

# Identification of shared components of protein complexes

J. Patrick Pett<sup>1,†</sup>, Daniel H. Mehnert<sup>1</sup>, Jonas Ibn-Salem<sup>1</sup>, Ole Eigenbrod<sup>1</sup>, Raik Otto<sup>1</sup>,  
Stephan Knorr<sup>1</sup>, Stina-Stephanie M. Richter<sup>1</sup> and Roland Krause<sup>1,2</sup>

<sup>1</sup> Free University Berlin, Department of Computer Science and Mathematics

<sup>2</sup> Max-Planck-Institute for Molecular Genetics, Department Vingron

<sup>†</sup> Corresponding author

**Abstract:** Interactions of proteins shape the processes of a cell, often as functional modules or complexes. A minority of proteins, called *shared components* in the biological literature, are part of multiple complexes. The detection of shared components with current methods for protein complex identification is only implicit and a secondary objective.

Here, we present and explore algorithms for finding shared components explicitly and compare their results based on a curated dataset of protein complexes. *Merged maximal cliques* is a global clustering algorithm computing overlapping clusters. An extension of the detection of articulation points to *articulation groups* searches for shared components as local cut vertices. Our combined approach, *Local cluster decomposition*, incorporates aspects of both and outperforms recent implicit methods for complex detection with respect to shared components.

## 1 Introduction

Large-scale identification of protein complexes by high-throughput methods has led to genomewide protein-protein interaction networks in *Saccharomyces cerevisiae* [KCY<sup>+</sup>06, GAG<sup>+</sup>06]. High confidence datasets can be obtained by integrating and refining published datasets [SFK<sup>+</sup>11, BRB<sup>+</sup>07]. Protein complexes are not necessarily distinct but can be overlapping. Proteins that are part of multiple complexes are called *shared components* (see Fig. 1). The term has its origin in biology and might refer to single proteins [RWR<sup>+</sup>98, CRGW98], not components in the graph theoretical sense. Gaining knowledge about shared components could be of interest for drug development, e.g. by providing hints at side effects or anti-microbial targets [KvMBD04]. The identification of shared components and their analysis is a largely unaddressed problem in the analysis of biological networks.

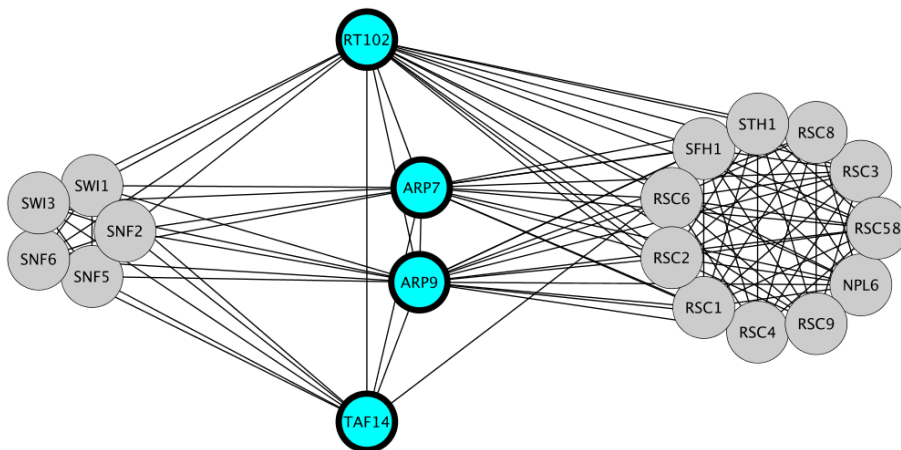


Figure 1: Examples for shared components of the SWI/SNF complex and the RSC complex. Depicted are the neighbors of Arp7 and Arp9 in the HC dataset and their interactions. Taf14, Rt102, Arp7 and Arp9 are shared between the complexes.

## 1.1 Background

Many protein complexes have been validated independently using high-throughput data and individual experiments. The current, curated gold standard of protein complexes for *Saccharomyces cerevisiae* contains 1342 proteins grouped into 236 complexes with more than two members and 176 shared components [PWT<sup>+</sup>09].

This number is relatively small and popular approaches like *Molecular Complex Detection* (MCODE) or approaches using *Markov Clustering* (MCL) do not address shared components in their core methods but in post-processing steps [BH03, GS11, VW09, FKZ08].

MCODE undertakes a weighting of all nodes by the k-core and density of its neighborhood, followed by a partition of the graph into disjoint complexes [BH03]. Protein complexes can be extended by proteins in their direct neighborhood if the weight of a protein exceeds a chosen threshold. MCL searches for groups within a weighted graph by simulating a random walk on the graph for its separation, which converges to a state with distinct clusters [EVDO02].

An expansion by Pu *et al.* computes shared components by applying a power law function to every complex defined by MCL. If a sufficient fraction of proteins belongs to two clusters, an overlap is noted. Information about shared components is solely used for improving protein complex definition [PVE<sup>+</sup>07].

Friedel *et al.* describe the application of MCL on a protein-protein interaction network to determine initial clusters. In a subsequent step, shared components are identified based on the strength of the connection between the nodes and clusters [FKZ08]. The results were found to be superior to previous methods but the shared components and their contributions were not explicitly benchmarked.

*Complex detection from coimmunoprecipitation data* (CODEC) is a recently described algorithm capable of native detection of overlapping clusters, which performed better than Friedel *et al.*. Its greedy search heuristic finds the heaviest bicliques in a bipartite graph, consisting of bait and prey interaction for the detection of protein complexes [GS11].

In perfect data complexes could be easier modelled as maximal cliques, which can overlap. Errors in biological network data necessitate modifications of this method and first described by Zhang *et al.* [ZPKS08].

## 1.2 Our contribution

Here, we present three approaches for the identification of shared components in protein-protein interaction networks. Our implementation and parametrization of the *merged max cliques* algorithm computes complexes on the complete network that can overlap [ZPKS08]. *Local cluster decomposition* detects nodes with several clusters in its immediate neighborhood.

The *articulation groups* algorithm computes sets of cut vertices without clustering. If shared components are removed from a set of overlapping complexes, the graph disconnects (see Fig 1). Therefore, the detection of shared components as cut vertices should be possible without definition of complexes.

Evaluation of the presented algorithms is based on a manually curated dataset comprising protein complexes and established shared components [PWT<sup>+</sup>09].

Local cluster decomposition outperforms the other approaches, as well as the tested methods for complex detection.

## 2 Methods

In the following, we consider the biological networks as unweighted, undirected graphs. Let  $G = (V, E)$  be a connected graph of protein-protein interactions. The set of all neighbors of a node  $v \in V$  is denoted as  $N(v) := \{u | u \in V, (u, v) \in E\}$ , where the cardinality of  $N(v)$  defines the degree of  $v$  as denoted by  $d(v)$ . The total number of nodes  $n$  in a graph  $G$  is defined as  $|V|$ .

The overlap of two groups of nodes  $V_i$  and  $V_j$  is measured by the Simpson coefficient

$$S(V_i, V_j) = \frac{|V_i \cap V_j|}{\min(|V_i|, |V_j|)}.$$

We define true positives (TP) as proteins that are shared components in CYC2008 (see 2.5) and that are found by the respective algorithm, with an analogue definition of true negatives (TN) as proteins that are not defined as shared components in CYC2008. If a protein is detected as a shared component by an algorithm and not marked as a shared component in CYC2008 it is called a false positive (FP). False negatives (FN) describe proteins that are not detected by the algorithm but are actually marked as shared components in the gold standard. Sensitivity is defined as the proportion actual positives that are predicted positive:

$$Sensitivity = \frac{TP}{(TP + FN)}$$

Specificity is analogously defined by the proportion of actual negatives that are predicted negative:

$$Specificity = \frac{TN}{(TN + FP)}$$

Matthews correlation coefficient (MCC) represents a measure of quality for binary classifications. Taking true and false positives and negatives into account it computes a correlation coefficient between observed and predicted binary classifications.

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

MMC computes a correlation coefficient between observed and predicted binary classifications, returning values between -1 (reverse prediction) and 1 (perfect prediction).

## 2.1 Merged Maximal Cliques (MMC)

Assuming a perfect method to detect protein-protein interactions, a complex would appear as a fully connected subgraph of  $G$ . Such a subgraph is known as a *maximal clique*, if it cannot be extended by a node and remains complete [BK73]. An appropriate algorithm to determine protein complexes respecting the undetected and false-positive interactions of high-throughput data is based on the merging of strongly overlapping maximal cliques [ZPKS08]. We adapt the algorithm to identify shared components as nodes that are members of two or more merged maximal cliques.

First, all maximal cliques  $C_i \subset V$  in the graph  $G$  are identified using an extension of the algorithm described by Bron and Kerbosh [BK73, CK08].

In contrast to Zhang's algorithm, MMC merges cliques in order of decreasing size. MMC takes  $G$  and an overlap threshold  $t_0$  as input and computes a set of overlapping cliques  $C$  as described in Algorithm 1. The proteins in the overlap of any two cliques are shared components.

---

### Algorithm 1 MMC

---

```

1: procedure MERGE MAXIMAL CLIQUES( $G, t_0$ )
2:    $C \leftarrow \{\forall \text{ maximal cliques } c \subseteq V \text{ in } G \mid |c| > 2\}$ 
3:   while  $\max\{S(c_i, c_j), \forall c_i, c_j \in C\} \geq t_0$  do
4:     for all  $c_i \in C$  in descending size order do
5:        $c_j \leftarrow \max_{c_j \in C} \{S(c_i, c_j), \forall c_j \in C\}$ 
6:       if  $S(c_i, c_j) \geq t_0$  then
7:          $c_k \leftarrow c_i \cup c_j$ 
8:          $C \leftarrow \{c_k\} \cup C \setminus \{c_i, c_j\}$ 
9:   return  $C$ 

```

---

The detection of all  $k$  maximal cliques in  $G$  is an NP-hard problem with a worst case run-time of  $O(3^{\frac{n}{3}})$  [TTT06]. Merging those cliques takes  $O(k^3)$  time, resulting in a total run-time for MMC of  $O(3^{\frac{n}{3}} + k^3)$ . The application of the algorithm on typical protein-protein interaction networks does not exceed several hours.

## 2.2 Local Cluster Decomposition (LCD)

The neighborhood of a node can be segmented into clusters. The number of clusters determines whether a given node  $v$  is a shared component. To this end, *local cluster decomposition* applies MMC to the closed neighborhood  $N(v)$  and the edge-set

$$F(v) := \{\{u, w\} \in E \mid u, w \in N(v)\}.$$

LCD takes a graph  $G$ , an overlap threshold  $t_0$  and  $s$  as input (See Algorithm 2). A vertex  $v \in V(G)$  is returned as a shared component, if its neighborhood  $N(v)$  comprises two or more groups of clustered vertices with a minimum number of elements  $s$ .

---

### Algorithm 2 LCD

---

```

1: procedure LOCAL CLUSTER DECOMPOSITION( $G, t_0, s$ )
2:   for all  $v \in V(G)$  do
3:      $G_{v,s} \leftarrow (N(v), F(v))$ 
4:      $K \leftarrow$  MERGE MAXIMAL CLIQUES( $G_{v,s}, t_0$ )
5:      $n_c(v) \leftarrow |\{k \in K \mid |k| \geq s\}|$ 
6:   return  $\{v \in V \mid n_c(v) \geq 2\}$ 

```

---

Considering the neighborhood graph  $G_{v,s} := (N(v), F(v))$  of a node  $v \in V(G)$ , the clustering with MMC runs in  $O(3^{\frac{m}{3}})$ , where  $m$  is defined as  $|G_{v,s}|$ .

Since the cluster identification has to run for each node, a time complexity of  $O(n \cdot 3^{\frac{\Delta G}{3}})$  with the number of nodes  $n$  and the graph's maximum degree  $\Delta G$  follows. Due to the scale-free distribution of the degree in the graph the exponent can be expected to be much smaller than  $\frac{\Delta G}{3}$  in practice. As LCD applies MMC on subsets, it runs faster than the global MMC. The fact that typical interaction networks are sparse further shortens the run-time of LCD.

## 2.3 Articulation Groups (AG)

In contrast to LCD and MMC, the articulation groups algorithm searches for shared components without defining complexes.

An articulation point or cut vertex  $v \in V$  is any node that increases the number of connected components when removed from the graph. A linear-time algorithm for detecting articulation points, based on a depth-first search, was provided by Hopcroft and Tarjan [HT73]. The articulacy  $a_v$  of a node  $v \in V$  is given by the minimal number of nodes  $k$  which have to be removed, before removing  $v$  disconnects  $G$ . To find nested shared components, like Arp7 and Arp9 (see Fig. 1), we extended this algorithm for detecting cut vertices of articulacy  $> 0$ . For any articulacy  $a_v > 0$  an articulation group  $V_A \subset V$  is defined as the set of  $a_v + 1$  nodes that has to be removed to disconnect the graph.

The algorithm could be executed on the giant component of the network, the biggest connected component of the given graph, considering all found articulation points up to a certain articulacy threshold  $a_t$  as shared components, which becomes infeasible to compute for a meaningful number of shared components.

Large, connected complexes do not have a shortest path length  $> 3$ . We can therefore select each node as a center node  $v_c$  of a subgraph  $G_s(v_c)$  containing all nodes within a distance of 3 to  $v_c$  and apply the algorithm repeatedly on each  $G_s(v_c)$ . The result is a list of all articulation points found in the subgraphs and their frequency.

For identifying all nodes showing a specified articulacy  $a_t$ , the algorithm has a run-time of  $O(n^{a_t+1})$ . To speed up the algorithm for articulacy  $> 0$ , only nodes with betweenness-centrality  $b_v \geq \text{median}(b_v \forall v \in V)$  are considered being candidates for articulation points. By this preprocessing step, peripheral nodes and uninformative node-chains are excluded. Furthermore, a node  $v$  can never be considered as an articulation point of any order if the local clustering coefficient  $c_v$  is 1, which is equal to  $v$  being in a maximal clique  $M$  and having no edges to any node  $u \notin M$ .

## 2.4 Data sources

High confidence networks of protein interaction in yeast were selected for analysis. The semi-manually curated, unweighted HC network comprises 2534 nodes and 6398 interactions [BRB<sup>+</sup>07]. The STRING database V8.3 contains weighted interactions. We selected the yeast networks at edge weights  $> 0.7$  comprising 5203 nodes and 69225 edges (STRING700) and  $> 0.9$ , 4246 nodes and 31934 interactions (STRING900), respectively [SFK<sup>+</sup>11].

Two datasets retrieved by mass spectrometry of tandem affinity purification data (TAP-MS) were analyzed, one by Gavin *et al.* containing 2551 proteins and 21413 interactions [GAG<sup>+</sup>06] and one by Krogan *et al.* containing 2705 proteins forming 7139 interactions [KCY<sup>+</sup>06].

## 2.5 Validation

The manually curated catalogue of yeast protein complexes CYC2008 serves as gold standard [PWT<sup>+</sup>09]. Complexes composed of only two members cannot be identified in unweighted interaction graphs and are not considered for the analysis. CYC2008 contains megacomplexes, holding recognized complexes as subsets. One example is the ribosome, consisting of two subunits which may or may not be considered as one complex [KvMBD04]. Another complication arises from variant complexes that differ only in a few proteins. An example is the RSC complex, which includes two homologous subunits RSC1 and RSC2, which are mutually exclusive [NRYS02].

To take the ambiguity due to megacomplexes and variant complexes into account, we merge complexes in the CYC2008 that are highly similar to each other, as measured by the Simpson coefficient  $S$ . Note, if  $C_i \subseteq C_j$  or  $C_j \subseteq C_i$  holds for two complexes  $C_i$  and  $C_j$ , then  $S(i, j) = 1$ . Shared components are proteins located in at least two CYC2008 complexes  $i$  and  $j$  with a Simpson coefficient  $S(i, j) \leq 0.8$ . To find the best parameters for MMC, LCD and AG, we determine the maximum MCC of the algorithms applied to the networks.

Only the intersections of CYC2008 and the corresponding datasets were considered, leading to different numbers of possible shared components contained in each dataset (see Table 1). Sensitivity and specificity were computed based on the set of binary classifiers obtained from each network. MCODE was parametrized according to [BVH06]. Results of CODEC were taken from the supplementary material of the original publication [GS11].

Table 1: Matthews correlation coefficient, sensitivity and specificity of the algorithms in networks with different number of proteins ( $n$ ), average degree ( $d(G)$ ) and number of shared components in CYC2008 (# SC). Data for CODEC is exclusively available for Gavin’s and Krogan’s interaction networks [GS11]. All algorithms’ prediction accuracies apparently depend on the network’s average degree  $d(G)$ .

	HC	STRING700	STRING900	Gavin	Krogan
$n$	2534	5203	4246	2551	2705
$d(G)$	5.0	26.6	15.0	16.8	5.3
# SC	80	98	96	58	59
MCODE	0.083 0.41/0.76	0.015 0.75/0.3	0.056 0.69/0.49	0.058 0.84/0.33	0.068 0.43/0.75
CODEC $w_0$	-	-	-	0.064 0.65/0.5	0.014 0.92/0.11
CODEC $w_1$	-	-	-	0.099 0.68/0.55	0.051 0.89/0.27
MMC	0.279 0.54/0.92	0.092 0.98/0.34	0.144 0.82/0.65	0.126 0.88/0.55	0.313 0.73/0.91
LCD	0.357 0.36/0.98	0.170 0.69/0.81	0.239 0.62/0.9	0.209 0.53/0.9	0.419 0.44/0.99
AP	0.134 0.63/0.72	0.064 0.63/0.6	0.085 0.27/0.9	-0.013 0.02/0.97	0.085 0.56/0.71

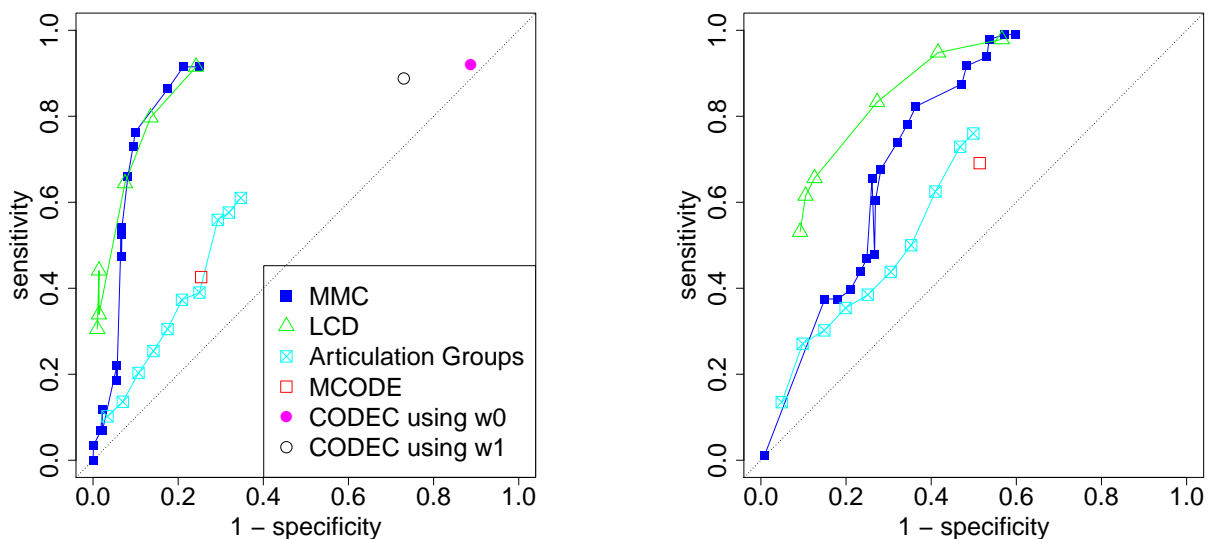


Figure 2: ROC-curves of the algorithms' performance on Krogan (left) and STRING900 (right). Note the superiority of the LCD algorithm in the dense STRING900 network. See Table 2 for comparison.

### 3 Results and discussion

The performance of all methods is dependent on the attributes of interaction data sets like density and degree distribution. However, a consistent picture emerges that finds LCD superior. The comparison table 1 lists the results of the presented algorithms and from the implicit algorithms CODEC and MCODE.

Table 2: Areas under the curve for the three explicit algorithms. Parameters used for gaining the underlying ROC-curves are size for LCD, Simpson coefficient for MMC and percentile detection cutoff in case of AG.

	HC	STRING700	STRING900	Gavin	Krogan
MMC	0.867	0.657	0.760	0.719	0.881
LCD	0.870	0.761	0.856	0.823	0.896
AP	0.673	0.608	0.643	0.474	0.634

#### 3.1 Previous work

The two differently weighted CODEC versions, though verifiably producing decent results for clustering co-immunoprecipitation data [GS11], yield poor results for the detection of shared components in the Gavin and Krogan data sets. The specificity is low and indicates copious amounts of false positives.

MCODE results cannot compete with any of the explicit approaches as its Matthew correlation coefficient are comparably poor. If ever, the MCCs gained by MCODE are comparable to AG's results.

The exclusive use of complex finding algorithms is not satisfying the demands of shared component detection.

#### 3.2 Merged Maximal Cliques

MMC yields results with improved shared component detection quality compared to the implicit approaches considered here, particularly in sparsely connected networks like HC and Krogan. The higher the density of a network is, the less reliable the detection of shared components becomes, which is especially true for MMC. As the higher connectivity leads to a higher number of cliques to be merged, false positive rates increase. MMC is a straightforward algorithm for the detection of shared components in high quality interaction networks that performs remarkably well given the simplistic and strict approach and the high noise in experimentally determined networks.

### 3.3 Local Cluster Decomposition

For all regarded networks, LCD shows a consistently increased Matthews correlation coefficient over MMC. See Fig. 2, Table 1 and Table 2. LCD's superiority for finding shared components shows in dense networks like STRING700 and STRING900, which can be referred to the locally executed clustering in its process, resulting in lower false positive rates. As a whole, LCD gave the best results for all tested networks, thus representing the most appropriate algorithmic approach for detecting shared components. Its stringent and local nature appears to be well suited and could constitute a working definition for shared components.

### 3.4 Articulation Groups

Theoretically, two overlapping complexes contain a set of cut vertices, which would be identified by the articulation groups algorithm. Its practical application however disappoints: The results are much inferior to other algorithms and of a comparable quality to MCODE that uses a simple method to group nodes to complexes.

We explain the finding that many nodes in the network are articulation points or groups but do not correspond to shared components. Proteins in the periphery of a protein complex might be mapped to several complexes but are not an integral part of any complex [GAG<sup>+</sup>06]. Occasionally, interacting proteins of small degrees build chains. The resulting structures comprise articulation groups but neither graph theoretic nor biological considerations would identify them as protein complex.

The identification of shared components without consideration of clusters is apparently not solved by identifying sets of cut vertices and might not constitute an advisable strategy. It might be possible to include considerations of the degree of a vertex to improve the sensitivity.

The results of the brute force articulation groups computation are not particularly fruitful but sizes of different neighborhoods might provide enhancements.

### 3.5 Run-time

The time complexity of the algorithms is reflected in the practical run-times, which in term depend on the size of the network. LCD yielded results in under one hour for all networks. With our implementations, MMC as the second most complex algorithm typically takes several hours. Even with a parallelization of the AG algorithm, results required days for higher articulatory. The current implementations in Python could be optimized.

### 3.6 Discussion

It remains to be seen as to whether the shared components not part of CYC2008 set are bona fide shared components. Visual inspection suggests that several are but it would require substantial efforts in data integration to assess whether the results can be supported by external methods. It is unclear how shared components of protein complexes appear in gene expression studies or functional data sets. As complex membership is often used to annotate the function of a protein, simply using the provided functional annotation might be insufficient to make meaningful statements about the functional relevance.

## 4 Conclusion

Protein complex detection in interaction networks is difficult to improve without considering shared components. Shared components in protein interaction networks can be detected by several methods. The quality and experimental background of the interaction network influences the results. Explicit approaches like LCD outperform published implicit approaches and could be used to improve the detection of protein complexes in conjunction with disjoint clustering approaches.

## References

- [BH03] G. D. Bader and C. W. V. Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC bioinformatics*, 4(1):2, 2003.
- [BK73] C. Bron and J. Kerbosch. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577, 1973.
- [BRB<sup>+</sup>07] N. N. Batada, T. Reguly, A. Breitkreutz, L. Boucher, B. J. Breitkreutz, L. D. Hurst, and M. Tyers. Still status not altocumulus: further evidence against the date/party hub distinction. *PLoS Biol*, 5(6), 2007.
- [BVH06] S. Brohee and J. Van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC bioinformatics*, 7(1):488, 2006.
- [CK08] F. Cazals and C. Karande. A note on the problem of reporting maximal cliques. *Theoretical Computer Science*, 407(1-3):564–568, 2008.
- [CRGW98] H. J. Chial, M. P. Rout, T. H. Giddings, and M. Winey. *Saccharomyces cerevisiae* Ndc1p is a shared component of nuclear pore complexes and spindle pole bodies. *The Journal of cell biology*, 143(7):1789, 1998.
- [EVDO02] A. J. Enright, S. Van Dongen, and C. A. Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic acids research*, 30(7):1575, 2002.
- [FKZ08] C. Friedel, J. Krumsiek, and R. Zimmer. Bootstrapping the interactome: unsupervised identification of protein complexes in yeast. In *Research in Computational Molecular Biology*, pages 3–16. Springer, 2008.
- [GAG<sup>+</sup>06] A. C. Gavin, P. Aloy, P. Grandi, R. Krause, M. Boesche, M. Marzioch, C. Rau, L. J. Jensen, S. Bastuck, B. Duempefeld, and Others. Proteome survey reveals modularity of the yeast cell machinery. *Nature*, 440(7084):631–636, 2006.
- [GS11] G. Geva and R. Sharan. Identification of protein complexes from co-immunoprecipitation data. *Bioinformatics*, 27(1):111, 2011.
- [HT73] J. Hopcroft and R. Tarjan. Algorithm 447: efficient algorithms for graph manipulation. *Commun. ACM*, 16:372–378, June 1973.
- [KCY<sup>+</sup>06] N. J. Krogan, G. Cagney, H. Yu, G. Zhong, X. Guo, A. Ignatchenko, J. Li, S. Pu, N. Datta, A. P. Tikuisis, and Others. Global landscape of protein complexes in the yeast *Saccharomyces cerevisiae*. *Nature*, 440(7084):637–643, 2006.
- [KvMBD04] R. Krause, C. von Mering, P. Bork, and T. Dandekar. Shared components of protein complexes—versatile building blocks or biochemical artefacts? *BioEssays*, 26(12):1333–1343, December 2004.
- [NRYS02] H. H. Ng, F. Robert, R.A. Young, and K. Struhl. Genome-wide location and regulated recruitment of the RSC nucleosome-remodeling complex. *Genes & development*, 16(7):806, 2002.
- [PVE<sup>+</sup>07] S. Pu, J. Vlasblom, A. Emili, J. Greenblatt, and S. J. Wodak. Identifying functional modules in the physical interactome of *Saccharomyces cerevisiae*. *Proteomics*, 7(6):944–960, March 2007.
- [PWT<sup>+</sup>09] S. Pu, J. Wong, B. Turner, E. Cho, and S. J. Wodak. Up-to-date catalogues of yeast protein complexes. *Nucleic acids research*, 37(3):825, 2009.
- [RWR<sup>+</sup>98] C. Rongo, C. W. Whitfield, A. Rodal, S. K. Kim, and J. M. Kaplan. LIN-10 is a shared component of the polarized protein localization pathways in neurons and epithelia. *Cell*, 94(6):751–759, 1998.
- [SFK<sup>+</sup>11] D. Szklarczyk, A. Franceschini, M. Kuhn, M. Simonovic, A. Roth, P. Minguéz, T. Doerks, M. Stark, J. Muller, P. Bork, and Others. The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Research*, 39(suppl 1), 2011.
- [TTT06] E. Tomita, A. Tanaka, and H. Takahashi. The worst-case time complexity for generating all maximal cliques and computational experiments. *Theoretical Computer Science*, 363(1):28–42, 2006.
- [VW09] J. Vlasblom and S. Wodak. Markov clustering versus affinity propagation for the partitioning of protein interaction graphs. *BMC Bioinformatics*, 10(1):99+, March 2009.
- [ZPKS08] B. Zhang, B. H. Park, T. Karpinets, and N. F. Samatova. From pull-down data to protein interaction networks and complexes with biological relevance. *Bioinformatics*, 24(7):979, 2008.