Nature Methods

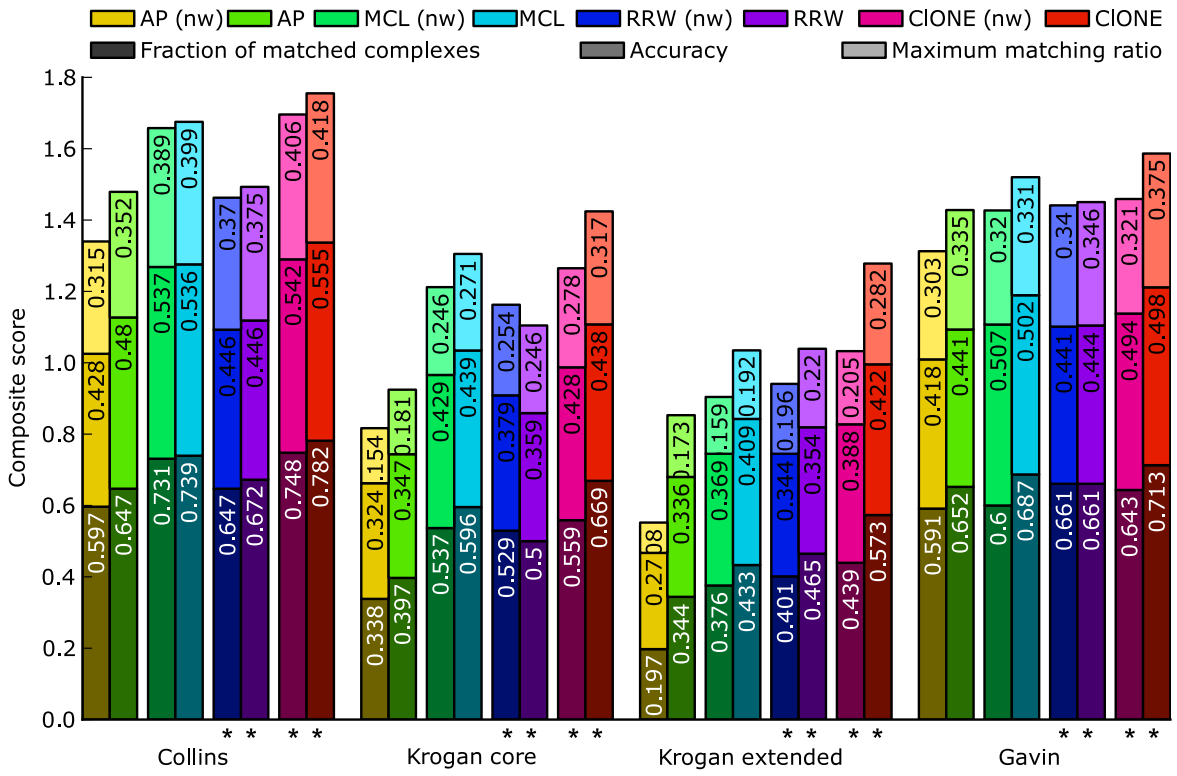
Detecting overlapping protein complexes in protein-protein interaction networks

Tamás Nepusz, Haiyuan Yu & Alberto Paccanaro

Supplementary Figure 1	Comparison of algorithm performance on weighted and unweighted versions of the same input dataset.
Supplementary Figure 2	Comparison of algorithm performance on unweighted datasets only.
Supplementary Figure 3	Comparison of algorithm performance on the SGD gold standard.
Supplementary Figure 4	The RSC and SWI/SNF complexes as detected by non-overlapping algorithms.
Supplementary Figure 5	The RSC and SWI/SNF complexes as detected by overlapping algorithms.
Supplementary Figure 6	The DASH complex as detected by the algorithms.
Supplementary Figure 7	
Supplementary Figure 8	
Supplementary Table 1	Co-localization and overrepresentation scores of the predicted complexes.
Supplementary Table 2	Overrepresentation scores of the predicted complexes when ignoring IPI evidence codes.
Supplementary Table 3	Properties of the protein-protein interaction datasets.
Supplementary Table 4	Properties of the gold standard datasets.
Supplementary Discussion	Supplementary discussion and results.

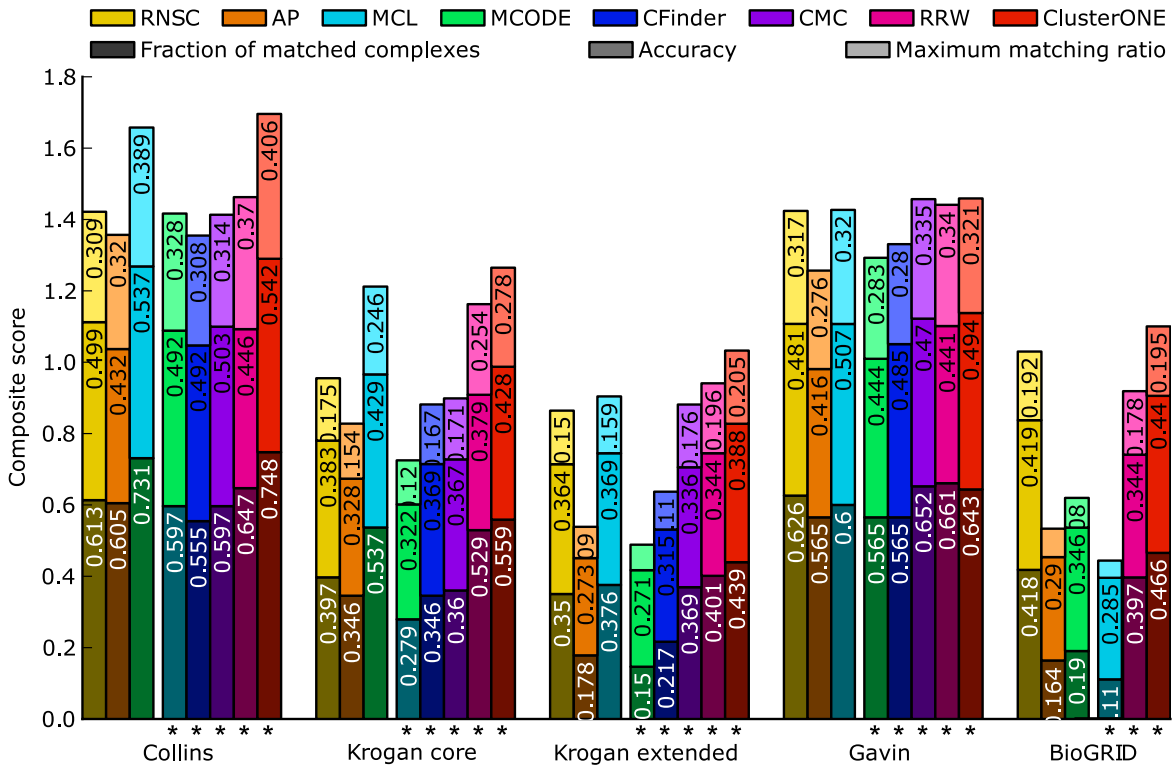
Note: Supplementary Data 1–2 are available on the Nature Methods website.

Nature Methods: doi:10.1038/nmeth.1938

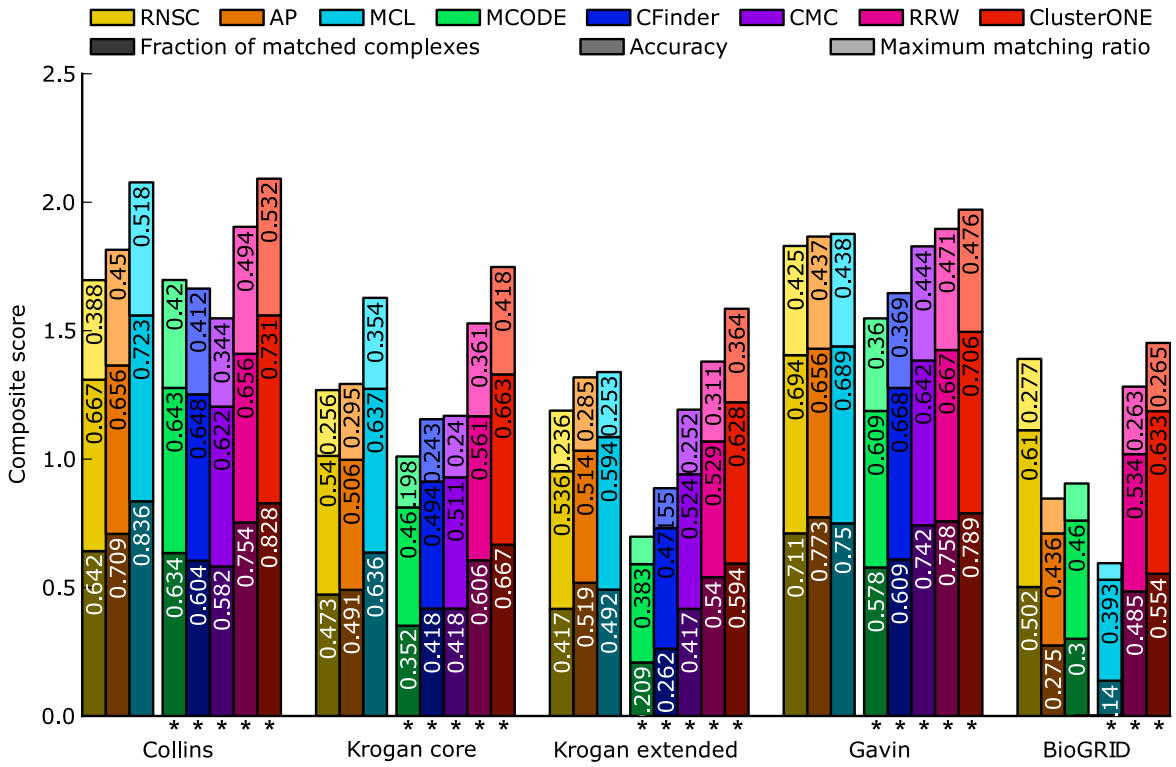


Supplementary Figure 1: Comparison of the performance of affinity propagation (AP), MCL, RRW and ClusterONE on weighted and unweighted versions of the same input dataset, using the MIPS gold standard. The *(nw)* suffix after the name of the algorithm denotes the unweighted variant. Overlapping algorithms are marked by an asterisk (*) below their columns.

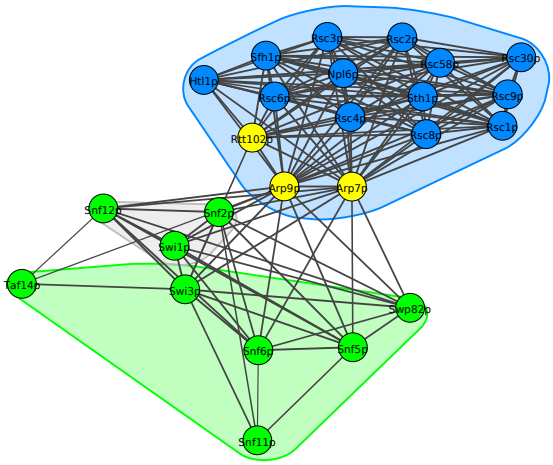
Nature Methods: doi:10.1038/nmeth.1938



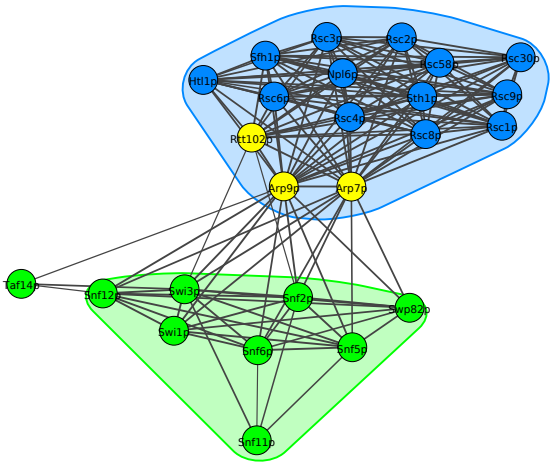
Supplementary Figure 2: Comparison of the performance of all the eight algorithms (RNSC, affinity propagation (AP), MCL, MCODE, CFinder, CMC, RRW and ClusterONE) on all the five unweighted datasets (the binarized versions of Collins, Krogan core, Krogan extended, Gavin, and BioGRID) using the MIPS gold standard. Overlapping algorithms are marked by an asterisk (*) below their columns.



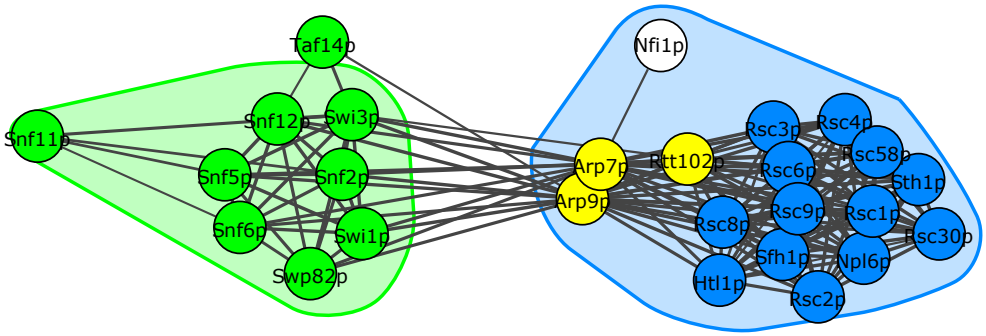
Supplementary Figure 3: Benchmark results of the tested algorithms on the SGD gold standard. Colors correspond to the various algorithms (red = ClusterONE), various shades of the same color denote the individual components of the composite score of the algorithm (dark = fraction of matched complexes, medium = geometric accuracy, light = maximum matching ratio). The numbers on the bars show the exact scores for each component of the composite score. The total height of each column is the value of the composite score for a given algorithm on a given dataset. Larger scores are better. Overlapping algorithms are marked by an asterisk (*) below their columns.



(a) Affinity propagation



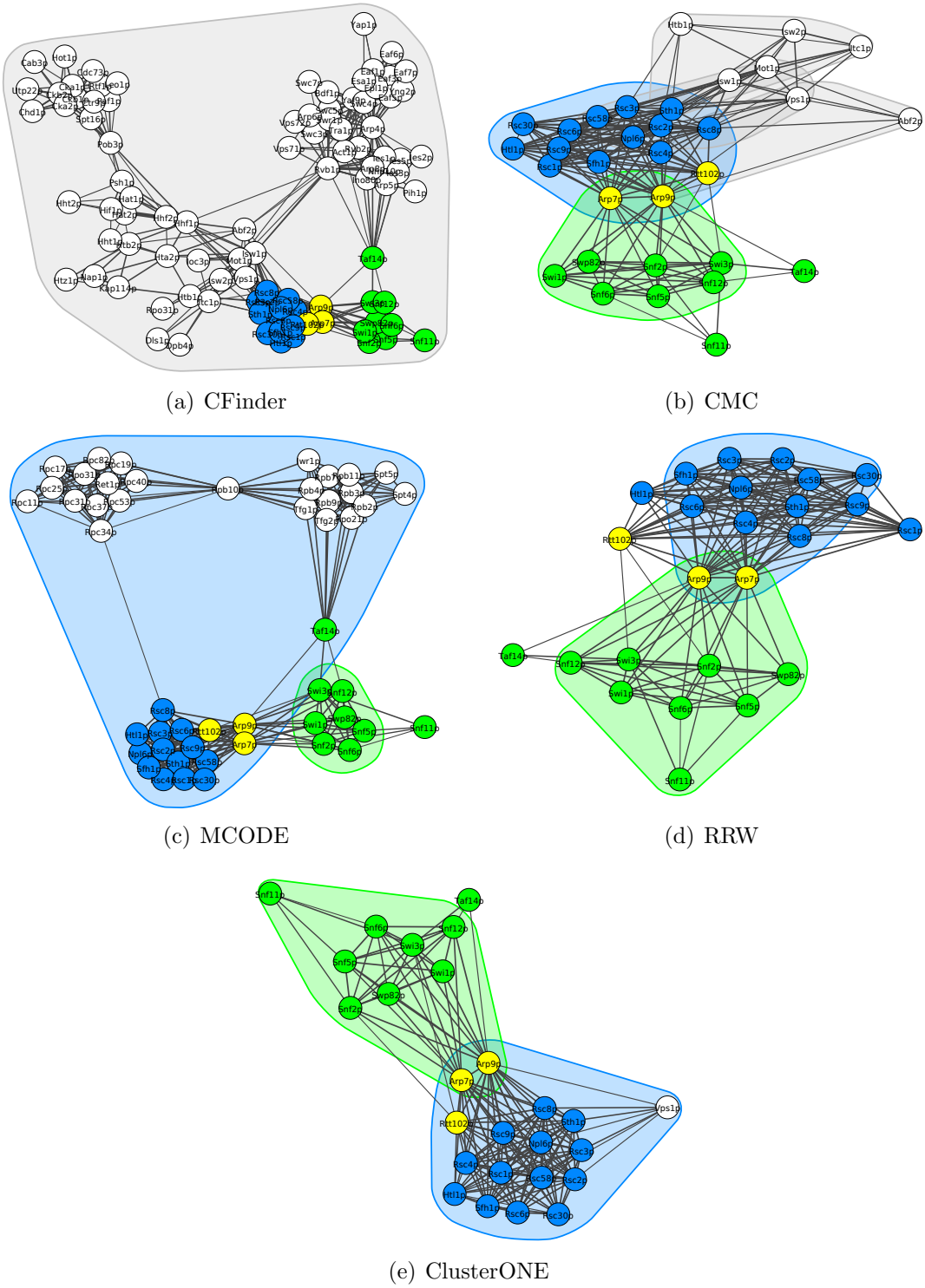
(b) RNSC



(c) MCL

Supplementary Figure 4: The RSC and SWI/SNF complexes as detected by the three non-overlapping clustering algorithms studied in the main manuscript. Blue and green nodes represent subunits of the RSC and SWI/SNF complexes, respectively; yellow nodes belong to both complexes, white nodes belong to neither. Shaded areas represent the clusters detected by the algorithms.

Nature Methods: doi:10.1038/nmeth.1938



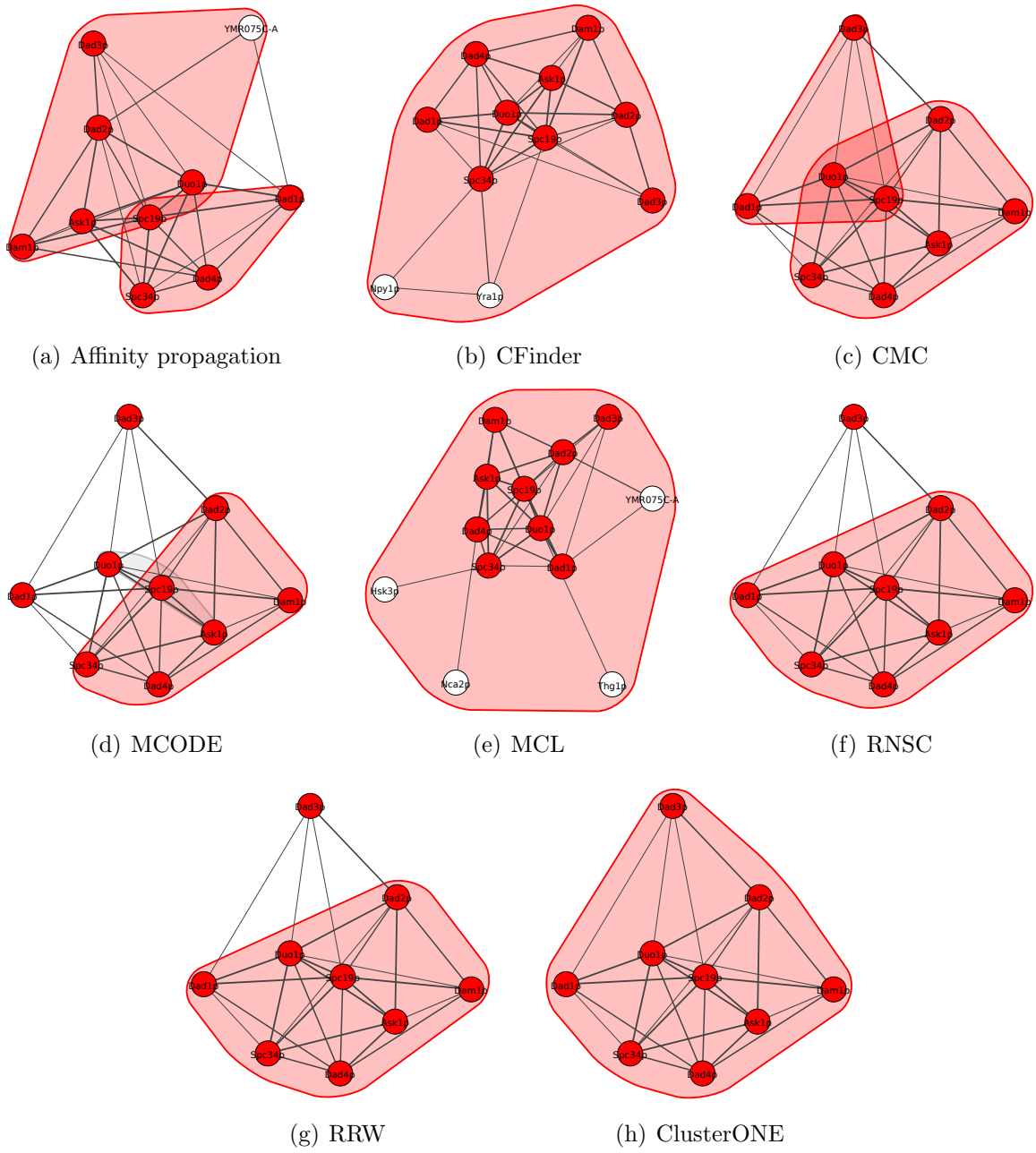
(a) CFinder (b) CMC

(c) MCODE (d) RRW

(e) ClusterONE

Supplementary Figure 5: The RSC and SWI/SNF complexes as detected by the five overlapping clustering algorithms studied in the main manuscript. Blue and green nodes represent subunits of the RSC and SWI/SNF complexes, respectively; yellow nodes belong to both complexes, white nodes belong to neither. Shaded areas represent the clusters detected by the algorithms.

Nature Methods: doi:10.1038/nmeth.1938



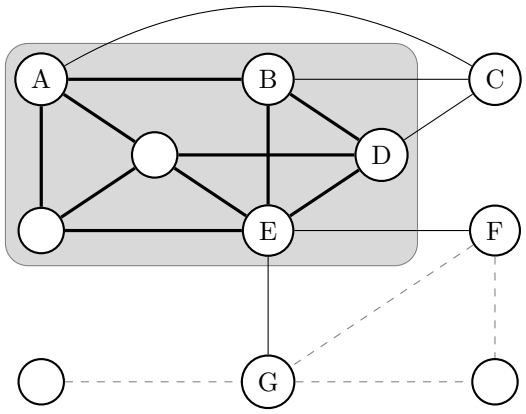
(a) Affinity propagation (b) CFinder (c) CMC

(d) MCODE (e) MCL (f) RNSC

(g) RRW (h) ClusterONE

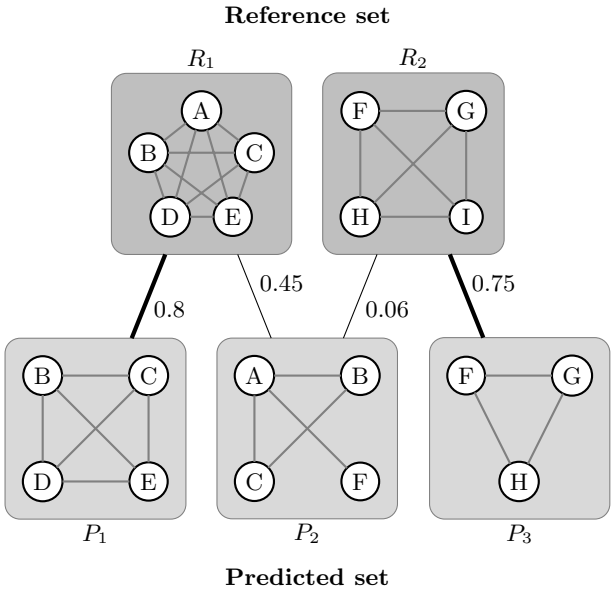
Supplementary Figure 6: The DASH complex as detected by the eight algorithms studied in the main manuscript. Red nodes represent subunits of the DASH complex in MIPS; shaded areas represent the clusters detected by the algorithms. The color of the shaded area is red if the cluster contains at least half of the proteins of the DASH complex.

Nature Methods: doi:10.1038/nmeth.1938



Supplementary Figure 7: Illustration of the greedy cohesive group detection process. The group itself is denoted by a shaded background. Thick black edges are internal, thin black edges are boundary edges, while thin gray dashed edges are completely external. Vertices marked by a letter are incident on at least one boundary edge, therefore only these vertices will be considered for addition or removal by the algorithm. The best choice is to extend the group by vertex C as it would convert three boundary edges to internal ones and would not add any additional boundary edges.

Nature Methods: doi:10.1038/nmeth.1938



Supplementary Figure 8: Illustration of the maximum matching ratio between a reference and a predicted complex set. R_1 and R_2 are members of the reference set, while P_1, P_2 and P_3 are three predicted complexes. An edge connects a reference complex and a predicted complex if their overlap score is larger than zero. The maximum matching is shown by the thick edges. Note that P_2 was not matched to R_1 since P_1 provides a better match with R_1. The maximum matching ratio in this example is $(0.8 + 0.75)/2 = 0.775$.

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 1: Co-localization and overrepresentation scores of the predicted complexes for ClusterONE and MCL when ignoring IEA, ND and NAS evidence codes

			Overrepresentation		
Dataset	Method	Co-localization	BP	CC	MF
Collins	ClusterONE	0.880	**0.882**	**0.826**	**0.735**
	MCL	**0.900**	0.869	0.776	0.712
Krogan core	ClusterONE	**0.775**	**0.670**	**0.525**	**0.499**
	MCL	0.746	0.580	0.473	0.443
Krogan extended	ClusterONE	**0.771**	**0.675**	**0.551**	**0.526**
	MCL	0.682	0.493	0.383	0.419
Gavin	ClusterONE	**0.883**	**0.918**	**0.827**	**0.733**
	MCL	0.812	0.676	0.632	0.510
BioGRID	ClusterONE	**0.741**	**0.821**	**0.751**	**0.657**
	MCL	0.605	0.618	0.538	0.539
MIPS complexes		0.813	0.995	0.965	0.893

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 2: Overrepresentation scores of the predicted complexes for ClusterONE and MCL when ignoring IEA, ND, NAS and IPI evidence codes

Dataset	Method	Overrepresentation		
		BP	CC	MF
Collins	ClusterONE	**0.882**	**0.754**	**0.708**
	MCL	0.858	0.721	0.694
Krogan core	ClusterONE	**0.658**	**0.477**	**0.482**
	MCL	0.570	0.437	0.423
Krogan extended	ClusterONE	**0.667**	**0.508**	**0.505**
	MCL	0.493	0.349	0.397
Gavin	ClusterONE	**0.913**	**0.769**	**0.638**
	MCL	0.668	0.591	0.494
BioGRID	ClusterONE	**0.794**	**0.634**	**0.672**
	MCL	0.604	0.500	0.494
MIPS complexes		0.995	0.921	0.897

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 3: Properties of the protein-protein interaction datasets used in the experiments.

	Collins	Krogan		Gavin	BioGRID
		core	extended		
Number of proteins	1,622	2,708	3,672	1,855	5,640
Number of interactions	9,074	7,123	14,317	7,669	59,748
Weighted	yes	yes	yes	yes	no
Threshold	top 9,074	0.273	0.101	5	N/A

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 4: Properties of the gold standard datasets used in the experiments.

	MIPS	SGD
Number of proteins	1,189	1,279
Number of complexes	203	323
Overlapping complex pairs	401 (2.0%)	296 (0.6%)
Proteins in ≥ 2 complexes	820 (69.0%)	332 (26.0%)

Nature Methods: doi:10.1038/nmeth.1938

Detecting overlapping protein complexes in protein-protein interaction networks

Tamás Nepusz, Haiyuan Yu and Alberto Paccanaro

Supplementary Discussion

Contents

Nature Methods: doi:10.1038/nmeth.1938

1 Assessing the quality of predicted complexes

1.1 Comparing predicted complexes with a gold standard

In the manuscript, we have used three independent quality measures to assess the similarity between a set of predicted complexes and a set of reference complexes. The first measure was the fraction of pairs between predicted and reference complexes with an overlap score ω larger than 0.25. Recall that the overlap score between two protein sets A and B is defined as follows [1]:

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

The threshold of 0.25 was chosen because it represents the case when the intersection is at least half of the complex size if the two complexes being compared are equally large.
The second measure was the maximum matching ratio (MMR), which we introduced in **Online Methods**. The measure is based on a maximal one-to-one mapping between predicted and reference complexes. See the **Online Methods** for a more precise description and Section 1.2 in this Supplementary Discussion for the motivation for developing MMR.
The third measure we used was the geometric accuracy as introduced by Brohée and van Helden [2], which is the geometric mean of two other measures, namely the clustering-wise sensitivity (Sn) and the clustering-wise positive predictive value (PPV). Sn and PPV are based on the confusion matrix $\mathbf{T} = [t_{ij}]$ of the complexes. Given n reference and m predicted complexes, let t_{ij} denote the number of proteins that are found both in reference complex i and predicted complex j, and let n_i denote the number of proteins in reference complex i. Sn and PPV are then defined as follows:

Nature Methods: doi:10.1038/nmeth.1938

$$\mathrm{Sn} = \frac{\sum_{i=1}^{n} \max_{j=1}^{m} t_{ij}}{\sum_{i=1}^{n} n_i}$$
$$\mathrm{PPV} = \frac{\sum_{j=1}^{m} \max_{i=1}^{n} t_{ij}}{\sum_{j=1}^{m} \sum_{i=1}^{n} t_{ij}}$$

Since the clustering-wise sensitivity can be inflated by putting every protein in the same cluster, while the positive predictive value can be maximized by putting every protein in its own cluster, it is necessary to balance the two measures by introducing the geometric accuracy (Acc), which is simply the geometric mean of the clustering-wise sensitivity and the positive predictive value:

$$\mathrm{Acc} = \sqrt{\mathrm{Sn} \times \mathrm{PPV}}$$

1.2 Motivation for the maximum matching ratio (MMR)

Our motivation for developing the MMR was that the positive predictive value (a component of the accuracy score) tends to be lower if there are substantial overlaps between the predicted complexes, and this puts overlapping clustering algorithms at a disadvantage. In the next few paragraphs, we will elaborate on this statement and provide an example which demonstrates this property of the geometric accuracy measure.

1.2.1 Caveats of the positive predictive value

The value of PPV can be misleading if some proteins in reference complex i appear in either more than one predicted complex or in none of them. In this case, n_i is not equal to the sum of row i in the confusion matrix $\mathbf{T}$. In general, n_i may be larger, smaller or equal to the sum of row i, which we will denote with t_{i*} from now on.
Consider the case when the set of reference and predicted complexes is the same. In this case, $t_{ii} = n_i$ for every i, but there may be other non-zero elements in $\mathbf{T}$, as $t_{ij} > 0$ if complex i and j overlap partially. However, these non-zero elements may never exceed t_{ii}, meaning that $\max_{j=1}^{m} t_{ij} = \max_{i=1}^{n} t_{ij} = n_i$ in all cases.
The Sn and PPV measures are then as follows:

$$\mathrm{Sn} = \frac{\sum_{i=1}^{n} n_i}{\sum_{i=1}^{n} n_i} = 1$$
$$\mathrm{PPV} = \frac{\sum_{i=1}^{n} n_i}{\sum_{i=1}^{n} t_{i*}} \leq 1$$

The consequence is that a perfect clustering algorithm that always returns the reference complexes from the data may have a *lower* positive predictive value than a dummy algorithm which places every protein in a separate cluster. In fact, assuming that we have k

Nature Methods: doi:10.1038/nmeth.1938

proteins, and protein j is a member of complex c_j, the positive predictive value for such a dummy algorithm would be:

$$\mathrm{PPV} = \frac{k}{\sum_{j=1}^{k} c_j} = 1$$

A concrete example is the set of MIPS complexes we used in our benchmarks. MIPS complexes containing at least three and at most 100 proteins cover a total of 1189 unique proteins, thus $k = 1189$ in the above example. However, the total size of such MIPS complexes (which is also equal to $\sum_{j=1}^{k} c_j$) is 2541, yielding a PPV of $1189/2541 = 0.468$ for the dummy algorithm which places every protein in a separate cluster, compared to the PPV of 0.3475 when comparing the MIPS complexes with themselves. Therefore, the PPV scores of the algorithms should be interpreted with care.

Finally, we would like to point out a substantial difference between the basic assumptions of the maximum matching ratio and the geometric accuracy. The geometric accuracy measure explicitly penalizes predicted complexes that do not match any of the reference complexes. However, gold standard sets of protein complexes are often incomplete [3]. As a consequence, predicted complexes not matching any known reference complexes may still exhibit high functional similarity or be highly co-localized, and therefore they could still be prospective candidates for further in-depth analysis. In other words, a predicted complex that does not match a reference complex is not necessarily an undesired result, and optimizing for the geometric accuracy measure might prevent us from detecting novel complexes from a PPI dataset. The maximum matching ratio sidesteps this problem by dividing the total weight of the maximum matching with the number of *reference* complexes. However, it is of course advised to quantify the functional homogeneity of the detected complexes with alternative methods to complement the maximum matching ratio, similarly to the approach we have followed in the manuscript. Some of these alternative methods are described later in Section 2.8.

1.2.2 Caveats of the clustering-wise separation

To solve some of the problems of the clustering-wise sensitivity and positive predictive value measures, Brohée and van Helden [2] have also suggested the *clustering-wise separation* measure as an alternative metric. In this section, we will discuss why the clustering-wise separation measure is also not a suitable measure for our problem – the reasons are similar to the ones given against the positive predictive value (PPV) measure.

The clustering-wise separation measure is defined as follows. First, let us define relative frequencies of the confusion matrix with respect to the marginal row-wise or column-wise sums as follows:

$$F_{ij}^{r} = \frac{t_{ij}}{\sum_{j=1}^{m} t_{ij}}$$

$$F_{ij}^{c} = \frac{t_{ij}}{\sum_{i=1}^{n} t_{ij}}$$

Nature Methods: doi:10.1038/nmeth.1938

The *separation* of predicted complex i and reference complex j is then given by:

$$\mathrm{Sep}_{ij} = F^r_{ij} F^c_{ij}$$

The *complex-wise* and the *cluster-wise separation* scores are then calculated for the whole set of reference and predicted complexes as follows:

$$\mathrm{Sep}_{co} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} \mathrm{Sep}_{ij}}{m}$$
$$\mathrm{Sep}_{cl} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} \mathrm{Sep}_{ij}}{n}$$

The *clustering-wise separation*, i.e. the final quality score is then given as the geometric mean of Sep_{co} and Sep_{cl}:

$$\mathrm{Sep} = \sqrt{\mathrm{Sep}_{co}\mathrm{Sep}_{cl}}$$

According to Brohée and van Helden [2],

> "The maximal value $\mathrm{Sep}_{ij} = 1$ indicates a perfect and exclusive correspondence between complex j and cluster i: it indicates that the cluster contains all the members of the complex and only them."

This again presents a problem in cases when the gold standard and/or the set of predicted complexes contains overlaps: calculating the clustering-wise separation of the MIPS complexes with themselves yields a clustering-wise separation of 0.3260 only. On one hand, the measure is correct since the MIPS complexes are not particularly well-separated from each other; on the other hand, it gives a misleading result because the MIPS complexes match themselves perfectly.
We also note that our *maximum matching ratio* measure (outlined in the Online Methods) is designed in a very similar vein to the clustering-wise separation score as it also starts with calculating a single quality score (the match score) for every reference-predicted complex pair; on the other hand, it proceeds by finding a maximum matching between reference and predicted complexes in a way that does not penalize overlaps explicitly.

2 Testing ClusterONE against other algorithms

2.1 General considerations

The standard procedure for evaluating the performance of a machine learning algorithm starts by dividing the data into a training and a testing set. The parameters of the algorithm are then tuned on the training set, and the optimal parameters are used to calculate the final performance score of the algorithm on the testing set. However, this procedure assumes that the input dataset can naturally be decomposed into problem *instances* such

Nature Methods: doi:10.1038/nmeth.1938

that 1) each instance is a complete input for the learning algorithm on its own and 2) each instance is independent of the other ones. Unfortunately, neither of these assumptions hold for graph clustering algorithms, especially in biological contexts where the input dataset consists of a single biological network which cannot be easily decomposed. In fact, it is reasonable to expect that removing a pre-defined fraction of edges from a network would change its structural properties in a way that could affect the outcome of a clustering algorithm substantially; in other words, removing edges from a network is similar to adding noise to a feature vector in a standard machine learning algorithm rather than to putting a set of problem instances aside in a testing set.
Evaluating the performance of clustering algorithms on graphs is therefore a tricky problem where we cannot turn to the well-established methodology of k-fold cross-validation. Nevertheless, careful counter-measures can and should be taken to avoid the typical biases in the evaluation of a novel method, in particular the over-optimization of algorithm parameters to a given dataset or a given quality score [4]. To this end, we have decided on the following:

1. We have tested each of the algorithms on five different datasets: three high-throughput experimental datasets [5, 6], a computationally derived network that integrates the results of these studies [7], and a compendium of all known yeast protein-protein interactions [8].

2. We have used more than one quality score to assess the performance of each algorithm: the fraction of matched complexes with a given overlap score threshold ($\omega \geq 0.25$), the geometric accuracy [2] and the maximum matching ratio that we have proposed.

3. We have also used two different gold standards: the MIPS compendium of protein complexes [9] and a set derived from the Gene Ontology annotations of the Saccharomyces Genome Database [10] (**Supplementary Data 2**). Note that since the two gold standards are not entirely consistent with respect to the membership of some proteins in some complexes, we decided to test these two gold standards separately.

4. For each algorithm, *except ClusterONE*, the final results were obtained after having optimized the algorithm parameters to yield the best possible results as measured by the maximum matching ratio on the gold standard that was being used in the benchmark (either MIPS or SGD). On the other hand, the results for ClusterONE were obtained without tuning the parameters. Therefore, the scores of ClusterONE represent its performance when the method is adapted to a wider problem domain (i.e. detecting overlapping protein complexes from high-throughput experimental PPI networks in general), while the scores of other algorithms measure their performance when they are optimized to a *specific* dataset. It is thus rightly expected that these latter scores are optimistic estimates.

Interestingly enough, we will see that there seems to be a clear separation between the behaviour of overlapping and non-overlapping algorithms in our benchmarks: the optimal

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 5: Characteristics of the various clustering algorithms studied in this paper.

Algorithm	Version	Weighted	Overlapping	Reference
Affinity propagation	Unknown (5 Dec 2007)	yes	no	[11]
CFinder	2.0.5	no[a]	yes	[12, 13]
ClusterONE	0.93	yes	yes	this manuscript
CMC	2.0	yes[b]	yes	[14]
MCL	10-201	yes	no	[15, 16]
MCODE	1.31[c]	no	yes	[1]
RNSC	Unknown (2004)	yes	no	[17]
RRW	Unknown (8 Aug 2011)	yes	yes	[18]

[a]CFinder has a weighted variant, but it turned out to be too slow for the protein complex detection task; see Section 2.5.2.

[b]Only in the initial stage; see Section 2.5.3.

[c]Patched to fix a bug with the density calculation; see Section 2.5.4

parameter values for non-overlapping algorithms such as affinity propagation, RNSC and MCL seem to vary wildly between datasets, meaning that parameters that work well for a given PPI network may not be suitable at all for a different dataset. On the other hand, overlapping algorithms seem to be less affected by the exact values of their parameters as long as they remain within a certain range. For ClusterONE, the default settings outperformed all the alternative approaches (see **Figure 1a** in the main manuscript and **Supplementary Figure 3**, **Supplementary Table 7** and **Supplementary Table 9** later in this document).

2.2 Common settings for all the algorithms

We compared the performance of ClusterONE to a representative set of other approaches: MCL, MCODE, affinity propagation, RNSC, CFinder, CMC and RRW. Some of these algorithms supported the use of edge weights (affinity propagation, MCL, RRW), and some could handle overlapping clusterings (MCODE, CFinder, CMC, RRW). **Supplementary Table 5** shows a summary of the different algorithm characteristics and the version numbers of the software we have used to test them.

In order to run algorithms not supporting arbitrary edge weights (MCODE, RNSC, CMC and CFinder) on weighted networks, these were first binarized using the threshold values originally suggested by the authors of the datasets. Interactions with a weight smaller than the proposed threshold were ignored; interactions with a weight larger than the proposed threshold were kept. We have checked the suitability of these thresholds using the heuristic proposed by Apeltsin et al [19].

Predicted complexes containing less than three proteins were excluded from the results unless the authors of the original algorithm suggested different size limits. In such cases,

Nature Methods: doi:10.1038/nmeth.1938

the new size limits are always mentioned explicitly in the upcoming sections.

2.3 Implementation details

Our goal was to make the implementation of ClusterONE as efficient and deterministic as possible. The key point in achieving efficiency was to recognize that one does not have to evaluate every single protein in the network when looking for a candidate protein to add to the cohesive subgroup being grown. This is achieved by maintaining two variables for each protein i in the network (where V_t will denote the current cohesive subgroup in step t):

1. The total weight of edges that connect protein i with members of V_t, denoted by w_i^{in}.
2. The total weight of edges that connect protein i with non-members of V_t, denoted by w_i^{out}.

We also maintain the so-called *boundary set* of V_t, which contains all the proteins with at least one incident edge that connects the protein with a member of V_t. In step t, it is enough to evaluate the boundary set of V_t and the members of V_t to find the protein whose addition or removal yields the largest increase in cohesiveness. The new cohesiveness after the addition of i can be calculated by using the current w^{in} and w^{bound} of V_t, and w_i^{in} and w_i^{out} as follows:

$$f(V_t \cup \{i\}) = \frac{w^{in}(V_t) + w_i^{in}}{w^{in}(V_t) + w^{out}(V_t) + w_i^{out} + p(|V_t| + 1)}$$

since

$$\begin{aligned} w^{in}(V_t \cup \{i\}) &= w^{in}(V_t) + w_i^{in} \\ w^{out}(V_t \cup \{i\}) &= w^{out}(V_t) - w_i^{in} + w_i^{out} \end{aligned}$$

Similar reasoning shows that for proteins already in V_t, the cohesiveness after their removal is as follows:

$$f(V_t \setminus \{i\}) = \frac{w^{in}(V_t) - w_i^{in}}{w^{in}(V_t) + w^{out}(V_t) - w_i^{out} + p(|V_t| - 1)}$$

w_i^{in} and w_i^{out} then has to be updated for every protein adjacent to protein i in the network. For sparse networks, this is significantly faster than calculating the cohesiveness scores from scratch for every protein that is considered for addition or removal.
In rare cases, it may happen that the addition of more than one vertex or the removal of more than one vertex would lead exactly to the same maximal increase in $f(V_t)$. Our reference implementation uses the following rules when resolving such ties:

1. If at least one addition and at least one removal would yield the same maximal increase in $f(V_t)$, only the additions are considered.

Nature Methods: doi:10.1038/nmeth.1938

2. If the addition of more than two vertices individually would result in the same maximal increase of the goal function, all such vertices are added at the same time.

3. If the removal of more than two vertices individually would result in the same maximal increase of the goal function, all such vertices are removed at the same time.

We have also tried several alternative resolution strategies used by other authors in algorithms based on similar growth processes (e.g., [20, 21, 22, 23, 24]), and we found that the exact rules of tie resolution do not affect the quality of the results substantially. We have settled on the above strategy in order to make ClusterONE as deterministic as possible. Finally, we would like to note that the source code of ClusterONE allows a developer to replace various parts of the algorithm easily without affecting others. For instance, several authors have proposed different kinds of quality functions for clusters: Lancichinetti et al [22] proposed a resolution parameter μ that can be used to tune the granularity of clusters, while Fortney et al [24] showed an example of how additional biological knowledge (in their case, gene expression data) can be incorporated into a cohesiveness-like function. Both of these goal functions can be used with the default growth process built into ClusterONE by replacing the classes responsible for the calculation of the goal function. The HC-PIN algorithm of Wang et al [23] uses a bottom-up iterative merging procedure, which stops when a measure similar to cohesiveness falls below a given threshold; this could also easily be implemented by modifying ClusterONE's source code and replacing the class responsible for the greedy growth process with a hierarchical merging procedure.

2.4 ClusterONE parameter settings

For ClusterONE, we did not tune the parameters to a particular dataset and we used the default parameters of our implementation. These were: density threshold set to 0.3 for weighted networks and to either 0.6 or 0.5 for unweighted networks (depending on the network transitivity, see next subsection). The merging threshold was set to 0.8 and the penalty value of each node was 2.

2.4.1 Transitivity and the BioGRID dataset

In our experiments, we tested the various algorithms on two types of unweighted datasets: those obtained by binarizing the weighted networks (from Collins, Krogan and Gavin) and BioGRID. Importantly these dataset were derived with different experimental techniques: the Collins, Krogan and Gavin 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. This makes the BioGRID network structurally very different, and particularly it shows an unexpectedly high fraction of star-like structures. One way to quantify this, is to count the probability of triangles (i.e. triads of proteins all interacting with each other) given three proteins connected by at least two edges. This measure is known as transitivity (or also global clustering coefficient). In other words, transitivity tells us the probability of finding a third edge among triplets of proteins where

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 6: Transitivity scores of yeast two-hybrid, TAP tagging / mass spectrometry and curated protein-protein interaction datasets.

Dataset	Ref.	Transitivity
Yeast two-hybrid		
Ito et al, 2001	[25]	0.006
Uetz et al, 2000	[26]	0.040
TAP tagging / mass spectrometry		
Krogan et al, 2006, extended	[6]	0.100
Krogan et al, 2006, core	[6]	0.195
Gavin et al, 2002	[27]	0.196
Gavin et al, 2006	[5]	0.560
Collins et al, 2007	[7]	0.619
Database curation		
BioGRID	[8]	0.066

at least two of the possible three connections exists. Therefore, transitivity has a low value if the network contains many star-like structures. **Supplementary Table 6** contains the transitivity scores of several protein-protein interaction datasets, including the ones analyzed in this manuscript. Differences in structural properties of these networks are also pointed out in [28, 29].
A low value of the transitivity implies a high presence of star-like structures in the interaction network and these hamper the effectiveness of ClusterONE's filtering based on density (the last step of our algorithm). In these cases we recommend that use higher value for the density threshold in order to discard trivial clusters. Given an unweighted network, ClusterONE automatically tests the value of the transitivity and sets the density threshold to either 0.5 or 0.6 (for the BioGRID dataset it uses 0.6).

2.5 Description of the other algorithms' parameters

2.5.1 Affinity propagation

Each data point in the affinity propagation algorithm [11] has a parameter called *preference*, which controls the likelihood of that data point being an *exemplar* (i.e. a representative element of a cluster). It is a common practice to set the preference value equal for all data points; in this case, one can think of the algorithm as having a single parameter only. In our benchmarks, the optimal preference value was determined by sampling the interval [-1; 1] uniformly with a step size of 0.1 and settling on the preference value that results in the best quality score. In our benchmarks, we used the 64-bit Linux version of affinity propagation, downloaded from `http://www.psi.toronto.edu/index.php?q=affinity%20propagation`.

Nature Methods: doi:10.1038/nmeth.1938

2.5.2 CFinder

CFinder [12, 13] was one of the first overlapping clustering methods published in the literature. The original version of the algorithm operates on undirected, unweighted networks. CFinder finds all k-cliques of the original network (where k is a tunable parameter of the algorithm) and constructs a k-clique accessibility graph where two k-cliques are considered accessible from each other if they share $k-1$ vertices. The connected components of the k-clique accessibility graph (whose vertices represent cliques in the original network) are then used to derive the overlapping communities. A detailed description of the algorithm is to be found in [12].

One may easily recognise that it is enough to enumerate the maximal cliques of the original network that have at least k vertices instead of finding all k-cliques, as each subset of a maximal clique is also a clique, therefore a maximal clique of size n will be mapped to a connected subgraph consisting of $\binom{n}{k}$ vertices in the k-clique accessibility graph. Such subgraphs can be shrank into a single vertex that will represent the whole maximal clique without affecting the connectivity properties of the k-clique accessibility graph.

Later on, a weighted extension of CFinder was proposed in [30]. This variant introduces a second parameter I, acting as an intensity threshold for the detected cliques: the product of the edge weights in a clique must exceed I in order to include that clique in the accessibility graph. This is computationally more prohibitive than the unweighted variant as we have to enumerate all k-subcliques of maximal cliques and check their intensities explicitly. In fact, the reference implementation of CFinder (as downloaded from `http://www.cfinder.org` on 8 Aug 2010) did not provide a result for the Collins dataset for a fairly conservative setting of $k = 4$ and $I = 0.8$ in 48 hours, therefore we decided to proceed with the unweighted variant in our benchmarks.

2.5.3 CMC

The CMC algorithm [14] is based on an iterative scoring algorithm that assesses the probability of whether two given proteins are in the same complex, followed by a maximal clique finding process. Highly overlapping cliques are then merged in order to achieve the final set of complexes. The algorithm is primarily governed by the *overlap threshold* which determines when should two cliques be considered highly overlapping, and the *merge threshold*, which determines what to do with two highly overlapping cliques: they will be merged if the part of the network between the two complexes is denser than the merge threshold, otherwise the smaller clique will be discarded.

The range of both parameters is between zero and one, although low overlap thresholds do not make sense as they would result in only a few giant complexes. Similarly, high overlap thresholds would result in a very large number of redundant complexes, as almost none of them would be allowed to merge with others. Therefore, the tested range of the overlap threshold was limited to real values between 0.2 and 0.8, sampled with a step size of 0.1. The merge threshold was tested on uniformly sampled real values between 0 and 1 with a step size of 0.1.

Nature Methods: doi:10.1038/nmeth.1938

In our benchmarks, we used the original implementation of the CMC software (version 2), downloaded from `http://www1.comp.nus.edu.sg/~wongls/projects/complexprediction/CMC-26may09/`. According to the suggestions of the authors of the algorithm [14], a size limit of 4 was used instead of the default size limit. N/A for the BioGRID dataset indicates that the algorithm produced a prohibitively large number of clusters (more than 6000) for all parameter settings we have tried.

2.5.4 MCODE

The MCODE algorithm [1] consists of three phases: vertex weighting, protein complex formation and post-processing. The vertex weighting phase assigns a score to each vertex measuring the "cliquishness" of the neighborhood of the vertex. Protein complexes are then grown from each vertex, starting from the one with the highest weight. The *depth limit* parameter controls how far the growth process is willing to proceed from the seed vertex when considering other vertices to be added to the seed vertex in order to form a protein complex. The *vertex weight percentage* controls how much difference is allowed between the scores of the vertices within the same complex. Finally, there are two possible post-processing operations: *haircut*, which iteratively removes vertices that are connected by only a single edge to the rest of the complex, and *fluffing*, which tries to expand the complex with other vertices if they connect to many vertices of the same complex. MCODE is able to produce overlapping complexes in the fluffing phase, but our experiments have shown that the algorithm performs better when fluffing is turned off.
We tried all the possible combinations of the following parameters:

- Depth limit: 3, 4, 5
- Vertex weight percentage: 10% to 50% in steps of 5%
- Haircut: on or off
- Fluffing: on or off
- Fluffing percentage: 0, 10% or 20%

At the time of the submission of this manuscript, the most recent version of the MCODE Cytoscape plugin (version 1.31) calculated the complex densities and hence the complex scores incorrectly, hence we used a patched version of the plugin that fixes this issue. The complex score calculation in the patched version was exactly according to the original publication [1].

2.5.5 MCL

The MCL algorithm [16] has a single parameter called *inflation*, which tunes the granularity of the clustering. Larger inflation values result in smaller clusters, while smaller inflation values generate only a few large clusters. The range of possible inflation values for the

Nature Methods: doi:10.1038/nmeth.1938

MCL algorithm (1.2 to 5.0) was sampled uniformly with a step size of 0.1. The optimal inflation values for each dataset are as follows:
In our benchmarks, we used MCL version 10-201, downloaded from `http://www.micans.org/mcl`.

2.5.6 RNSC

The RNSC algorithm [17] has a large number of tunable parameters, but according to the benchmarks of Brohée et al [2], none of these parameters influence the overall quality of the clusters substantially in the case of protein-protein interaction datasets. Following the approach chosen by Brohée et al [2], we tried all the possible combinations of the following parameter values:

- Shuffling diversification length: 3, 5, 9
- Diversification frequency: 10, 20, 50
- Number of experiments: 1, 3, 10
- Naive stopping tolerance: 10, 20, 50
- Scaled stopping tolerance: 1, 5, 15
- Tabu length: 1, 10, 50, 100
- Tabu tolerance: 1, 3, 5

The total number of parameter combinations tried was 2916. Since the RNSC algorithm is randomized, each combination was tried 5 times for each dataset and the one resulting in the best maximum matching ratio was kept. According to the suggestions of the authors of the algorithm [17], a size limit of 4 was used.

2.5.7 RRW

The RRW algorithm [18] derives complexes from results of repeated restarted random walks on the graph of protein-protein interactions. It requires one to specify the minimum and maximum size of the clusters, the restart probability of the random walk at each step, and two threshold parameters (the overlap threshold and the early cutoff). We note that the authors recommended a maximum cluster size of 11 in their original publication. Such a threshold obviously prevents RRW from accurately detecting complexes that are larger than this size; however, preliminary experiments with maximum cluster sizes of 50, 100 and 150 indicated that the results obtained with these settings were in general worse than using a more conservative maximum cluster size, while also taking substantially more time (10–15 minutes for a single run, depending on the network size and the cluster size threshold). Therefore, we left the maximum size at 11 and tuned the remaining parameters by trying all possible combinations of the following values:

Nature Methods: doi:10.1038/nmeth.1938

- Restart probability: 0.5 to 0.9 in steps of 0.1
- Overlap threshold: 0.05 to 0.3 in steps of 0.05
- Early cutoff: 0.5 to 0.9 in steps of 0.1

2.6 The MIPS gold standard

2.6.1 Parameter settings for each algorithm

Affinity propagation

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Preference	-0.9	0.35	0.4	-0.15	-0.5

CFinder

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
k-clique template size	3	3	3	4	N/A

N/A for the BioGRID dataset indicates that even the unweighted CFinder implementation did not give any result within 24 hours.

CMC

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Overlap threshold	0.7	0.7	0.7	0.7	N/A
Merge threshold	0.5	0.4	0.5	0.5	N/A

MCODE

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Depth limit	3	3	3	3	3
Vertex weight percentage	20%	20%	10%	10%	10%
Fluff complexes	no	no	no	no	no
Fluff threshold	N/A	N/A	N/A	N/A	N/A
Haircut complexes	yes	yes	yes	yes	yes

Nature Methods: doi:10.1038/nmeth.1938

MCL

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Inflation	4.9	2.3	2.3	3.2	3.3

RNSC

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Shuffling diversification length	5	9	9	9	9
Diversification frequency	50	20	20	20	20
Number of experiments	3	3	3	3	3
Naive stopping tolerance	10	10	20	20	20
Scaled stopping tolerance	5	5	1	5	5
Tabu length	100	10	50	100	10
Tabu tolerance	1	3	1	5	1

RRW

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Restart probability	0.5	0.5	0.5	0.6	0.9
Overlap threshold	0.2	0.2	0.2	0.1	0.2
Early cutoff	0.5	0.6	0.7	0.6	0.6

2.6.2 Benchmark results

First we have tested all the algorithms mentioned above on the weighted Collins, Krogan and Gavin datasets and on the unweighted BioGRID dataset (**Supplementary Data 1**). **Supplementary Table 7** contains the detailed benchmark results when the MIPS gold standard dataset was used as a gold standard.

2.6.3 The effect of random matches

In this section, we provide estimates for the expected values of each quality score when applied to randomized predicted complex sets. These estimates will then serve as a baseline against which we can compare our actual, observed quality scores. Quality scores close to the baseline would indicate cases when a particular algorithm managed to produce a complex set that is not substantially better than a randomly created complex set.
In order to randomize a predicted complex set while maintaining the size distribution of the clusters, we first concatenated these complexes into a list of proteins. This list was

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 7: Benchmark results of various protein complex detection algorithms on PPI datasets using the MIPS gold standard

Dataset	Method	#clusters	#matched	Sn	PPV	Accuracy	Matching ratio
Collins	MCL	183	88	0.587	0.490	0.536	0.399
	MCODE	112	71	0.519	0.467	0.492	0.328
	CMC	184	71	0.498	**0.508**	0.503	0.314
	AP	152	76	0.459	0.503	0.480	0.352
	ClusterONE	195	**93**	0.639	0.483	**0.555**	**0.418**
	RNSC	94	73	0.516	0.482	0.499	0.309
	RRW	190	80	0.396	0.503	0.446	0.375
	CFinder	114	66	**0.661**	0.367	0.492	0.308
Krogan core	MCL	376	81	0.450	0.428	**0.439**	0.271
	MCODE	79	38	0.283	0.367	0.322	0.123
	CMC	156	49	0.315	0.429	0.367	0.171
	AP	222	54	0.272	0.437	0.345	0.174
	ClusterONE	522	**91**	0.486	0.396	0.438	**0.317**
	RNSC	87	54	0.321	**0.458**	0.383	0.175
	RRW	329	68	0.296	0.436	0.359	0.246
	CFinder	115	47	**0.532**	0.256	0.369	0.167
Krogan extended	MCL	483	68	0.411	0.408	0.409	0.192
	MCODE	64	23	0.199	0.369	0.271	0.071
	CMC	421	58	0.295	0.383	0.336	0.176
	AP	234	53	0.255	0.436	0.333	0.166
	ClusterONE	530	**90**	0.443	0.402	**0.422**	**0.282**
	RNSC	93	55	0.301	0.439	0.364	0.150
	RRW	232	73	0.283	**0.444**	0.354	0.220
	CFinder	121	34	**0.611**	0.162	0.315	0.106
Gavin	MCL	253	79	0.508	0.497	**0.502**	0.331
	MCODE	135	65	0.426	0.464	0.444	0.283
	CMC	339	75	0.480	0.460	0.470	0.335
	AP	246	75	0.396	0.490	0.441	0.335
	ClusterONE	196	**82**	0.519	0.479	0.498	**0.375**
	RNSC	138	72	0.484	0.478	0.481	0.317
	RRW	234	76	0.395	**0.498**	0.444	0.346
	CFinder	137	65	**0.577**	0.409	0.485	0.280
BioGRID	MCL	338	37	0.346	0.350	0.348	0.083
	MCODE	85	21	0.285	0.284	0.285	0.048
	AP	586	46	0.227	0.373	0.291	0.096
	ClusterONE	473	**88**	**0.454**	0.427	**0.440**	**0.195**
	RNSC	209	79	0.399	**0.441**	0.419	0.192
	RRW	253	75	0.276	0.429	0.344	0.178

Abbreviations: Sn = Clustering-wise sensitivity, PPV = Clustering-wise positive predictive value

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 8: Expected values of the fraction of matched complexes (top), the geometric accuracy (middle) and the maximum matching ratio (bottom) when a randomized clustering is compared with the MIPS gold standard.

Frac.matched	Collins	Krogan core	Krogan extd	Gavin	BioGRID
AP	0.0012	0.0012	0.0013	0.0022	0.0005
CFinder	0.0007	0.0013	0.0009	0.0004	–
ClusterONE	0.0014	0.0024	0.0017	0.0011	0.0012
CMC	0.0007	0.0019	0.0080	0.0014	–
MCL	0.0014	0.0009	0.0008	0.0016	0.0000
MCODE	0.0017	0.0007	0.0005	0.0010	0.0002
RNSC	0.0005	0.0012	0.0007	0.0011	0.0004
RRW	0.0026	0.0026	0.0019	0.0025	0.0010

Accuracy	Collins	Krogan core	Krogan extd	Gavin	BioGRID
AP	0.1418	0.1454	0.1433	0.1583	0.1600
CFinder	0.1998	0.2258	0.2459	0.1511	–
ClusterONE	0.1590	0.1619	0.1540	0.1466	0.1677
CMC	0.1523	0.1212	0.1341	0.1591	–
MCL	0.1557	0.1636	0.1632	0.1606	0.2081
MCODE	0.1421	0.1169	0.1077	0.1418	0.1090
RNSC	0.1334	0.1110	0.1053	0.1411	0.1207
RRW	0.1530	0.1678	0.1469	0.1611	0.1319

MMR	Collins	Krogan core	Krogan extd	Gavin	BioGRID
AP	0.0393	0.0374	0.0379	0.0503	0.0467
CFinder	0.0321	0.0274	0.0242	0.0315	–
ClusterONE	0.0408	0.0450	0.0410	0.0385	0.0460
CMC	0.0281	0.0221	0.0294	0.0415	–
MCL	0.0441	0.0487	0.0441	0.0474	0.0242
MCODE	0.0335	0.0201	0.0137	0.0362	0.0116
RNSC	0.0232	0.0167	0.0144	0.0316	0.0226
RRW	0.0508	0.0550	0.0408	0.0535	0.0348

Nature Methods: doi:10.1038/nmeth.1938

then shuffled using the Fisher–Yates shuffle [31], and divided into groups in a way that preserved the sizes of the original complexes. The value of each quality score can then be computed for this grouping. This procedure was repeated 100 times, to estimate the expected value for each quality score. Since the Fisher–Yates shuffle chooses any possible permutation of a list with equal probability, the resulting set of randomized complex sets can be used to obtain an unbiased estimate for the expected value of any chosen quality score (given the number of predicted complexes and their sizes, which are not varied).
We estimated these values for each algorithm on each dataset. **Supplementary Table 8** shows the expected values of the three components of our composite score (i.e. the fraction of matched complexes, the geometric accuracy and the maximum matching ratio). We can conclude that the expected values are very close to zero in the case of the fraction of matched complexes (< 0.01) and the maximum matching ratio (≤ 0.06) for all datasets and all algorithms. The expected value of the geometric accuracy is higher, but nevertheless still smaller than the values we have observed in our benchmark results (**Supplementary Table 7**), which confirm that the correspondence between the predicted and the reference complexes is not merely a result of chance.

2.6.4 An example: the RSC and SWI/SNF complexes

Supplementary Figure 4 and **Supplementary Figure 5** illustrate how a particular overlapping complex pair (the RSC and the SWI/SNF complexes) is found by the clustering algorithms we have studied. **Supplementary Figure 4** shows the results of the non-overlapping algorithms, while **Supplementary Figure 5** shows the results of the overlapping algorithms. It can be seen that non-overlapping methods like affinity propagation (AP), MCL and RNSC grouped the three overlapping subunits (Rtt102p, Arp9p and Arp7p, shown in yellow) together with the RSC complex unequivocally. Among the overlapping methods, ClusterONE, RRW and CMC managed to identify Arp9p and Arp7p as overlaps between the two complexes, but RRW missed some of the other subunits of the RSC complex, and CMC produced many false overlaps between the RSC complex and clusters of other proteins that do not constitute valid reference complexes. CFinder grouped both RSC and SWI/SNF together in a large cluster containing more than 80 proteins, while MCODE grouped RSC together with the DNA-directed RNA polymerase II and III complexes.

2.6.5 An example: the DASH complex

Supplementary Figure 6 shows how the various algorithms studied in this manuscript identify the DASH complex, an important microtubule-binding component of the kinetochore [32] from the Krogan extended dataset. The complex is characterised in MIPS as containing 9 proteins. ClusterONE is the only algorithm that is able to detect this complex completely correctly. All other algorithms made various mistakes: 1) RRW, MCODE, and RNSC were only able to detect part of the whole complex; 2) MCL and CFinder clustered unrelated proteins into the DASH complex; more interestingly, the erroneous pro-

Nature Methods: doi:10.1038/nmeth.1938

teins added by the two algorithms are completely different; 3) CMC separated the DASH complex into two overlapping complexes in a non-biologically relevant way; 4) Affinity propagation not only separated the complex into two parts, but also added an erroneous component – YMR075C-A, a dubious ORF unlikely to produce a protein product.

2.7 The SGD gold standard

2.7.1 Parameter settings for each algorithm

We have used the same parameter tuning method as the one outlined in Section 2.6.1. The optimal parameter settings for each algorithm are provided in the next paragraphs.

Affinity propagation

	Collins [7]	Krogan [6]		Gavin [5]	BioGRID [8]
		core	extended		
Preference	0.4	0.35	0.3	-0.6	-0.7

CFinder

	Collins [7]	Krogan [6]		Gavin [5]	BioGRID [8]
		core	extended		
k-clique template size	3	3	4	4	N/A

N/A for the BioGRID dataset indicates that CFinder did not give any result within 24 hours.

CMC

	Collins [7]	Krogan [6]		Gavin [5]	BioGRID [8]
		core	extended		
Overlap threshold	0.7	0.7	0.7	0.7	N/A
Merge threshold	0.5	0.4	0.3	0.5	N/A

N/A for the BioGRID dataset indicates that the algorithm produced a prohibitively large number of clusters (more than 6000) for all parameter settings we have tried.

MCODE

Nature Methods: doi:10.1038/nmeth.1938

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Depth limit	3	3	3	3	3
Vertex weight percentage	20%	20%	10%	10%	10%
Fluff complexes	no	no	no	no	no
Fluff threshold	N/A	N/A	N/A	N/A	N/A
Haircut complexes	yes	yes	yes	yes	yes

MCL

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Inflation	4.6	2.0	2.6	4.7	3.2

RNSC

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Shuffling diversification length	9	3	9	9	9
Diversification frequency	50	20	50	10	10
Number of experiments	3	3	10	3	1
Naive stopping tolerance	50	50	50	20	20
Scaled stopping tolerance	5	5	1	15	15
Tabu length	100	50	50	100	1
Tabu tolerance	1	1	1	1	1

RRW

	Collins [7]	Krogan [6] core	Krogan [6] extended	Gavin [5]	BioGRID [8]
Restart probability	0.5	0.5	0.5	0.6	0.9
Overlap threshold	0.2	0.2	0.2	0.1	0.2
Early cutoff	0.5	0.6	0.7	0.6	0.6

2.7.2 Benchmark results

We have tested all the algorithms mentioned in the previous section on the weighted Collins, Krogan and Gavin datasets and on the unweighted BioGRID dataset (**Supplementary Data 1**). **Supplementary Table 9** and **Supplementary Figure 3** contain the detailed benchmark results when the complexes derived from SGD were used as a gold standard.

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 9: Detailed benchmark results of various protein complex detection algorithms on PPI datasets using the SGD complex set

Dataset	Method	#clusters	#matched	Sn	PPV	Accuracy	Matching ratio
Collins	MCL	181	**112**	0.797	0.657	0.723	0.518
	MCODE	112	85	0.707	0.585	0.643	0.420
	CMC	184	78	0.658	0.588	0.622	0.344
	AP	151	95	0.645	0.666	0.656	0.450
	ClusterONE	195	111	0.827	0.646	**0.731**	**0.532**
	RNSC	95	86	0.698	0.638	0.667	0.388
	RRW	190	101	0.614	**0.702**	0.656	0.494
	CFinder	114	81	**0.858**	0.489	0.648	0.412
Krogan core	MCL	367	105	0.686	0.592	0.637	0.354
	MCODE	79	58	0.420	0.504	0.460	0.198
	CMC	156	69	0.461	0.566	0.511	0.240
	AP	216	81	0.409	0.627	0.506	0.295
	ClusterONE	522	**110**	**0.703**	0.626	**0.663**	**0.418**
	RNSC	88	78	0.482	0.604	0.540	0.256
	RRW	264	100	0.465	**0.676**	0.561	0.361
	CFinder	115	69	0.680	0.359	0.494	0.243
Krogan extended	MCL	517	92	0.584	0.604	0.594	0.253
	MCODE	64	39	0.326	0.450	0.383	0.107
	CMC	351	78	0.484	0.567	0.524	0.252
	AP	266	97	0.415	0.637	0.514	0.285
	ClusterONE	530	**111**	**0.635**	0.620	**0.628**	**0.364**
	RNSC	97	78	0.471	0.610	0.536	0.236
	RRW	232	101	0.433	**0.645**	0.529	0.311
	CFinder	88	49	0.550	0.401	0.470	0.155
Gavin	MCL	253	96	0.697	0.681	0.689	0.438
	MCODE	135	74	0.612	0.607	0.609	0.360
	CMC	339	95	0.710	0.580	0.642	0.444
	AP	239	99	0.642	0.671	0.656	0.437
	ClusterONE	196	**101**	0.783	0.637	**0.706**	**0.476**
	RNSC	143	91	0.731	0.658	0.694	0.425
	RRW	237	97	0.636	**0.699**	0.667	0.471
	CFinder	137	78	**0.817**	0.545	0.668	0.369
BioGRID	MCL	335	70	0.447	0.474	0.460	0.144
	MCODE	85	32	0.400	0.386	0.393	0.065
	AP	606	64	0.361	0.527	0.436	0.136
	ClusterONE	481	**129**	**0.699**	0.573	**0.633**	0.265
	RNSC	220	117	0.611	0.610	0.610	**0.277**
	RRW	270	113	0.467	**0.611**	0.534	0.263

Abbreviations: Sn = Clustering-wise sensitivity, PPV = Clustering-wise positive predictive value

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 10: General properties of the clusterings generated by ClusterONE.

	Collins		Krogan core		Krogan extd		Gavin		BioGRID	
General statistics										
Clusters	195		522		530		196		473	
Overlapping cluster pairs	97	0.5%	986	0.7%	924	0.6%	168	0.8%	434	0.3%
Proteins covered	1295		1876		1878		1095		2580	
Proteins in ≥ 2 clusters	222	17.1%	603	32.1%	661	35.2%	233	21.2%	721	27.9%
Overlap size distribution between cluster pairs										
1	29	29.9%	325	32.9%	279	30.1%	59	35.1%	250	57.6%
2	15	15.4%	108	10.9%	128	13.8%	31	18.4%	49	11.2%
3	18	18.5%	233	23.6%	206	22.2%	21	12.5%	40	9.2%
4	14	14.4%	103	10.4%	112	12.1%	12	7.1%	27	6.2%
5 or larger	21	21.6%	217	22.0%	199	21.5%	45	26.7%	68	15.6%

2.8 Biological relevance of the clusters generated by ClusterONE

To assess the biological relevance of the clusters generated by ClusterONE, we have first calculated some general statistics for the clusterings generated by ClusterONE and compared them with the properties of the MIPS and SGD gold standards (**Supplementary Table 10, 11**). As for the number of complexes, the results obtained from the Collins and Gavin datasets are closer to the actual number of MIPS complexes, while the results from the Krogan datasets and BioGRID seem to contain a large number of extra clusters. The fraction of overlapping cluster pairs is close to that observed in the SGD gold standard and slightly lower than that of the MIPS gold standard, which may be explained by the fact that some of the MIPS categories are not real protein complexes but groups of related complexes. The same applies to the fraction of proteins contained in at least two clusters. In almost all cases, the generated clusters cover more proteins than the corresponding gold standards.

Owing to the fact that gold standard protein complex sets are incomplete [3], a predicted complex that does not match any of the reference complexes may belong to a valid but previously uncharacterized complex as well. To this end, the comparison measures outlined in Section 1.1 should be complemented with scores that assess the biological relevance of predicted complexes based on the functional homogeneity or the co-localization of the constituent proteins instead of relying on a pre-defined gold standard. This is motivated by the fact that a protein complex can be formed only when its constituents are to be found in the same cellular compartment [33], and also that protein complexes tend to be responsible for a given biological function or molecular process.

The co-localization score [34] quantifies the extent to which proteins in a protein complex or a set of protein complexes belong to the same cellular compartment, given a pre-existing classification of individual proteins into localization categories. The co-localization score of a single complex is simply the maximum fraction of proteins in the complex which are

Nature Methods: doi:10.1038/nmeth.1938

Supplementary Table 11: General properties of the gold standard protein complexes.

	MIPS		SGD	
General statistics				
Complexes	203		323	
Overlapping complex pairs	401	2.0%	296	0.6%
Proteins covered	1189		1279	
Proteins in ≥ 2 complexes	820	69.0%	332	26.0%
Overlap size distribution between complex pairs				
1	89	22.2%	159	53.7%
2	33	8.2%	46	15.5%
3	63	15.7%	39	13.2%
4	37	9.2%	20	6.8%
5 or larger	179	44.6%	32	10.8%

found at the same localization. The co-localization score of a set of complexes is the mean co-localization score of all complexes in the set, weighted by the sizes of the complexes. In our benchmarks, we have used the localization classification of Huh et al [35].
The co-localization measure is suitable for flat (i.e. non-hierarchical) localization annotations, but it does not cater for homogeneity in biological processes or molecular functions, which are usually described in terms of a hierarchical classification scheme like the Gene Ontology. For the Gene Ontology annotations, we conducted an overrepresentation analysis of the annotations on each predicted complex as follows, following the method of Zhang et al [36].
Let M denote the total number of proteins in the original PPI network, and for a given predicted complex and an annotation term X, let m and K denote the number of proteins in the predicted complex and in the category, respectively. Furthermore, let us assume that k out of the m proteins in the predicted complex are annotated by X. The probability of observing k or more proteins annotated by X in a set of size m by pure chance is then given by:

$$P = \sum_{i=k}^{m} \frac{\binom{M-K}{m-i}\binom{K}{i}}{\binom{M}{m}}$$

Category X is then said to be enriched in the predicted complex at significance level p if $P < p$. The number of predicted complexes with at least one enriched annotation at significance level 0.05 divided by the total number of predicted complexes yields the overrepresentation score of the predicted complex set. Note that multiple hypothesis testing is performed to determine whether a predicted complex contains an enriched category or not, therefore the significance levels of individual tests have to be adjusted according to the Benjamini-Hochberg method [37] in order to control the false discovery rate (FDR) and keep the overall significance level of the test at 0.05. Gene Ontology annotations with IEA, ND and NAS evidence codes ("Inferred from electronic annotation", "No biological data available" and "Non-traceable author statement", respectively) were ignored. Since

one may argue that using GO annotations with evidence code IPI ("Inferred from physical interaction") presents a case of circular reasoning as the algorithm also uses interaction data to infer the complexes, we have also repeated the overrepresentation analysis while ignoring annotations supported only by an IPI evidence code as well.
The co-localization and overrepresentation analysis results (**Supplementary Tables 1–2**) also confirm that the clusters generated by ClusterONE are biologically relevant. The tables contain the scores of MCL-derived complexes as well for comparison. ClusterONE consistently produces higher overrepresentation scores across all aspects of the GO tree than MCL both with and without the IPI annotations, and it also achieves a higher co-localization score for all except the Collins dataset. We notice that the overrepresentation scores were lower for both algorithms across all three GO categories when IPI evidence codes were ignored; part of this effect may be attributed to the fact that less data was available for the overrepresentation analysis.

3 Discussion on the importance of weights in protein complex detection

A reasonable argument against using the weights in protein-protein interaction networks is their unreliability: the weights themselves are usually calculated using complicated machine learning approaches that operate on the original noisy experimental datasets. This is one of the reasons why most analysis aimed at detecting complexes from PPI networks have used unweighted datasets so far. However, here we argue that, altough the value of a single weight can be highly unreliable, network weights, taken globally, can improve the detection of protein complexes. **Supplementary Figure 1** shows a comparison of the performance of those algorithms that can handle weights (affinity propagation, MCL, RRW and ClusterONE) on the weighted datasets (Collins, Krogan core, Krogan extended, Gavin) and on their unweighted variants (obtained by thresholding the weighted datasets using thresholds indicated in the respective dataset publications). We re-tuned the parameters of affinity propagation, MCL and RRW for these unweighted datasets. We can see that, with the exception of RRW on the Krogan core dataset, the use of weights improves the performance of all the algorithms.
This leads us to another question on whether the performance of ClusterONE is better only thanks to its ability to take weights into account, or it is also due to a fundamentally different underlying algorithm. To address this question, **Supplementary Figure 2** shows the performance of all the eight algorithms on all the unweighted datasets (again, the unweighted variants of the weighted datasets were obtained by thresholding the weighted datasets using thresholds indicated in the respective dataset publications). We can see that ClusterONE outperforms all its competitors. Only on the Gavin dataset, CMC and RRW exhibit a performance close to the one of ClusterONE.

Nature Methods: doi:10.1038/nmeth.1938

4 A brief summary of how PPI weights were generated from experiments

4.1 Deriving the weights for the Gavin et al dataset

In the paper of Gavin et al [5], the weights of the interactions were defined by using the so-called *socio-affinity index* introduced in [5] that is based on the log-odds of the number of times two proteins were observed together in a purification, relative to the expected frequency of such a co-occurrence based on the number of times the proteins appeared in purifications. In other words, pairs of proteins with high socio-affinity indices were seen together in a purification more frequently than what one would have expected by random chance.

Formally, let $n_{i,j|i=\text{bait}}$ be the number of times that protein i as a tagged bait retrieved protein j as a prey in a purification, let f_i^{bait} be the fraction of purifications in which protein i was bait, f_j^{prey} be the fraction of all retrieved preys that were protein j, n_{bait} the total number of purifications (i.e. baits) and $n_{i=\text{bait}}^{\text{prey}}$ the number of preys retrieved with protein i as a bait. Furthermore, let $n_{i,j}^{\text{prey}}$ be the number of times that proteins i and j were seen together in a purification with baits different from i and j, and n_{prey} be the number of preys observed with a particular bait, excluding the bait itself. The socio-affinity index is then defined as a sum of three log-odds: $S_{i,j|i=\text{bait}}$ (the log-odds of protein i as bait retrieving protein j as prey, assuming the spoke model), $S_{i,j|j=\text{bait}}$ (the log-odds of protein j as bait retrieving protein i as prey, assuming the spoke model) and $M_{i,j}$ (the log-odds of protein i and j appearing together in a purification where neither i nor j was bait, assuming the matrix model). These are defined as follows:

$$
\begin{aligned}
S_{i,j|i=\text{bait}} &= \log\left(\frac{n_{i,j|i=\text{bait}}}{f_i^{\text{bait}} n_{\text{bait}} f_j^{\text{bait}} n_{i=\text{bait}}^{\text{prey}}}\right) \\
M_{i,j} &= \log\left(\frac{n_{i,j}^{\text{prey}}}{f_i^{\text{prey}} f_j^{\text{prey}} \sum_{\text{all baits}} n_{\text{prey}}(n_{\text{prey}}-1)/2}\right)
\end{aligned}
$$

The socio-affinity index $A_{i,j}$ of proteins i and j is then simply

$$A_{i,j} = S_{i,j|i=\text{bait}} + S_{i,j|j=\text{bait}} + M_{i,j}$$

Gavin et al [5] argued that "[g]enerally, pairs with socio-affinity indices below 5 should be considered with caution". To this end, we have considered only those protein-protein interactions in the Gavin dataset that had a socio-affinity index above 5, and then divided all the socio-affinity indices with the maximum socio-affinity index encountered in the dataset to constrain the weights between zero and one.

4.2 Deriving the weights for the Krogan et al datasets

Krogan et al [6] have used MALDI-TOF mass spectrometry and LC-MS/MS to identify protein-protein interactions, based on the observation that either mass spectrometry

method often fails to identify a protein, and the usage of two independent methods can increase the coverage and confidence of the obtained interactome. The results of the two methods were combined by supervised machine learning methods with two rounds of learning, using hand-curated protein complexes in the MIPS reference database [9] as a gold standard dataset.
In the first round of learning, Krogan et al have tested Bayesian inference networks and 28 different kinds of decision trees, eventually settling on three methods: Bayesian networks and C4.5-based and boosted decision stumps. The output of these three methods were then used as the input for a second round of learning where the stacked generalization algorithm [38] was used with a logistic regression learner. The output of the stacked generalization algorithm (i.e. a probability value between 0 and 1) was then thresholded at two different levels to obtain the *core* and *extended* datasets. The Krogan core dataset included all interactions with posterior probability higher than 0.273, while the extended dataset included all interactions with posterior probability higher than 0.101. The posterior probability scores were attached as weights to the interactions when the datasets were analyzed by the clustering methods in this manuscript. The authors have also noted that the interaction probabilities are likely to be underestimated since the complexes from MIPS gold standard do not include interactions between proteins in different complexes.

4.3 Deriving the weights for the Collins et al dataset

Collins et al [7] have combined the experimentally derived PPI networks of Krogan et al [6] and Gavin et al [5] by re-analyzing the raw primary affinity purification data of these experiments using a novel scoring technique called *purification enrichment* (PE). The PE scores were motivated by the probabilistic socio-affinity scoring framework of Gavin et al [5] but also take into account negative evidence (i.e. pairs of proteins where one of them fails to appear as a prey when the other one is used as a bait).
Similarly to the socio-affinity index, the PE score also consists of three components: S_{ij} (the sum of direct bait-prey evidence in favor of an interaction between protein i as bait and protein j as prey), S_{ji} (the sum of direct bait-prey evidence when protein j is bait and protein i is prey) and M_{ij} (the sum of indirect prey-prey evidence when proteins i and j have a common bait in some purification). The S_{ij} and S_{ji} components account for evidence under the spoke model of interactions, while M_{ij} accounts for the matrix model. Both S_{ij} and M_{ij} are defined as the sum of log-evidence terms as follows:

$$S_{ij} = \sum_k s_{ijk} \qquad M_{ij} = \sum_l m_{ijl}$$

where k iterates over the purifications where protein i was used as a bait, and l iterates over the purifications where proteins i and j were simultaneously observed as preys.
The formulation of both s_{ijk} and m_{ijl} follows the general scheme of log-evidence, i.e. $\text{Evidence(observation)} = \log_{10} \frac{P(\text{observation}|\text{true PPI})}{P(\text{observation}|\text{false PPI})}$. For the s_{ijk} terms, the exact form is

Nature Methods: doi:10.1038/nmeth.1938

as follows:

$$s_{ijk} = \begin{cases} \log_{10} \frac{r+(1-r)p_{ijk}}{p_{ijk}} & \text{if protein } j \text{ is a prey in purification } k \text{ using bait } i \\ \log_{10}(1-r) & \text{otherwise} \end{cases}$$

where r is the probability that a true association will be preserved and detected in a purification experiment and p_{ijk} is the probability that a bait-prey pair will be observed for nonspecific reasons. r was estimated using the observed frequency of successful purifications over a set of very confident interactions (the intersection of MIPS small scale experiments and MIPS complexes) and turned out to be 0.51 for the Krogan and 0.62 for the Gavin dataset. p_{ijk} was also estimated from the MIPS complexes using a Poissonian model and the observed number of preys in purification k with bait i, the number of times protein i was used as a bait, and the estimate of the non-specific frequency of occurrence of prey j in the dataset.
For the m_{ijk} terms, the general form of the formula is similar:

$$m_{ijk} = \log_{10} \frac{r + (1-r)p'_{ijk}}{p'_{ijk}}$$

where p'_{ijk} is an estimate of the probability that proteins i and j occur nonspecifically as preys in the same purification at least once in the dataset. p'_{ijk} was again estimated using a Poissonian model.
PE scores were then calculated individually for the Krogan and Gavin datasets using the raw primary purification results. The final PE score was combined from the individual PE scores as follows:

$$\mathrm{PE}_{ij}^{\mathrm{combined}} = 0.5 \times \mathrm{PE}_{ij}^{\mathrm{Krogan}} + \mathrm{PE}_{ij}^{\mathrm{Gavin}}$$

where the 0.5 multiplier for the Krogan PE scores accounts for the redundancy in the Krogan et al dataset due to the two independent mass spectrometry methods used (LC-MS/MS and MALDI-TOF).
Finally, the PE scores were smoothed and mapped to the range 0-1 using the combination of monotonic LOESS regression [39] and the pool adjacent violators algorithm [40]. Scores below 0.05 were set to zero for computational efficiency, and a subset of reliable interactions was established by taking 9074 interactions with the highest PE scores. The exact score threshold was selected based on the true positive to true negative rate evaluated on the MIPS small scale experiments.

References

[1] Bader, G. D. & Hogue, C. W. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics* **4**, 2 (2003).

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

[3] Jansen, R. & Gerstein, M. Analyzing protein function on a genomic scale: the importance of gold-standard positives and negatives for network prediction. *Curr. Opin. Microbiol.* **7**, 535–545 (2004).

[4] Boulesteix, A.-L. Over-optimism in bioinformatics research. *Bioinformatics* **26**, 437–439 (2009).

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

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

[7] Collins, S. R. *et al.* Toward a comprehensive atlas of the physical interactome of Saccharomyces cerevisiae. *Mol. Cell. Proteomics* **6**, 439–450 (2007).

[8] Stark, C. *et al.* BioGRID: a general repository for interaction datasets. *Nucl Acids Res* **34**, D535–539 (2006).

[9] Mewes, H. W. *et al.* MIPS: analysis and annotation of proteins from whole genomes. *Nucl. Acids Res.* **32**, D41–44 (2004).

[10] Hong, E. *et al.* Gene Ontology annotations at SGD: new data sources and annotation methods. *Nucl Acids Res* **36**, D577–581 (2008).

[11] Frey, B. J. & Dueck, D. Clustering by passing messages between data points. *Science* **315**, 972–976 (2007).

[12] Palla, G., Derényi, I., Farkas, I. & Vicsek, T. Uncovering the overlapping community structure of complex networks in nature and society. *Nature* **435**, 814–818 (2005).

[13] Adamcsek, B., Palla, G., Farkas, I., Derényi, I. & Vicsek, T. CFinder: locating cliques and overlapping modules in biological networks. *Bioinformatics* **22**, 1021–1023 (2006).

[14] Liu, G., Wong, L. & Chua, H. N. Complex discovery from weighted PPI networks. *Bioinformatics* **25**, 1891–1897 (2009).

[15] Enright, A. J., Dongen, S. V. & Ouzounis, C. A. An efficient algorithm for large-scale detection of protein families. *Nucl. Acids Res.* **30**, 1575–1584 (2002).

[16] van Dongen, S. Graph clustering via a discrete uncoupling process. *SIAM J. Matrix Anal. Appl.* **30**, 121–141 (2008).

[17] King, A., Pržulj, N. & Jurisica, I. Protein complex prediction via cost-based clustering. *Bioinformatics* **20**, 3013–3020 (2004).

[18] Macropol, K., Can, T. & Singh, A. RRW: repeated random walks on genome-scale protein networks for local cluster discovery. *BMC Bioinformatics* **10**, 283 (2009).

Nature Methods: doi:10.1038/nmeth.1938

[19] Apeltsin, L., Morris, J., Babbitt, P. & Ferrin, T. Improving the quality of protein similarity network clustering algorithms using the network edge weight distribution. *Bioinformatics* **27**, 326–333 (2011).

[20] Baumes, J., Goldberg, M. & Magdon-Ismail, M. Efficient identification of overlapping communities. In *Proceedings of the IEEE International Conference on Intelligence and Security Informatics (ISI 2005)*, vol. 3495 of *Lecture Notes in Computer Science*, 27–36 (Springer Berlin / Heidelberg, 2005).

[21] Chen, J., Zaïane, O. R. & Goebel, R. Detecting communities in large networks by iterative local expansion. In *Proceedings of the 2009 International Conference on Computational Aspects of Social Networks*, 105–112 (IEEE Computer Society, 2009).

[22] Lancichinetti, A., Fortunato, S. & Kertész, J. Detecting the overlapping and hierarchical community structure in complex networks. *New J. Phys.* **11**, 033015 (2009).

[23] Wang, J., Li, M., Chen, J. & Pan, Y. A fast hierarchical clustering algorithm for functional modules discovery in protein interaction networks. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **8**, 607–620 (2011).

[24] Fortney, K., Kotlyar, M. & Jurisica, I. Inferring the functions of longevity genes with modular subnetwork biomarkers of Caenorhabditis elegans aging. *Genome Biol.* **11**, R13 (2010).

[25] Ito, T. *et al.* A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc. Natl. Acad. Sci. USA* **98**, 4569–4574 (2001).

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

[27] Gavin, A. *et al.* Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* **415**, 141–147 (2002).

[28] Jensen, L. & Bork, P. Not comparable, but complementary. *Science* **322**, 56–57 (2008).

[29] Yu, H. *et al.* High-quality binary protein interaction map of the yeast interactome network. *Science* **322**, 104–110 (2008).

[30] Farkas, I., Ábel, D., Palla, G. & Vicsek, T. Weighted network modules. *New J. Phys.* **9**, 180 (2007).

[31] Durstenfeld, R. Algorithm 235: Random permutation. *Communications of the ACM* **7**, 420 (1964).

[32] Cheeseman, I. *et al.* Implication of a novel multiprotein Dam1p complex in outer kinetochore function. *J. Cell Biol.* **155**, 1137–1145 (2001).

[33] Jansen, R. *et al.* A Bayesian networks approach for prediction protein-protein interactions from genomic data. *Science* **302**, 449–453 (2003).

[34] Friedel, C. C., Krumsiek, J. & Zimmer, R. Bootstrapping the interactome: unsupervised identification of protein complexes in yeast. *J. Comput. Biol.* **16**, 971–987 (2009).

[35] Huh, W.-K. K. *et al.* Global analysis of protein localization in budding yeast. *Nature* **425**, 686–691 (2003).

[36] Zhang, B., Park, B.-H., Karpinets, T. & Samatova, N. F. From pull-down data to protein interaction networks and complexes with biological relevance. *Bioinformatics* **24**, 979–986 (2008).

[37] Benjamini, Y. & Hochberg, Y. Controlling the false discovery date: a practical and powerful approach to multiple testing. *J. Roy. Statist. Soc. Ser. B* **57**, 289–300 (1995).

[38] Wolpert, D. Stacked generalization. *Neural Networks* **5**, 241–259 (1992).

[39] Hastie, T., Tibshirani, R. & Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 165–173, 277–278 (Springer, 2001).

[40] Robertson, T., Wright, F. & Dykstra, R. *Order Restricted Statistical Inference* (John Wiley and Sons, 1988).

Nature Methods: doi:10.1038/nmeth.1938