Experimental Evaluation of Topological-based Fitness Functions to Detect Complexes in PPI Networks

Clara Pizzuti
National Research Council of Italy, CNR
ICAR
Via P. Bucci 41C, 87036 Rende(CS), Italy
pizzuti@icar.cnr.it

Simona Rombo
National Research Council of Italy, CNR
ICAR and DEIS, Univ. della Calabria
Via P. Bucci 41C, 87036 Rende(CS), Italy
simona.rombo@deis.unical.it

ABSTRACT

The detection of groups of proteins sharing common biological features is an important research issue, intensively investigated in the last few years, because of the insights it can give in understanding cell behavior. In this paper we present an extensive experimental evaluation campaign aiming at exploring the capability of Genetic Algorithms (GAs) to find clusters in protein-protein interaction networks, when different topological-based fitness functions are employed. A complete experimentation on the yeast network, along with a comparative evaluation of the effectiveness in detecting true complexes on the yeast and human networks, reveals GAs as a feasible and competitive computational technique to cope with this problem.

Categories and Subject Descriptors

H.2.8 [**Database Managment**]: Database Applications —*Data Mining*; J.3 [**Computer Applications**]: Life and Medical Science—*Biology and Genetics*; I.5.3 [**Computing Methodologies**]: Pattern Recognition—*Clustering*

General Terms

Algorithms

Keywords

Protein-Protein Interaction Networks, Complex Detection, Genetic Algorithms

1. INTRODUCTION

The molecular characterization of cellular activity is a challenging issue, thus, in the last few years, an increasing number of scientists, such as biologists, computer scientists and mathematicians, have been working to model and analyze biological processes. The most common assumption in this context is understanding the cell as a complex and dynamic system of interacting components, that cannot be analyzed independently [32]. In particular, the discovery and study of interactions between proteins is receiving great attention, also due to both high-throughput (e.g., yeast two-hybrid and coimmunoprecipitation) and computational techniques [14, 15] exploited to obtain a large amount of available interactions.

A powerful way of modeling the whole set of protein-protein interactions of a given organism is the protein-protein interaction (PPI) network. A PPI network is an undirect graph where nodes represent proteins and each edge is associated with a physical interaction (actual or predicted) between two proteins.

PPI networks can also be viewed as sets of interacting complexes, i.e. groups of physically or functionally related proteins joining together to accomplish distinct functions [6]. Thus, proteins can be grouped in clusters such that the proteins in the same cluster share common biological features, such as participating in the same processes, having similar functions, belonging to the same cellular compart. The detection of such clusters provides important knowledge about biological processes, giving a valuable help in understanding the behavior of the cell. This pushed for the proposal of several clustering techniques applied to PPI networks, most of which can be broadly categorized as distance-based and graph-based ones [19]. Distance-based clustering approaches apply traditional clustering techniques, such as hierarchical clustering, by employing the concept of distance between two proteins (e.g., [7, 4, 25]). Graph-based clustering techniques consider instead the topology of the PPI network under analysis (e.g., [31, 23, 2, 9, 27, 12]).

The clustering techniques proposed in the literature are based on various strategies, for example searching for subgraphs having maximum density [23, 2, 12], partitioning the graph by optimizing a cost function [31], exploiting the concept of flow simulation [9] or co-clustering approaches [27]. However, at the best of our knowledge, very few evolutionary techniques have been applied to cluster PPI networks. In particular, in [20] the authors proposed an algorithm based on evolutionary computation for enumerating maximal cliques and apply it to the yeast genomic data. This approach uses chaotic variables to initialize the population of individuals and adds chaotic disturbance in the fitness computation. The method needs to set some threshold values that bias it in the search for an optimal solution, but how to set these thresholds is neither discussed nor investigated. More recently, an immune genetic algorithm to find dense subgraphs based on efficient vaccination method, variable-

GECCO'12, July 7-11, 2012, Philadelphia, Pennsylvania, USA.

length antibody schema definition and new local and global mutations has been proposed in [28] and applied to clustering protein-protein interaction networks.

In this paper, we present an extensive experimental evaluation campaign aiming at exploring the capability of Genetic Algorithms (GAs) to find clusters in PPI networks, when different topological-based fitness functions are employed. The representation of individuals we adopted is the graph-based adjacency representation, originally proposed in [24], and particularly apt for the detection of dense groups of nodes in networks [26]. A complete experimentation on the *Saccaromycaes Cerevisiae* (yeast) PPI network has been performed, and a comparative evaluation of their effectiveness in detecting complexes is reported by using various evaluation metrics, currently adopted to assess computational methods for complex detection. In particular, the clusters predicted by the genetic algorithm using each fitness function are compared with the true known complexes stored in the MIPS databases [21], according to some validation measures widely exploited in the literature [3, 5, 17]. Furthermore, a comparison with the well known $MCODE$ method [5] to detect protein complexes has been performed. The analysis shows that evolutionary computational methods can constitute a valid alternative to state of the art approaches.

The paper is organized as follows. The next section formalizes the problem of complex detection, introduces the fitness functions that will be used, and briefly describes the adopted genetic operators. Section 3 describes the evaluation measures used for comparison. In Section 4 the results of the experiments are reported. Finally, Section 5 concludes the paper and suggests future developments.

2. METHODS

A PPI network N can be modeled as an undirected graph $G = (V, E)$ where V is a set of $n =\mid V \mid$ nodes, each corresponding to a specific protein, and E is a set of $m =\mid E \mid$ undirected edges corresponding to the pairwise interactions. The problem of clustering PPI networks may be interpreted as that of finding dense regions, that is, finding sub-graphs of the graph G associated with N having high density of edges within them, and lower density of edges between groups. This definition of clustering is rather intuitive and vague, thus several criteria have been introduced in order to understand at the best its intrinsic meaning, and heuristics to optimize them have been proposed. However, different criteria can generate different groupings of nodes, thus it is difficult to choose what is deemed the best.

Recently, Leskovec et al. [16] observed that the concept of *good cluster* relies on two criteria. The first is the number of edges between the members of the cluster, the second is the number of edges between the members of the cluster and the rest of the network. Thus they group quality indices in two categories: multi-criterion scores, that combine both criteria, and single criterion scores, that are based on only one criterion. The authors used these criteria to compare a range of community detection methods. In the following we propose to use these quality indices as fitness functions with the aim of detecting complexes in PPI networks, and to perform an experimental evaluation of the obtained results by comparing the complex predicted by using these measures with respect to the true complexes. In the following the definition of these measures is first reported, then the adopted genetic representation and variation operators are described.

2.1 Fitness Functions

Let $G = (V, E)$ be the graph modeling a PPI network, S be a cluster of nodes having n_s nodes and m_s edges, and $c_s = \{(u, v) \mid u \in S, v \notin S\}$ be the number of edges on the boundary of S. Let $\{S_1, \ldots, S_k\}$ be a partition of G in k clusters. The following metrics, reported from [16], that catch the concept of quality of a clustering, are defined.

Conductance: $Co = \sum_{s=1}^{k} \frac{c_s}{2m_s+c_s}$ measures the fraction of edges pointing outside the clustering.

Expansion: $Ex = \sum_{s=1}^{k} \frac{c_s}{n_s}$ measures the number of edges per nodes that point outside the clustering.

Cut Ratio: $CR = \sum_{s=1}^{k} \frac{c_s}{n_s(n-n_s)}$ measures the fraction of all possible edges leaving the clustering.

Normalized Cut: $NC = \sum_{s=1}^{k} \frac{c_s}{2m_s+c_s} + \frac{c_s}{2(m-m_s)+c_s}$ measures the fraction of total edge connections to all the nodes in the graph.

The lower the values of these scores, the better the quality of the clustering obtained. All the above measures are considered multi-criterion scores since they take into account both the edges inside a cluster and those crossing between groups. A single criterion score is the well known concept of *modularity* introduced by Girvan and Newman [22].

Modularity: $Q = \sum_{s=1}^{k} [\frac{2m_s}{m} - (\frac{d_s}{2m})^2]$ measures the expected number of edges between the nodes of a cluster S in a random graph with the same degree sequence, where d_s is the sum of degrees of the nodes of s.

Thus the first term of each summand is the fraction of edges inside a cluster, and the second one is the expected value of the fraction of edges that would be in the network if edges fall at random without regard to the cluster structure. Values approaching 1 indicate high quality clustering. However, it has been proved [11] that the optimization of modularity has a topological resolution limit that depends on both the total size of the network and the interconnections of groups. This implies that small, tightly connected clusters could not be found. This limit implies the drawback that, searching for partitioning of maximum modularity, may lead to solutions in which important structures at small scales are not discovered. To overcome this problem, Granell et al. [13] introduced a resolution control parameter γ in the modularity formulation, $Q_R = \sum_{s=1}^{k} [\frac{2m_s}{m} - \gamma(\frac{d_s}{2m})^2]$. When $\gamma = 1$ the original formulation is obtained and, for increasing values of γ, smaller groups of nodes can be found.

2.2 Genetic representation and operators

The genetic algorithm uses locus-based adjacency representation proposed in [24]. In this graph-based representation an individual of the population consists of n genes $g_1, \ldots, g_n$ and each gene can assume allele values j in the range $\{1, \ldots, n\}$. Genes and alleles represent nodes of the graph $G = (V, E)$ modelling a PPI network, and a value j assigned to the ith gene is interpreted as a link between the proteins i and j. This means that in the clustering solution found i and j will be in the same cluster. The initialization process assigns to each node i one of its neighbors j. This guarantees a division of the network in connected groups of nodes. The kind of adopted crossover operator is uniform crossover. Given two parents, a random binary vector is created. Uniform crossover then selects those genes where the vector is a 0 from the first parent, and those genes where the

vector is a 1 from the second parent, and combines genes to generate the child. The mutation operator, analogously to the initialization process, randomly assigns to each node i one of its neighbors.

3. EVALUATION MEASURES

Available interaction data stored in public databases are not always reliable, since they are often obtained by prediction and computational techniques. MIPS databases [21] provide a collection of manually curated high-quality PPI data, collected from the scientific literature by expert curators. Only data from individually performed experiments are included, since they usually provide the most reliable evidence for physical interactions. By considering curated protein complexes stored in MIPS databases [21], the effectiveness of a method in detecting such known complexes can be evaluated by comparing the predicted clusters with the true known complexes.

In the following we describe some validation measures widely exploited in the literature [3, 5, 17] that will be used for the comparative analysis presented in this work. For the generic predicted cluster P_i and the generic known complex K_j, let $\mid P_i \mid$ and $\mid K_j \mid$ be their sizes, respectively. Furthermore, let $\mid P_i \cap K_j \mid$ be the size of the intersection set of the predicted cluster and the known complex. To evaluate how a predicted cluster P_i matches a known complex K_j, the *overlapping score* between P_i and K_j is defined as $OS(P_i, K_j) = \frac{|P_i \cap K_j|^2}{|P_i| \cdot |K_j|}$.

A known complex and a predicted cluster are considered a *match* [17] if $OS(P_i, K_j) \geq \sigma_{OS}$, i.e. their overlapping score is equal to or larger than a specific threshold σ_{OS}. To estimate the performance of algorithms for detecting protein complexes w.r.t. the overlapping score, the notions of *sensitivity* and *specificity*, commonly used in information retrieval and machine learning (also known as *recall* and *precision*), as well as a cumulative measure called *f-measure* are introduced.

Sensitivity: $S_n = \frac{TP}{TP+FN}$ is the fraction of the true-positive predictions out of all the true predictions, where TP (true positive) is the number of the predicted clusters matched by the known complexes with $OS(P_i, K_j) \geq \sigma_{OS}$, and FN (false negative) is the number of the known complexes that are not matched by the predicted clusters.

Specificity: $S_p = \frac{TP}{TP+FP}$ is the fraction of the true-positive predictions out of all the positive predictions, where FP (false positive) equals the total number of the predicted clusters minus TP.

F-measure: $F_m = \frac{2 \cdot S_n \cdot S_p}{S_n + S_p}$ is a measure that summarizes sensitivity and specificity. High values of f-measure means that both sensitivity and specificity are sufficiently high.

4. EXPERIMENTAL RESULTS

In this section we present the results of an extensive experimentation of the genetic algorithm to evaluate prediction capability when different topological measures, described above, are used as fitness functions. It is known that setting parameter values is a challenging research problem in evolutionary algorithms [10]. Though recently Smith and Eiben [30] found that good parameter values can be obtained for a set of problems, general tuning allowing for good performance on a wide range of problems raises specific difficulties. In particular, there are no studies regarding the application domain of PPI networks. Thus a complete experimental campaign has been performed by running the genetic algorithm for all combinations of values of crossover fraction and mutation rate, ranging from 0.2 to 1, with an increment step of 0.2. Furthermore we set elite reproduction 10% of the population size, roulette selection function, population size 50, and number of generations 50. For all the experiments, the statistical significance of the obtained results has been checked by performing a t-test at the 5% significance level. The p-values returned are very small, thus the significance level is very high since the probability that a complex could be obtained by chance is very low. The implementation has been written in MATLAB 4.3 R2010a, using Genetic Algorithms and Direct Search Toolbox 2. We run the GA method on a publicly available benchmark, namely the Database of Interacting Proteins, DIP, consisting of 17,203 interactions among 4,930 proteins. In order to evaluate the predicted complexes, a benchmark set of 428 gold standard complexes coming from different sources, such as MIPS and SGD database based on Gene Ontology annotations, have been used. Both the network and the true complexes have been provided by Li et al. [18].

Figure 1, Figure 2, and Figure 3 show the sensitivity, specificity, and f-measure values obtained for all the fitness functions, when crossover fraction and mutation rate vary in the range of values [0.2, 0.4, 0.6, 0.8, 1]. The first observation that can be done is that the values computed by using the evaluation measures do not show a high variation at the varying of crossover and mutation values, the differences being of the order of at most one decimal digit. For example, in Figure 1(a) the highest value of sensitivity is 0.393, obtained with mutation rate and crossover fraction both equal to 0.4, while the lowest value is 0.35 when mutation rate is 1 and crossover fraction is 0.6, differing only of 0.04 with respect to the second decimal digit. Thus the genetic approach seems to be rather stable as regards the choice of the parameter values that could give improved performance on the yeast network. The best value of sensitivity is obtained when the fitness function is modularity with resolution parameter $\gamma = 3$, mutation rate 0.2, and crossover fraction 0.8 (see Figure 1(b)). Actually, the differences with modularity are not very high, but the resolution parameter allows to partition the network in a larger number of smaller clusters. This can be seen in Table 1, where the number of complexes predicted by using each fitness function is shown, together with the number of predicted complexes that match at least a true complex. From the table it can be observed that, when the fitness function is Q_R, the number of both predicted clusters and matched complexes are the highest with respect to the other measures. However, looking at Figure 2, modularity is not the best performing measure. In fact, in such a case, all the other fitness functions obtain higher values of specificity. This means that modularity partitions the network such that the fraction of proteins correctly predicted is higher than the other scores. On the other hand, higher values of specificity, means lower number of false positive, that is in the same cluster the fraction of proteins effectively belonging to that cluster is higher. However, since maximizing both scores is often difficult, a tradeoff between the two, i.e. the f-measure, allows to choose a model with good values of both sensitivity and specificity. To this end, Figure 3 shows that modularity obtains the best values of f-measure.

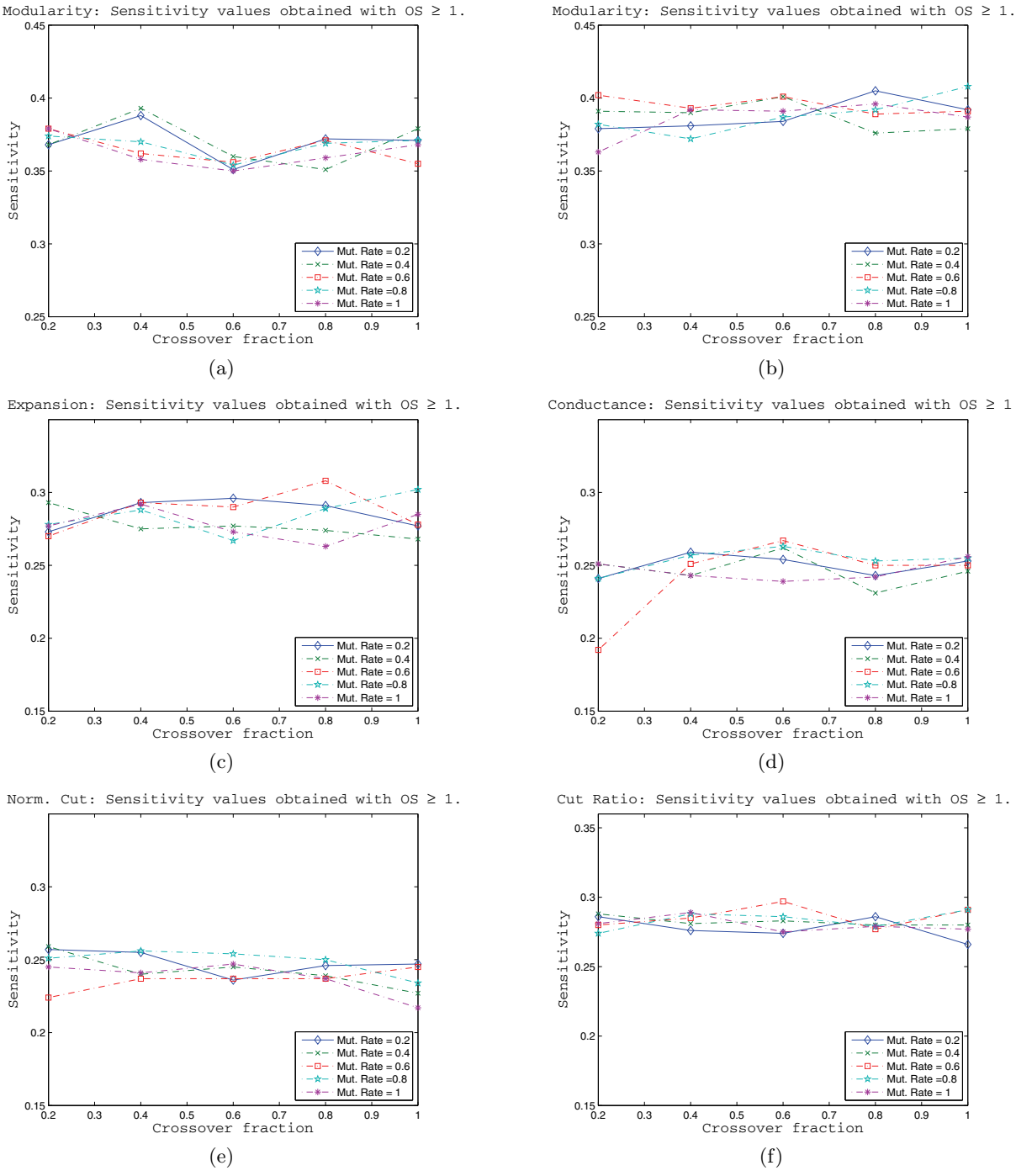

Figure 1: Sensitivity values for different values of crossover and mutation rate, with $\sigma_{OS} \geq 1$, for: (a) modularity (b) modularity with γ=3, (c) expansion, (d) conductance, (e) Normalized Cut, (f) CutRatio.

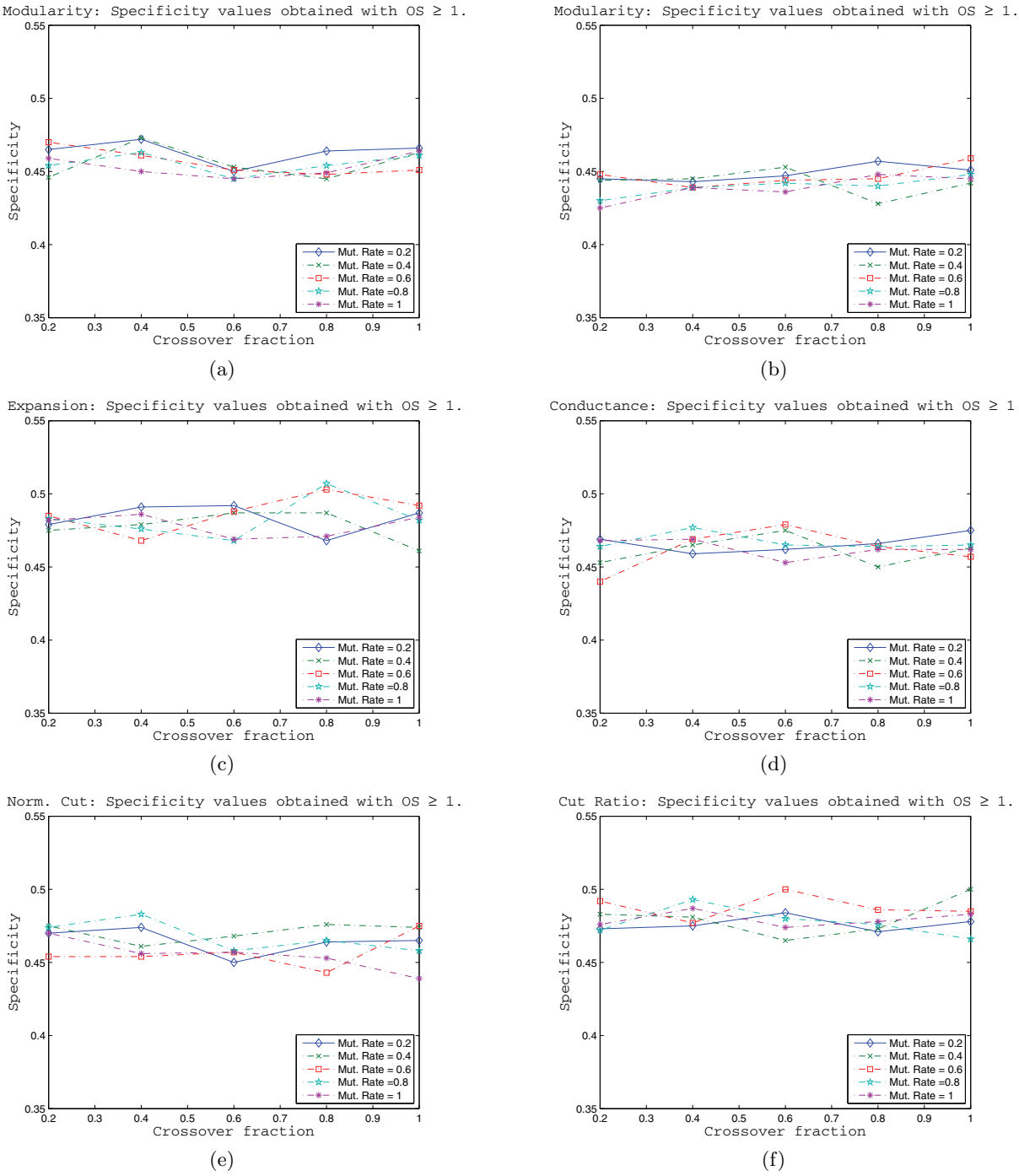

Figure 2: Specificity values **for different values of crossover and mutation rate, with $\sigma_{OS} \geq 1$, for: (a) modularity (b) modularity with γ=3, (c) expansion, (d) conductance, (e) Normalized Cut, (f) CutRatio.**

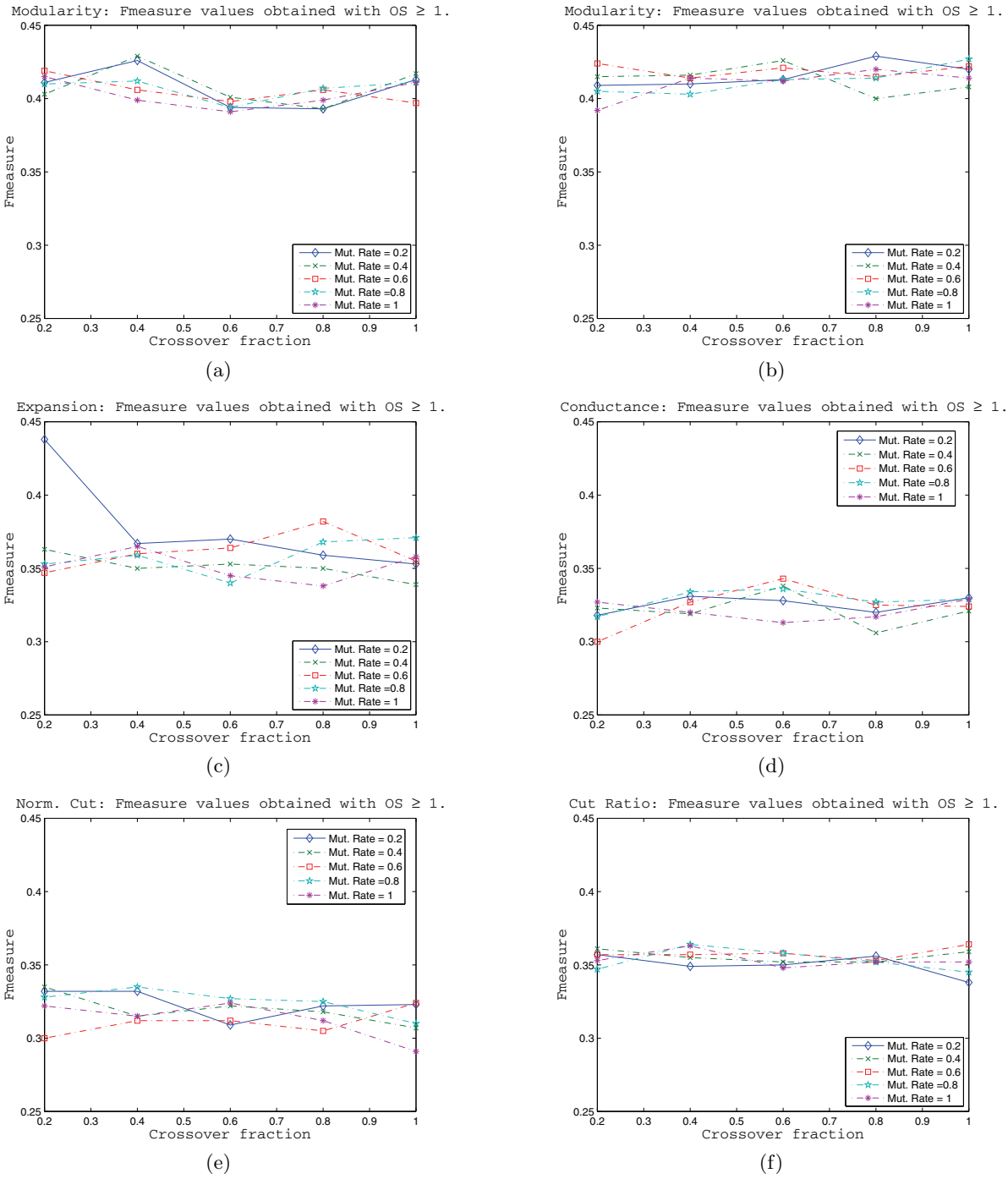

Figure 3: `Fmeasure values` **for different values of crossover and mutation rate, with $\sigma_{OS} \geq 1$, for: (a) modularity (b) modularity with γ=3, (c) expansion, (d) conductance, (e) Normalized Cut, (f) CutRatio.**

Table 1: Results with the various fitness functions on the DIP network.

Fitness	Co	Ex	CR	NC	Q	Q_R
# predicted complexes	221.8 (15.2)	242.2 (13.4)	242.4 (11.9)	215.6 (15.7)	329.8(11.8)	360.4 (13.6)
# matched complexes	116.9 (6.5)	112.6 (9.1)	130.7 (3.4)	114.3 (4.9)	167.7 (5.5)	177.6 (5.03)

Table 2: Comparative performance between the GA approach with modularity as fitness function, and *MCODE*, both non overlapping and overlapping (MCODE-OV) on the Human network.

Algorithms	MCODE	MCODE-OV	GA
Specificity	0.252	**0.306**	0.153
Sensitivity	0.141	0.176	**0.461**
F-measure	0.181	0.224	**0.23**

Another important consideration about the results we obtained is that the number of predicted clusters is lower than the number of true complexes. This means that the genetic approach, endowed with the topological measures previously described, finds clusters of larger size that could include some true complex. Considering that the benchmark set of 428 gold standard complexes do not cover all the proteins of the DIP network, thus many proteins are not assigned to any group, the detection of clusters of larger size, including a true complex, could be exploited for functional annotation of proteins. In fact, as pointed out by Sharan et al. [29], an approach to functional annotation of proteins is based on assigning the function that is prevalent in a group of proteins, obtained by dividing the PPI network in dense clusters. Figure 4 shows an example of a cluster predicted constituted by 7 proteins, namely YAL003W, YBR118W, YKL081W, YPL048W, YCR042C, YHL034C, YPL226W, that contains true MIPS complex composed by the first four proteins. It is interesting to note that the protein YPL226W is not assigned in the benchmark set of true complexes, though it is connected with the proteins YKL081W and YDR142C. Because of the above observations, YPL226W could be annotated with the same function of the 4-proteins MIPS complex.

Finally, we compare the GA approach, when modularity is used as fitness function, with one of the most known clustering technique proposed in the literature for PPI networks, i.e. the algorithm *MCODE* [5]. This method allows to detect also overlapping clusters, i.e. a protein can belong to more than one cluster, thus we report the results for both the versions of *MCODE*. For such a comparison, we used a different PPI network that is, the *Homo Sapiens* (human) network by setting mutation rate to 0.2, crossover fraction to 0.4, and by choosing the modularity as fitness function. The choice of another network has been done to test the performance of the better parameter values obtained by the previous experimentation on the yeast network. The human network, consisting of 6,716 nodes and 16,322 interactions, has been downloaded from the MINT database [8], while the benchmark set of $1,083$ known and curated complexes for human has been taken from [1]. Table 2 points out that the genetic approach is comparable with *MCODE*. In fact, it obtains higher values of sensitivity and f-measure, while specificity is lower than both non-overlapping *MCODE* and overlapping *MCODE* (denoted as *MCODE-OV*).

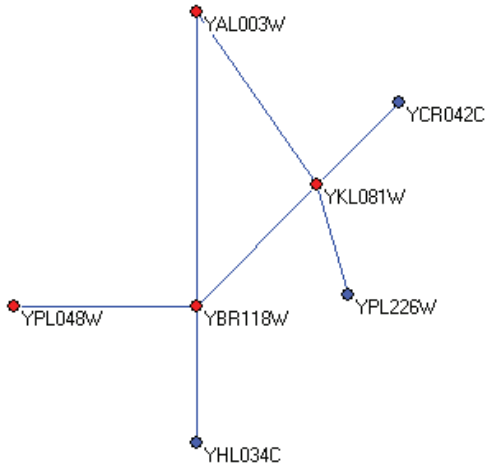

Figure 4: An example of predicted cluster containing the true MIPS complex YAL003W, YBR118W, YKL081W, YPL048W (red nodes).

5. CONCLUSIONS

This paper presented an extensive experimentation by using Genetic Algorithms endowed with six topological-based fitness functions, for the detection of dense groups of proteins in PPI networks. The results showed that this computational method is a viable choice to obtain significative solutions. It is worth to point out that a main drawback of the proposed approach is that a protein can be assigned to only one cluster. However, many proteins present the characteristic of being connected to a high number of other proteins, thus often participating in multiple biological processes and performing different functions. The incapability of detecting overlapping clusters is due to the genetic representation, that does not allow a node to be connected to more than one other node. Future work aims at extending the graph-based representation to allow for the detection of overlapping clusters, such that a protein can belong to several clusters, and to compare the approach with other state of the art clustering methods.

6. ACKNOWLEDGMENTS

This work has been partially supported by the project *MERIT : MEdical Research in Italy*, funded by MIUR.

7. REFERENCES

[1] Website title: http://mips.helmholtz-muenchen.de/genre/proj/corum.

[2] B. Adamcsek, G. Palla, I. J. Farkas, I. Dernyi, and T. Vicsek. Cfinder: locating cliques and overlapping modules in biological networks. *Bioinformatics*, 22(8):1021–1023, 2006.

[3] M. Altaf-Ul-Amin, Y. Shinbo, K. Mihara, K. Kurokawa, and S. Kanaya. Development and implementation of an algorithm for detection of protein complexes in large interaction networks. *BMC Bioinformatics*, 7(207), 2006.

[4] V. Arnau, S. Mars, and I. Marìn. Iterative cluster analysis of protein interaction data. *Bioinformatics*, 21(3):364–378, 2005.

[5] G. Bader and H. Hogue. An automated method for finding molecular complexes in large protein-protein interaction networks. *BMC Bioinformatics*, 4(2), 2003.

[6] A. Barabási and Z. N. Oltvai. Network biology: Understanding the cell's functional organization. *Nature Review Genetics*, 5:101–113, 2004.

[7] M. Blatt, S. Wiseman, and E. Domany. Superparamagnetic clustering of data. *Phisical Review Letters*, 76(18):3251–3254, 1996.

[8] A. Ceol et al. Mint, the molecular interaction database: 2009 update. *Nucleic Acids Research*, 38(Database issue):D532–D539, 2010.

[9] Y.-R. Cho, W. Hwang, M. Ramanathan, and A. Zhang. Semantic integration to identify overlapping functional modules in protein interaction networks. *BMC Bioinformatics*, 8(265), 2007.

[10] A. E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2):124–141, 1999.

[11] S. Fortunato and M. Barthélemy. Resolution limit in community detection. *Proc. National Academy of Science, USA*, 104(1):36–41, 2007.

[12] E. Georgii, S. Dietmann, T. Uno, P. Pagel, and K. Tsuda. Enumeration of condition-dependent dense modules in protein interaction networks. *Bioinformatics*, 25(7):933–940, 2009.

[13] C. Granell, S. Gómez, and A. Arenas. Unsupervised clustering analisys: A multiscale complex network approach. *Journal of Bifurcation and Chaos, in press*, 2012.

[14] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proceedings of the National Academy of Science of the United States of America*, 98(8):4569–4574, 2001.

[15] N. J. Krogan et al. Global landscape of protein complexes in the yeast Saccharomyces cerevisiae. *Nature*, 440(7084):637–643, 2006.

[16] J. Leskovec, K. Lang, and M. Mahoney. Empirical comparison of algorithms for network community detection. In *Proc. Int. World Wide Web Conference (WWW 2010)*, pages 631–640, 2010.

[17] M. Li, J. Chen, J. Wang, B. Hu, and G. Chen. Modifying the DPClus algorithm for identifying protein complexes based on new topological structures. *BMC Bioinformatics*, 9(398), 2008.

[18] X. Li, M. Wu, C. Kwoh, and S. Ng. Computational approaches for detecting protein complexes from protein interaction network: a survey. *BMC Genomics*, 11(Suppl 1): S3, 2010.

[19] C. Lin, Y. Cho, W. Hwang, P. Pei, and A. Zhang. Clustering methods in protein-protein interaction network. *in Knowledge Discovery in Bioinformatics: Techniques, Methods and Application, John Wiley & Sons,Inc*, pages 319–355, 2006.

[20] H. Liu and J. Liu. Clustering protein interaction data through chaotic genetic algorithm. In *Simulated Evolution and Learning*, volume 4247 of *Lecture Notes in Computer Science*, pages 858–864. Springer Berlin / Heidelberg, 2006.

[21] H. W. Mewes and al. MIPS: a database for genomes and protein sequences. *Nucleic Acids Res.*, 30(1):31–34, 2002.

[22] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review*, E69:026113, 2004.

[23] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435:814–818, 2005.

[24] Y. Park and M. Song. A genetic algorithm for clustering problems. In *Proc. of 3rd Annual Conference on Genetic Algorithms*, pages 2–9, 1989.

[25] P. Pei and A. Zhang. A two-step approach for clustering proteins based on protein interaction profiles. In *IEEE Int. Symposium on Bioinformatics and Bioengeneering (BIBE'2005)*, pages 201–209, 2005.

[26] C. Pizzuti. GA-Net: a genetic algorithm for community detection in social networks. In *Proc. of the 10th International Conference on Parallel Problem Solving from Nature (PPSN 2008)*, pages 1081–1090. LNCS 5189, Springer, 2008.

[27] C. Pizzuti and S. E. Rombo. A Coclustering Approach for Mining Large Protein-Protein Interaction Networks. *IEEE/ACM Transactions on Compututational Biology and Bioinformatics*, 9(3): 717–730, 2012.

[28] H. Ravaee, A. Masoudi-Nejad, S. Omidi, and A. Moeini. Improved immune genetic algorithm for clustering protein-protein interaction network. In *Proceedings of the 2010 IEEE International Conference on Bioinformatics and Bioengineering*, BIBE '10, pages 174–179. IEEE Computer Society, 2010.

[29] R. Sharan, I. Ulitsky, and R. Shamir. Network-based prediction of protein function. *Molecular Systems Biology*, 3(88), 2007.

[30] S. K. Smit and A. E. Eiben. Parameter tuning of evolutionary algorithms: Generalist vs. specialist. In *Applications of Evolutionary Computation, Springer*, pages 542–551, 2010.

[31] V. Spirin and L. A. Mirny. Protein complexes and functional modules in molecular networks. *PNAS*, 100:12123–12128, 2003.

[32] D. von Mering, C. Krause, and *et al.* Comparative assessment of a large-scale data sets of protein-protein interactions. *Nature*, 31:399–403, 2002.