node i and $\delta(C_i, C_j)$ equals 1 if nodes i and j belong to the same community and 0 otherwise. When two community structures are compared, a higher modularity value presumably indicates a better solution.

3. Extremal Optimization

Recently, a new heuristic approach that combines the modularity and community fitness and uses extremal optimization algorithm (EO) [2] as an underlying method has been proposed. Validated by means of numerical experiments, this approach has proven more efficient than other state-of-art methods when tested on usual benchmarks. With the purpose of improving the extremal optimization method, several other EO variants have been proposed, each one of them apparently leading to better results. This article compares these methods, in the context of the community structure detection problem, in the attempt to assess if there are significant differences among them and if so, if one of them may prove more efficient than the others.

The baseline method used here is the NoisyEO algorithm [11] which is described in Algorithms 1 and 2. A typical EO algorithm evolves a pair of individuals (s, s_{best}): s explores the search space and s_{best} preserves the best solution found by s. NoisyEO evolves a population of such pairs of individuals representing possible structures and evaluated with the modularity function. Also typical to EO is the fact that one individual is represented as a set of components with different fitnesses; during one iteration the component having the worst fitness value is randomly replaced. A more efficient variant, called $\tau-$EO, uses a probability distribution to decide which nodes are changed [2].

NoisyEO considers nodes as the components, and computes for each node a fitness function as the node's contribution to its community, i.e.:

$$f_i^{(node)}(C_1, \ldots, C_n) = f(C_i) - f(C_i \backslash \{i\}), \tag{2}$$

where C_i represents the community of player i, $s \in S$, and $C_i \backslash \{i\}$ is the same community without node i; f is the community score:

$$f(C) = \frac{k_{in}(C)}{(k_{in}(C) + k_{out}(C))^{\alpha}}, \tag{3}$$

$k_{in}(C)$ is the double of the number of internal links in community C; $k_{out}(C)$ the number of external links of C; and α - a parameter that controls the community size (in experiments presented in this paper $\alpha = 1$). Thus, a NoisyEO individual is a vector of length equal to the number of nodes, of the form $s = (C_1, \ldots C_n)$, where C_i is the community of node i.

Within NoisyEO several components are replaced simultaneously: their number starts from approximatively 10% of the number of nodes, linearly decreases until the middle of the search and after that it remains constant, equal to 1. But there are also other several ways this number can be changed, based on the intuition that larger values induce diversity and intensifies exploration of the space, while smaller values permit better exploitation. In this paper we explore the possibility of using

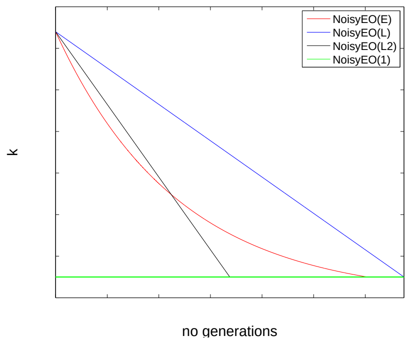

Figure 2: Different variants of updating the number of nodes κ that are changed each iteration during the search in order to maintain the equilibrium between exploration - at the beginning of the search - and exploitation - towards the end of the search.

other methods of adapting this number in an attempt to find the best version of EO suitable for the community structure detection problem. The following variants, all based on NoisyEO, are proposed:

- NoisyEO(L2) - κ decreases linearly to 1 until the middle of the search and remains 1 to the end [11];

- NoisyEO(L) - κ decreases linearly to 1;

- NoisyEO(E) - κ decreases exponentially from approx 10% of the number of nodes, to 1 at the end of the search:

$$\kappa_{NrGgen} = \max\left\{1, \left[\frac{1}{10} \cdot N \cdot (N-2)^{-\frac{NrGen}{MaxGen}}\right]\right\}, \tag{4}$$

  where $[\cdot]$ represents the integer part, N the number of network nodes, and $MaxGen$ the maximum number of generations/iterations allowed.

- NoisyEO(1) $\kappa = 1$, constant;

- $\tau-$EO, that uses the framework of NoisyEO with a $\tau-$EO iteration, in which nodes are ranked by fitness and the probability of choosing node r is $P(r) \propto r^{-\tau}$ [5].

A graphical representation of the four different variants of setting κ is illustrated in Fig. 2.

Algorithm 1 *NoisyEO* algorithm

Parameters:

- Population size - $popsize$;
- Probability of shift - p_{shift};
- Number of generations between switching networks - G;
- Total number of shifts - $NrShifts$;
- Expected minimum and maximum number of communities.

```
Randomly initialize $popsize$ pairs of configurations $(s, s_{best})$.
$noise$=false;
repeat
    if $noise$ then
        Induce noise with probability $p_{shift}^{(*)}$;
        Randomly reinitialize each $s_{best}$ in population;
    else
        perform search on the original network;
    end if
    $noise$=not $noise$;
    Update $k$ depending on the tested EO variant$^{(**)}$;
    for $G$ generations do
        Apply $\kappa$EO $(s, s_{best})$ for all pairs $(s, s_{best})$ - Alg. 2;
    end for
until $G * NrShifts >$ Maximum number of generations;
Return $s_{best}$ with highest fitness.
```

$^{(*)}$ Modify network by randomly deleting/adding nodes with probability p_{shift} which decreases linearly from an initial value to 0 during the search.

$^{(**)}$ One of the following variants are considered:

- NoisyEO(E) - κ decreases exponentially, eq. (4);
- NoisyEO(L) - κ decreases linearly to 1;
- NoisyEO(L2) - κ decreases linearly to 1 until the middle of the search and remains 1 to the end;
- NoisyEO(1) $\kappa = 1$, constant;

Algorithm 2 κEO(s, s_{best}) iteration

```
For current configuration $s$ evaluate $u_i(s)$, the fitness function corresponding of node $i \in \{1, \ldots, n\}$.
find the $\kappa$ worst components and replace them with a random value;
if ($s$ is better$^{(***)}$ than $s_{best}$) then
    set $s_{best} := s$.
end if
```

$^{(***)}$ better modularity value (1)

4. Numerical Experiments

Numerical experiments, performed on several benchmarks, are used to compare the results offered by the five NoisyEO variants with those offered by other state-of-art algorithms.

Experimental set-up. Numerical experiments were performed on synthetic benchmarks and real-world networks with the five variants of *NoisyEO*: NoisyEO(L2), NoisyEO(L),NoisyEO(E), NoisyEO(1), $\tau-$EO. Results were compared with four state-of-art methods: *Louvain* [1], *OSLOM* [9], *Infomap* [16], and *ModOpt* [17] - run by using the source code from `sites.google.com/site/andrealancichinetti/software`, last accessed May, 2015. *Louvain* and *ModOpt* optimize the modularity, *OSLOM* uses a probability of a node to belong to a community and *Infomap* is based on a random walk.

Parameter settings. The algorithm parameters are the same for all variants of *NoisyEO*: population size 50, initial value of $p_{shift} = 1$, $G = 45$, $NrShifts = 150$; the interval for the number of communities for each network is estimated such that the correct number is included and assigned to approx. 25% of the population.

Benchmarks. Four sets of synthetic networks were generated:

- GN: 128 nodes, 4 equal sized communities, node degree 16, z_{out} indicates the number of links a node has outside its community; 30 networks for each $z_{out} \in \{1, \dots 8\}$;
- LFR 128 nodes: average vertex degree 20, maximum vertex degree 50, community size $[10, 50]$;
- LFR 1000 nodes, S - small: average vertex degree 20, maximum vertex degree 50, community size $[10, 50]$;
- LFR 1000 nodes, B - big: average vertex degree 20, maximum vertex degree 50, community size $[20, 100]$

The LFR sets are characterized by the mixing parameter μ value - computed as the ratio between the number of links a node has outside its community and its degree. For each set and each μ value, we generated 30 networks. The most challenging sets are those where $\mu \in \{0.5, 0.6\}$ and $z_{out} = 8$, because they have a less well-defined community structure. Even among these networks (128 nodes, 1000 nodes small and big), the most difficult ones are the small ones (128 nodes), because if we increase the network size and the number of communities, a better-defined structure is created.

The real-world networks used for experiments are: the bottle-nose *dolphin* network [12], the *football* network [7], the Zachary *karate* club network [19], and the *books* about US politics network – `www.org\net.com`, last accessed 9/3/2015.

Performance evaluation. We use the normalized mutual information (NMI) proposed in [8] to evaluate and to compare results. A NMI value of 1 indicates, that two communities are identical. We compared obtained results for each method to the real community structure of the network.

For each benchmark set, results are further compared by using the Wilcoxon sign-rank nonparametric test (for 30 independent runs for each real network and on the 30 networks for each GN and LFR sets). The Wilcoxon sign rank specifies if the difference between two sample medians may be considered significant: the null hypothesis that two samples come from the same population can be rejected with a level of significance $\alpha = 0.05$ if the computed p-value is smaller than 0.05.

Results and discussion. Results are presented in the form of box-plots of NMI values obtained for the 30 runs by each method, in Figures 3–5. Next to each box-plot, a black-white matrix corresponding to Wilcoxon h values indicates the results of the pairwise comparisons of the tested methods: a black square shows a statistical difference between the two methods.

Regarding the difference between NoisyEO variants, the Wilcoxon h matrices show very few differences between the three adaptive variants: E, L, and L2. The exponential variant, E, shows worst results for the GN $z_{out} = 8$ set, but this result is still better than all the results obtained by the other methods. Setting $k = 1$ and $\tau-$EO do not yield good results compared to the other EO variants for the synthetic benchmarks. For the real-world networks, results are much closer, but still the three variants outperform the others.

5. Conclusion

A comparative analysis of four variants of extremal optimization updating procedures for the community structure detection problem is presented. The results show that the use of an adaptive method of setting the number of nodes to be randomly reassigned each iteration is beneficial; however, differences between tested variants are not significant enough to enable us to draw a conclusion regarding the best variant for

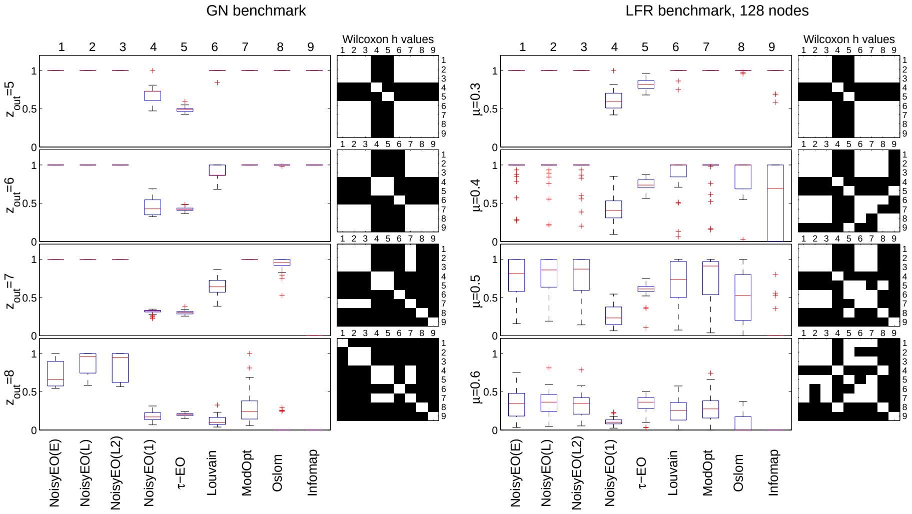

Figure 3: GN and LFR benchmarks, 128 nodes. Comparisons with other methods. Boxplots of NMI values obtained for the 30 networks in each set by each considered method. Wilcoxon h values matrices illustrate the statistical significance of the differences in results for the nine methods: a black box corresponds to $p < 0.05$ and rejection of the null hypothesis that the two samples have the same median.

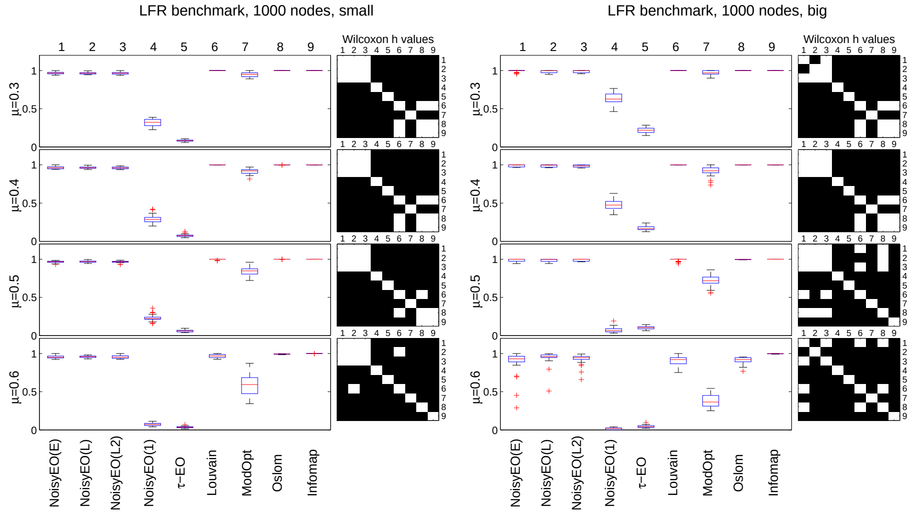

Figure 4: LFR benchmark, 1000 nodes, Small and Big. Results are represented in the same manner as in Fig. 3

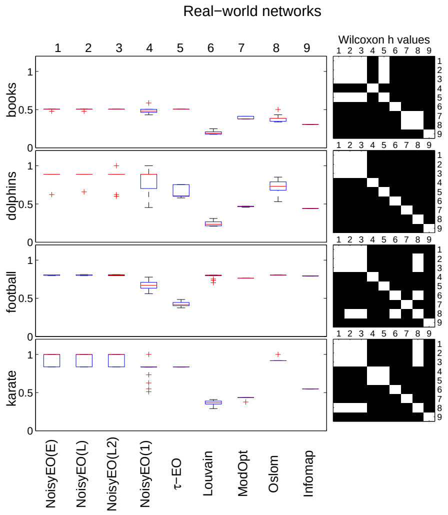

Figure 5: Real-world networks. Results are represented in the same manner as in Fig. 3.

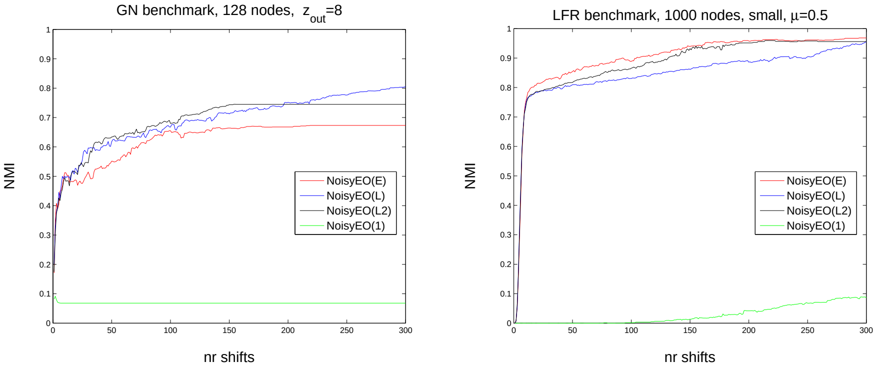

Figure 6: Evolution of NMI values in time for two of the most difficult sets: GN $z_{out} = 8$ and LFR 1000 nodes with $\mu = 0.5$. It is obvious that adapting the values of k leads to better results. The differences in means represented here have no statistical significance for the LFR benchmark.

the tested problems. Only when using an exponential rule, results are worse than the other EO variants, but even in those situations, they are very good.

Numerical results also show that extremal optimization may be very powerful in addressing the problem of community structure detection. Its main drawback, however, arises from the fact that random the computational time required by the iterative random changes makes this approach less efficient for large networks. On the other hand, this method proved very efficient for small networks with less visible community structures. Further work consists in finding the means to improve its scalability while maintaining its efficiency in dealing with ambiguous community structures.

Acknowledgment: This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS - UEFISCDI, project number PN-II-RU-TE-2014-4-2332.

References

[1] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

[2] S. Boettcher and A. Percus. Nature's way of optimizing. *Artificial Intelligence*, 119:275–286, 2000.

[3] S. Boettcher and A. G. Percus. Optimization with Extremal Dynamics. *Physical Review Letters*, 86:5211–5214, 2001.

[4] S. Boettcher and A. G. Percus. Extremal optimization: an evolutionary local-search algorithm. In *Computational Modeling and Problem Solving in the Networked World*. Operations Research/Computer Science Interfaces Series 21, pages 61–77, Kluwer Academic Publishers, 2002.

[5] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Physical Review E*, 72:027104, Aug 2005.

[6] S. Fortunato. Community detection in graphs. *Physics Reports*, 486:75–174, 2010.

[7] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[8] A. Lancichinetti, S. Fortunato, and J. Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, 2009.

[9] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato. Finding statistically significant communities in networks. *PloS One*, 6(4):e18961, 2011.

[10] Z. Li, S. Zhang, R.-S. Wang, X.-S. Zhang, and L. Chen. Quantitative function for community detection. *Physical Review E*, 77:036109, 2008.

[11] R. I. Lung, M. Suciu, and N. Gaskó. Noisy extremal optimization. *Soft Computing*, pages 1–18, 2015.

[12] D. Lusseau, K. Schneider, O. Boisseau, P. Haase, E. Slooten, and S. Dawson. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003.

[13] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004.

[14] C. Pizzuti. Ga-net: A genetic algorithm for community detection in social networks. *Lecture Notes in Computer Science*, 5199:1081–1090, 2008.

[15] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences*, 101(9):2658–2663, 2004.

[16] M. Rosvall and C. T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, 2008.

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

[18] M. Suciu, R. I. Lung, and N. Gaskó. Mixing Network Extremal Optimization for Community Structure Detection. *Lecture Notes in Computer Science*, 9026:126–137, 2015.

[19] W. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.