Robust Community Detection Methods with Resolution Parameter for Complex Detection in Protein Protein Interaction Networks

Twan van Laarhoven and Elena Marchiori

Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands
tvanlaarhoven@cs.ru.nl, elenam@cs.ru.nl

Abstract. Unraveling the community structure of real-world networks is an important and challenging problem. Recently, it has been shown that methods based on optimizing a clustering measure, in particular modularity, have a resolution bias, e.g. communities with sizes below some threshold remain unresolved. This problem has been tackled by incorporating a parameter in the method which influences the size of the communities. Methods incorporating this type of parameter are also called multi-resolution methods. In this paper we consider fast greedy local search optimization of a clustering objective function with two different objective functions incorporating a resolution parameter: modularity and a function we introduced in a recent work, called w-log-v. We analyze experimentally the performance of the resulting algorithms when applied to protein-protein interaction (PPI) networks. Specifically, publicly available yeast protein networks from past studies, as well as the present BioGRID database, are considered. Furthermore, to test robustness of the methods, various types of randomly perturbed networks obtained from the BioGRID data are also considered. Results of extensive experiments show improved or competitive performance over MCL, a state-of-the-art algorithm for complex detection in PPI networks, in particular on BioGRID data, where w-log-v obtains excellent accuracy and robustness performance.

1 Introduction

The development of advanced high-throughput technologies and mass spectrometry has boosted the generation of experimental data on protein-protein interaction and shifted the study of protein interaction to a global, network level. In particular, it has been shown that groups of proteins interacting more with each other than with other proteins, often participate in similar biological processes and often form protein complexes performing specific tasks in the cell. Detecting protein complexes, consisting of proteins sharing a common function, is important, for instance for predicting a biological function of uncharacterized proteins. To this aim protein-protein interaction (PPI) networks have been used as a convenient graph-based representation for the comparative analysis and detection of (putative) protein complexes [18]. A PPI network is a graph where nodes are proteins and edges represent interactions between proteins.

Detecting protein complexes in a PPI network can be formalized as a graph-clustering problem. Clustering amounts to divide data objects into groups (clusters) in such a way that objects in the same cluster are more similar to each other than to objects in the other clusters. Since clustering is an ill-posed and computationally intractable problem, many methods have been introduced, in particular for graph-clustering (see e.g. the recent review by Fortunato [7]). Effective methods for graph-clustering contain a parameter whose tuning affects the community structure at multiple resolution scales. These methods are also called multi-resolution methods (see e.g. [15]). The resolution parameter(s) can be used in two main ways: as a parameter to be tuned; or as a way to generate clusterings at multiple resolution scales, which can then be used to analyze the clustering behavior of objects across multiple resolutions [17], or to ensemble the results to produce a consensus clustering [22].

In [27], the resolution bias of state-of-the-art community detection methods has been analyzed, and a simple yet effective objective function was introduced. Results indicated that methods based on greedy local search optimization are robust to the choice of the clustering objective function, when a multi-resolution parameter is added to the objective function.

The goal of this paper is to investigate experimentally the performance of such multi-resolution methods when applied to PPI networks, with respect to data generated from different laboratory technologies as well as with respect to random removal or shuffling of edges in the network. This latter investigation is motivated by the fact that PPI data are still not fully reliable, with the potential inclusion of both false positive and false negative interactions (see e.g. the discussion in [18]). Specifically, we consider fast greedy local search optimization of a clustering objective function with two different objective functions incorporating a resolution parameter: modularity [11] and a function we introduced in a recent work, called w-log-v [27].

To analyze their performance we consider the yeast Saccharomyces cerevisiae, which is a well studied model organism for higher eukaryotes with several protein interaction data generated from diverse laboratory technologies. Specifically, we consider six PPI networks from past studies and the present BioGRID curated database of protein interactions [24]. In order to assess robustness with respect to random perturbations of the graph, we generate a large collection of networks using the BioGRID data, by either removing or by adding a percentage of randomly selected edges, or by randomly shuffling edges while keeping the original degree of each node.

Results of the experiments indicate improved performance of modularity and w-log-v over MCL (the Markov Cluster Algorithm) [26], a state-of-the art method for community detection in PPI networks based on stochastic flow in graphs. MCL was found to achieve best overall performance in yeast PPI networks [2] and competitive performance with methods for overlapping community detection in PPI networks [20].

In particular best performance is achieved by w-log-v on the BioGRID data, and excellent robustness on randomly perturbed versions of this network. Since PPI networks are known to be noisy with respect to the presence of both false positive and false negative interactions, the high robustness shown by the proposed algorithm substantiates its effectiveness on this type of data.

1.1 Related work

A vast literature on protein complex detection with PPI networks exists (see e.g. the review [18]). Previous related works on multi resolution algorithms for clustering PPI networks either apply an algorithm multiple times with different values of the resolution parameter in order to investigate how proteins cluster at different resolution scales, e.g. [17], or tune the resolution parameter in order to choose a best setting for the considered type of networks, e.g. [2, 20]. Here we aim at investigating thoroughly effectiveness and robustness of two such algorithms by means of an extensive experimental analysis.

In [2] a comparative assessment of clustering algorithms for PPI networks was conducted. In particular, robustness was analyzed, with respect to alterations (addition and/or removal of randomly selected edges) of a test graph which was constructed using a number of yeast complexes annotated in the MIPS database, by linking each pair of proteins belonging to the same complex. The considered algorithms with parameters tuned on the test graph, were then applied to various yeast datasets. Results showed that MCL with inflation (resolution) parameter value equal to 1.8 was performing best on the considered datasets. Robustness of MCL when applied to yeast PPI networks has previously also been analyzed in [21]. According to their results, MCL is rather robust across different networks and with respect to missing or noisy information on protein-protein associations. Here we show that greedy local search optimization of a clustering objective function (e.g. w-log-v) incorporating a resolution parameter achieves improved robustness (and accuracy) on the BioGRID data.

2 Methods

2.1 Greedy Local Search Optimization

A recent experimental study by Lancichinetti and Fortunato [16] showed that the best methods for graph community detection are those of Blondel *et al.* [1] and Rosvall and Bergstrom [23]. Both of these methods use a similar greedy local search optimization procedure (LSO), which is based on moving nodes between clusters, and constructing a clustering bottom-up. The only difference between these methods is the objective that is optimized. We briefly outline this LSO method here.

LSO is a discrete optimization method that finds a clustering without overlapping clusters. Initially, each node is assigned to a singleton cluster. Then, iteratively, nodes are moved between neighboring clusters as long as the objective improves.

Eventually a local optimum will be reached, but the clusters in this local optimum will often be too small. The next step is to repeat the optimization procedure, but this time moving these small clusters instead of single nodes. Effectively, we are then clustering a condensed graph, where each node in the condensed graph is a small cluster. When another local optimum is reached, the condensed graph is further condensed and clustered, and so on.

Because the condensed graphs are much smaller than the original graph, most of the time is spend clustering the original graph. Since this is done in an local and greedy fashion, the overall algorithm is very fast. For instance, it takes less than a second to cluster a graph with 6000 nodes.

2.2 Objectives

The considered optimization method is independent of the objective function that is optimized. Hence we are essentially free to choose the objective to best fit the application. In this paper we will limit ourselves to two objectives. The first is the popular modularity [12],

$$\text{modularity}(Cl) = \sum_{A \in Cl} -\text{nwithin}(A) + \text{nvol}(A)^2.$$

Here Cl denotes a clustering, i.e. a set of clusters. For a particular cluster $A \in Cl$, its volume is $\text{vol}(A) = \sum_{i \in A, j \in V} w_{ij}$, i.e. the sum of the weight of edges incident to nodes in A, which is equivalent to the sum of degrees. Based on the volume we define the normalized volume as $\text{nvol}(A) = \text{vol}(A)/\text{vol}(V)$, where V is the set of all nodes. Finally $\text{within}(A) = \sum_{i,j \in A} w_{ij}$ is the within cluster volume, and $\text{nwithin}(A) = \text{within}(A)/\text{vol}(V)$ is its normalized variant.

The second objective we consider is w-log-v, which was introduced in van Laarhoven and Marchiori [27]. An advantage of this objective over modularity is that it allows more diverse cluster sizes. Because the sizes of protein complexes can differ widely, we believe that this is a useful property. The w-log-v objective is defined as

$$\text{w-log-v}(Cl) = \sum_{A \in Cl} \text{nwithin}(A) \log(\text{nvol}(A)).$$

Using either of the above objectives directly for the task of clustering a PPI network is not advisable. Both objectives were designed for community detection; and communities are usually relatively large, much larger than protein complexes. Therefore, optimizing these objectives will lead to a clustering with a small number of large clusters. This inability to find small clusters is termed the resolution limit of the objective [8].

Table 1. Sizes of the different datasets. The last column lists the number of MIPS complexes that are (partially) contained in each dataset.

Dataset	Nodes	Edges	Complexes
Uetz *et al.* (2000) [25]	927	823	20
Ho *et al.* (2002) [13]	1563	3596	43
Gavin *et al.* (2002) [9]	1352	3210	50
Gavin *et al.* (2006) [10]	1430	6531	53
Krogan *et al.* (2006) [14]	2674	7075	75
Collins *et al.* (2007) [5]	1620	9064	63
BioGRID, all physical	5967	68486	97

To overcome the resolution limit, we add a parameter to the objectives as follows,

$$\text{modularity}_\alpha(Cl) = \text{modularity}(Cl) + \alpha \sum_{A \in Cl} \text{nwithin}(A),$$
$$\text{w-log-v}_\alpha(Cl) = \text{w-log-v}(Cl) + \alpha \sum_{A \in Cl} \text{nwithin}(A).$$

By increasing the parameter α, the clustering is punished for within cluster edges, and hence the optimal clustering will have smaller clusters. Alternatively, by decreasing the parameter α, the clustering is rewarded for within cluster edges, so the optimal clustering will then have larger clusters.

It can be shown that, because the overall scale of the objective is irrelevant for optimization, the modification is equivalent to assuming that the overall volume of the graph is different. For modularity, the adjustment corresponds to assuming that the graph has volume $(1 - \alpha)\,\text{vol}(V)$, while for w-log-v it corresponds to assuming the volume is $e^{-\alpha}\,\text{vol}(V)$. This equivalent interpretation provides some intuition for the resolution parameter: when we tune α to find smaller clusters, the objective is equivalent to that for finding the clusters in a smaller graph.

3 Experiments

3.1 PPI networks

We downloaded a set of protein interactions from version 3.1.88[1] of the BioGRID database [24]. This database contains a collection of protein interactions from different sources, and discovered with different methods. In this work we only consider interactions found by physical experiments, not those based on genetics.

The BioGRID also contains in full several datasets from high throughput experimental studies, including Uetz *et al.* [25], Ho *et al.* [13], Gavin *et al.* [9, 10], Krogan *et al.* [14], Collins *et al.* [5]. We consider these subnetworks as separate datasets in our experiments. These datasets are generated with different experimental techniques: the Collins [5], Krogan [14] and Gavin [10] datasets include the results of TAP tagging experiments only, while the BioGRID dataset contains a mixture of TAP tagging, Y2H and low-throughput experimental results [20]. Table 1 lists the sizes of these datasets.

[1] This version was released on April 25th, 2012

3.2 Complex validation

For validation, we compare clusters with the complexes from the MIPS database [19][2], which we take as the gold standard. MIPS specified a hierarchy of complexes and subcomplexes. Since we deal only with non-overlapping clustering, we only include (sub)complexes at the bottom of this hierarchy. And to avoid degenerate cases, we include only complexes with at least 3 proteins. We also exclude the complexes in category 550, since these are unconfirmed. They were found with computational clustering methods, using as input the same high throughput datasets that we consider.

In addition to the complexes from MIPS, we also use a set of complexes derived from the Gene Ontology annotations of the Saccharomyces Genome Database [3]. This dataset was created and also used by [20].

To compare clusters found by a method to either of these sets of gold standard complexes, we use the overlap score [2],

$$\omega(A, B) = \frac{|A \cap B|^2}{|A||B|}.$$

We consider a cluster to *match* a complex if their overlap score is at least 0.25. This threshold is also used in other works, e.g. [20, 4]. When the cluster and complex have the same size, a match then corresponds to the intersection containing at least half of the nodes in the complex and cluster.

Based on this matching we define *precision* as the fraction of clusters that are matched to any complex. Conversely, we define *recall* as the fraction of complexes that are matched to any cluster. Note that we use the terminology from other works such as [4]. These notions differ from the more standard definitions of precision and recall, because a cluster can match more than one complex and vice versa.

It is clearly possible to achieve a high precision or a high recall with a degenerate clustering. For example, by returning just a single easy to find cluster that matches a complex, the precision will be 1 at the cost of a low recall. And by returning all possible (overlapping) clusters, the recall will be 1 at the cost of a low precision. We therefore use the F_1 score, which is the harmonic mean of precision and recall, as a trade-off between the two scores.

For each of the methods, we include only clusters that contain at least 3 proteins. As a result, not all proteins will be in a cluster. We call the fraction of proteins that are in a cluster the *coverage* of a clustering.

The precision and recall as defined above depend heavily on the chosen threshold; and when few complexes are matched, the scores are very sensitive to noise. Therefore, we also look at the positive predictive value (PPV) and cluster-wise sensitivity scores [2], which are based directly on the size of the intersection between complexes and clusters,

$$\text{PPV} = \frac{\sum_{A \in Cl} \max_{B \in Co} |A \cap B|}{\sum_{A \in Cl} \sum_{B \in Co} |A \cap B|} \qquad \text{Sensitivity} = \frac{\sum_{B \in Co} \max_{A \in Cl} |A \cap B|}{\sum_{B \in Co} |B|},$$

where Cl is the set of predicted clusters and Co is the set of gold standard complexes. Note that the asymmetry between the denominators is to account for the case of overlapping clusters.

3.3 Precision vs recall

We took the BioGRID all physical dataset, and computed the precision and recall for a wide range of settings of the resolution control parameter α. These results are shown in figure 1 (left). For

[2] We used the latest version at the time of writing, which was released on May 18th, 2006

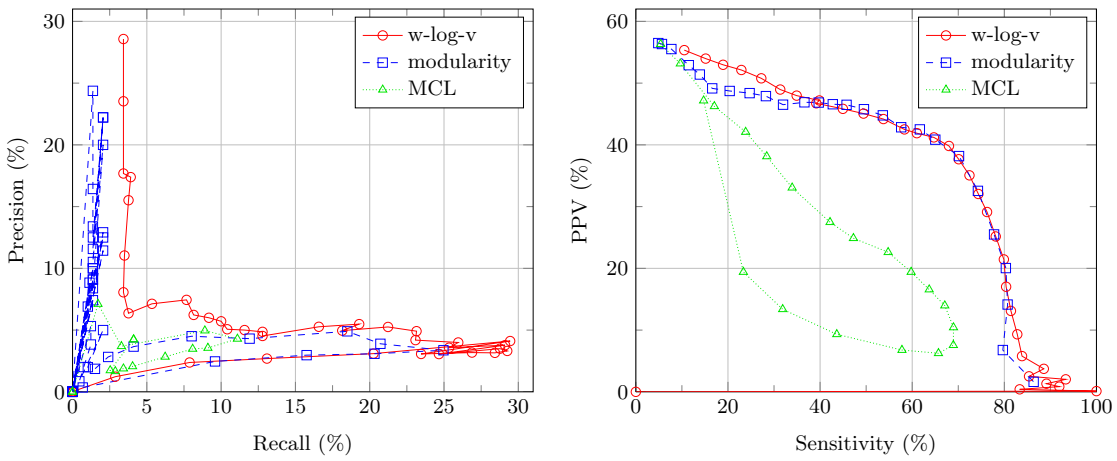

Fig. 1. Precision vs. Recall (left) and Sensitivity vs. PPV (right) on the BioGRID dataset.

comparison we also include results with the MCL algorithm for different settings of the inflation parameter. For readability we have applied smoothing in the form of merging points that are very close together.

The first thing that we observe is that despite smoothing, the figure is very noisy in some places. This is not very surprising considering how precision and recall are calculated. Consider a small change in the clustering, such as removing a protein from a cluster. This change might cause the cluster to no longer match a particular complex. If there are no other clusters that matched that complex, then the recall goes down, otherwise it stays the same. Similarly, if this are no other complexes matching the cluster, then the precision goes down. While obviously the change in the two scores is related, the relation is not monotonic, one can change while the other does not.

As the resolution control parameter α goes up, the methods find more clusters; and as a result the recall goes up while the precision goes down. However, after a certain point many of the clusters will become too small, and they will be removed before matching. This decreased coverage causes the recall to go down again.

To get a less noisy picture, we have also plotted the PPV and sensitivity scores, in figure 1 (right). The overall trend in this plot is the same as for the precision and recall: the w-log-v method slightly dominates modularity optimization, which in turn has significantly better results than MCL.

The best parameter settings according to the F_1 score are $\alpha = 2.8$ for w-log-v, $\alpha = 0.97$ for modularity, and inflation 2.7 for MCL. We will use these settings for the remainder of the experiments. As discussed in section 2.2, the parameter α corresponds to assuming a different volume of the graph. The optimal setting for w-log-v corresponds to considering a graph with 16 times fewer edges, while the optimal setting for modularity corresponds to 33 times fewer edges. The difference between the two objectives comes from their inherent resolution bias, by default w-log-v has a bias towards smaller clusters compared to modularity, and therefore the objective needs less adjustment.

3.4 Networks from Single Studies

We next compare the scores on the subnetworks from single studies. The results of this experiment are shown in table 2. The "MIPS method" is based on the gold standard complexes, but including only proteins that occur in the dataset under investigation. It represents the best possible scores.

Table 2. Results of applying the different methods to subnetworks for single studies. The best result for each dataset is highlighted in bold.

Dataset	Method	Clusters	Coverage	Precision	Recall	Sens.	PPV	F_1
Uetz *et al.* (2000)	MIPS	20	2.5%	85.0%	11.6%	9.3%	69.1%	20.5%
	w-log-v	173	21.1%	4.6%	**6.2%**	6.7%	73.6%	**5.3%**
	modularity	160	**21.8%**	**5.0%**	5.5%	**7.0%**	70.1%	5.2%
	MCL	143	17.2%	4.2%	4.1%	5.3%	**75.0%**	4.2%
Ho *et al.* (2002)	MIPS	43	5.2%	88.4%	26.0%	17.5%	67.8%	40.2%
	w-log-v	278	46.0%	2.5%	**6.8%**	11.8%	**69.3%**	3.7%
	modularity	257	**46.2%**	**2.7%**	6.2%	**12.0%**	67.4%	**3.8%**
	MCL	227	35.5%	1.8%	2.7%	10.7%	64.4%	2.1%
Gavin *et al.* (2002)	MIPS	50	7.9%	92.0%	32.9%	27.9%	62.5%	48.4%
	w-log-v	202	**39.7%**	12.4%	**21.2%**	**21.6%**	**72.0%**	15.6%
	modularity	199	39.0%	11.6%	19.9%	19.9%	69.6%	14.6%
	MCL	177	34.5%	**14.7%**	**21.2%**	21.4%	71.1%	**17.4%**
Gavin *et al.* (2006)	MIPS	53	7.8%	92.5%	34.9%	28.3%	63.6%	50.7%
	w-log-v	193	**40.9%**	13.5%	21.2%	23.3%	**72.3%**	16.5%
	modularity	188	37.9%	13.3%	19.9%	22.6%	70.6%	15.9%
	MCL	164	33.4%	**16.5%**	**21.9%**	**24.0%**	71.6%	**18.8%**
Krogan *et al.* (2006)	MIPS	75	8.6%	97.3%	50.7%	38.3%	66.1%	66.7%
	w-log-v	401	58.9%	8.0%	**25.3%**	26.6%	72.6%	12.1%
	modularity	314	**60.4%**	**9.6%**	23.3%	**26.8%**	70.3%	**13.5%**
	MCL	380	43.9%	7.4%	21.2%	22.4%	**73.5%**	10.9%
Collins *et al.* (2007)	MIPS	63	8.9%	98.4%	43.8%	32.9%	65.6%	60.7%
	w-log-v	194	**38.8%**	20.1%	**32.2%**	27.3%	**75.1%**	**24.8%**
	modularity	177	34.3%	20.3%	29.5%	26.7%	72.8%	24.1%
	MCL	172	36.8%	**20.9%**	30.1%	**29.1%**	69.9%	24.7%
BioGRID, all physical	MIPS	97	7.9%	96.9%	64.4%	55.3%	70.2%	77.4%
	w-log-v	505	**84.5%**	**5.9%**	**22.6%**	**38.6%**	69.4%	**9.4%**
	modularity	599	83.6%	4.2%	19.2%	35.6%	**72.1%**	6.9%
	MCL	283	69.0%	5.3%	11.6%	28.7%	40.6%	7.3%

On most datasets w-log-v has the best recall and F_1 score, except on the datasets from Gavin et al. Gavin *et al.* [9, 10], where MCL performs significantly better. The precision of modularity optimization is often slightly better than that for w-log-v optimization. This is due to the fact that with the settings chosen in the previous paragraph, we find more clusters with w-log-v optimization. Hence, in general recall will be higher at the cost of lower precision.

3.5 Randomly perturbed graphs

To further test the robustness of the methods, we applied them to randomly perturbed networks. We performed three different experiments, all starting from the BioGRID network.

1. Removing a randomly chosen subset of the interactions.
2. Randomly adding new spurious interactions between pairs of proteins.
3. Randomly rewire a subset of the edges, while maintaining the degree of each node. Note that such a move both removes an observed interaction and adds a new spurious one.

We varied the amount of edges affected by each type of perturbation. Each experiment was repeated 10 times with different seeds for the random number generator, we calculated the mean and

standard deviation of the F_1 score across these repetitions. The results are shown in figure 2. When edges are removed, the performance of all methods degrades similarly. On the other hand, the LSO methods are much more robust to the addition of extra edges than MCL. Also note that the standard deviation is much larger with the MCL method. That means that for some rewired graphs the method gives reasonably good results, while for others the result is very bad. Unsurprisingly, the experiment with rewired edges sits somewhere in between the two other experiments.

4 Discussion

Because of the incompleteness of both the PPI data and of knowledge on true complexes, care must be taken in the interpretation of the results. The "MIPS method", that is, the best possible method based on the MIPS complexes, covers only a small part of the proteins present in each of the datasets. Conversely, not all MIPS complexes are covered by the datasets, so the recall is always smaller than the precision. In general, results show that for each dataset, the majority of clusters induced by the intersection of that dataset with the complexes in MIPS, match a complex; with a percentage varying between 85% and 98%. These values provide upper bounds on the maximum precision and recall achievable on the considered dataset.

On all datasets the algorithms obtain precision smaller than recall: the difference of these values provides information on the fraction of clusters matching more than one complex. For instance on the Uetz dataset, there is almost a one-to-one correspondence between clusters and matched complexes (e.g. 4.6% precision and 6.2% recall for w-log-v), while on the BioGRID this relation is clearly one-to-many (e.g. 5.9% precision and 22.6% recall for w-log-v).

There are complexes that are matched by only one method: specifically, 6 complexes are matched only by w-log-v, 5 only by modularity, and 4 only by MCL. Comparing w-log-v and MCL, there are 18 complexes found only by w-log-v and 4 found only by MCL. This is not too surprising, since MCL has a rather low recall. An example of a complex detected by w-log-v and not by MCL is the Signal recognition particle (SRP) complex, consisting of six proteins, one of the complexes involved in Transcription and/or in the Nucleus [3].

The improved performance of w-log-v on the BioGRID data appears mainly due to its capability to generate a large number of clusters matching multiple complexes (high recall). Nevertheless, figure 1 shows that the precision vs. recall curve of w-log-v dominates the curve of the other two methods.

Robustness of a community detection method is and important issue also in the context of PPI networks, since they are known to contain a high amount of false negative and false positive interactions. Indeed, limitations of experimental techniques as well as the dynamic nature of protein interaction are responsible for the high rate of false-positives and false-negatives generated by high-throughput methods. For instance, Y2H screens have false negative rates in the range from 43% to 71% and TAP has false negative rates of 15%-50%, and false positive rates for Y2H could be as high as 64% and for TAP experiments they could be as high as 77% [6]. Results show that w-log-v achieves best robustness the under random addition, removal and rewiring of a percentage of edges in the BioGRID network. Such high robustness substantiates the effectiveness of w-log-v on this type of data.

5 Conclusions

This paper analyzed the performance of two fast algorithms on PPI networks that optimize in a greedy way a clustering objective function with resolution parameter. An extensive experimental

[3] see e.g. `http://pin.mskcc.org/web/align.SubtreeServlet?dbms=mysql&db=interaction&species=SC`

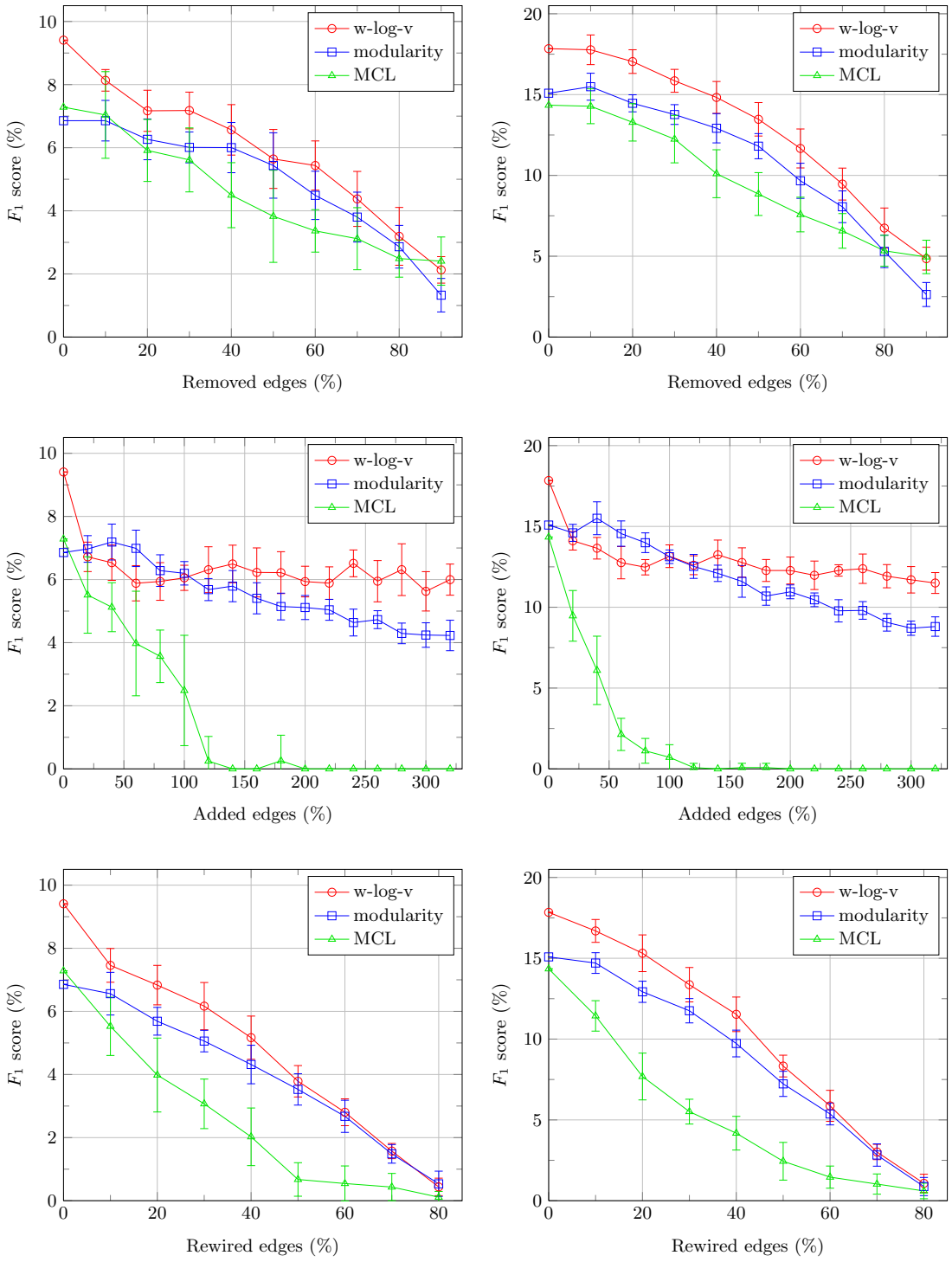

Fig. 2. F_1 score when a fraction of the edges is added (top), removed (middle) or rewired (bottom) at random in the BioGRID dataset. Error bars indicate standard deviation, measured over 10 runs. The left plots use MIPS complexes as the gold standard, the right plots use SGD complexes.

analysis was conducted on PPI data from previous studies as well as on the present BioGRID database. Results indicated improved performance of the considered algorithms over a state-of-the-art method for complex detection in PPI networks, in particular with respect to robustness. These results indicate that the considered algorithms provide an efficient, robust and effective approach for protein complex discovery with PPI networks. Interesting issues for future work include the assessment of the algorithms' robustness with respect to tailored models of false positive and false negative interactions which are present in data generated by specific technologies, as well as the extension of the considered methods to detect overlapping clusters of high quality.

Our implementation of the LSO method in Octave/C++ is available from `http://cs.ru.nl/~T.vanLaarhoven/prib2012/`.

Acknowledgments

This work has been partially funded by the Netherlands Organization for Scientific Research (NWO) within the NWO project 612.066.927.

References

[1] Blondel, V. D. *et al.* (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, **2008**(10), P10008+.

[2] Brohée, S. and van Helden, J. (2006). Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, **7**(1), 488+.

[3] Cherry, J. M. *et al.* (2011). Saccharomyces Genome Database: the genomics resource of budding yeast. *Nucleic Acids Research*.

[4] Chua, H. N. *et al.* (2008). Using indirect protein-protein interactions for protein complex prediction. *Journal of bioinformatics and computational biology*, **6**(3), 435–466.

[5] Collins, S. *et al.* (2007). Towards a comprehensive atlas of the physical interactome of saccharomyces cerevisiae. *Molecular Cellular Proteomics*, pages 600381–600200.

[6] Edwards, A. M. *et al.* (2002). Bridging structural biology and genomics: assessing protein interaction data with known complexes. *Trends in Genetics*, **18**(10), 529–536.

[7] Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, **486**, 75–174.

[8] Fortunato, S. and Barthélemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, **104**(1), 36–41.

[9] Gavin, A. C. *et al.* (2002). Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, **415**(6868), 141–7.

[10] Gavin, A.-C. *et al.* (2006). Proteome survey reveals modularity of the yeast cell machinery. *Nature*, **440**(7084), 631–636.

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

[12] Girvan, M. and Newman, M. E. J. (2002b). Community structure in social and biological networks. In *Proc. National. Academy of Science. USA 99*, pages 7821–7826.

[13] Ho, Y. *et al.* (2002). Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry. *Nature*, **415**(6868), 180–183.

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

[15] Lambiotte, R. (2010). Multi-scale modularity in complex networks. In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pages 546–553.

[16] Lancichinetti, A. and Fortunato, S. (2009). Community detection algorithms: A comparative analysis. *Phys. Rev. E*, **80**, 056117.

[17] Lewis, A. *et al.* (2010). The function of communities in protein interaction networks at multiple scales. *BMC Syst Biol.*, **4:100**.

[18] Li, X. *et al.* (2010). Computational approaches for detecting protein complexes from protein interaction networks: a survey. *BMC Genomics*, **11**(Suppl 1), S3+.

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

[20] Nepusz, T. *et al.* (2012). Detecting overlapping protein complexes in protein-protein interaction networks. *Nature Methods*.

[21] Pu, S. *et al.* (2007). Identifying functional modules in the physical interactome of Saccharomyces cerevisiae. *Proteomics*, **7**(6), 944–960.

[22] Ronhovde, P. and Nussinov, Z. (2009). Multiresolution community detection for megascale networks by information-based replica correlations. *Physical Review E*, **80**(1), 016109+.

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

[24] Stark, C. *et al.* (2006). Biogrid: a general repository for interaction datasets. *Nucleic Acids Research*, **34**(Database-Issue), 535–539.

[25] Uetz, P. *et al.* (2000). A comprehensive analysis of protein-protein interactions in saccharomyces cerevisiae. *Nature*, **403**(6770), 623–627.

[26] Van Dongen, S. (2008). Graph Clustering Via a Discrete Uncoupling Process. *SIAM Journal on Matrix Analysis and Applications*, **30**(1), 121–141.

[27] van Laarhoven, T. and Marchiori, E. (2012). Graph clustering with local search optimization: does the objective function matter? *submitted*.