BIOINFORMATICS ORIGINAL PAPER

Vol. 21 no. 3 2005, pages 364–378
doi:10.1093/bioinformatics/bti021

Iterative Cluster Analysis of Protein Interaction Data

Vicente Arnau[1], Sergio Mars[2] and Ignacio Marín[2,]*

[1]Departamento de Informática and [2]Departamento de Genética, Universidad de Valencia, Burjassot 46100, Valencia, Spain

Received on March 12, 2004; revised on September 7, 2004; accepted on September 7, 2004
Advance Access publication September 16, 2004

ABSTRACT

Motivation: Generation of fast tools of hierarchical clustering to be applied when distances among elements of a set are constrained, causing frequent distance ties, as happens in protein interaction data.

Results: We present in this work the program UVCLUSTER, that iteratively explores distance datasets using hierarchical clustering. Once the user selects a group of proteins, UVCLUSTER converts the set of primary distances among them (i.e. the minimum number of steps, or interactions, required to connect two proteins) into secondary distances that measure the strength of the connection between each pair of proteins when the interactions for all the proteins in the group are considered. We show that this novel strategy has advantages over conventional clustering methods to explore protein–protein interaction data. UVCLUSTER easily incorporates the information of the largest available interaction datasets to generate comprehensive primary distance tables. The versatility, simplicity of use and high speed of UVCLUSTER on standard personal computers suggest that it can be a benchmark analytical tool for interactome data analysis.

Availability: The program is available upon request from the authors, free for academic users. Additional information available at http://www.uv.es/~genomica/UVCLUSTER

Contact: ignacio.marin@uv.es

1 INTRODUCTION

The extraction of relevant information from massive amounts of biological data is becoming crucial in the post-genomic era. Current efforts are focused on the generation of tools able to allow the classification of large amounts of similar, correlated or interconnected elements and retrieve from that classification useful patterns or regularities that can be later explored either in the laboratory or *in silico*. Thus, in the last years we have witnessed an ever-increasing interest in the development of classification tools for gene expression data obtained from microarray analysis (Quackenbush, 2001; Gibbons and Roth, 2002). The recent generation of massive protein–protein interaction data has created a similar need for methods to efficiently explore the complex graphs of interconnected proteins that often faithfully represent complex metabolic functions of a cell (Schwikowski *et al*., 2000; Drees *et al*., 2001; reviewed in Salwinski and Eisenberg, 2003; Bader *et al*., 2003).

Among the best-known and most powerful methods of classification are those involving hierarchical clustering (reviewed in Everitt *et al*., 2001). Elements are progressively classified into sets either by sequentially putting them together in non-overlapping classes (agglomerative methods) or by progressively dividing the full set of elements into smaller groups (divisive methods). Many different types of hierarchical clustering methods exist that depend on how the distances among elements are evaluated to build (or split) the groups, and there is considerable interest in determining when some of those methods perform better than others (e.g. Gibbons and Roth, 2002). Hierarchical clustering is one of the most common methods of classification used in biology. Among many others, common uses of hierarchical clustering techniques are the classification of organisms of different populations or species according to quantitative similarities (numerical taxonomy), the ordering, according to sequence similarity, of sets of genes or proteins and, more recently, the determination of sets of genes with similar profiles of expression according to microarray-derived data (reviewed in Nei and Kumar, 2000; Quackenbush, 2001; Felsenstein, 2004). In this study, we explore the use of hierarchical clustering for the functional classification of proteins using protein–protein interaction data. Our approach has two steps. First, we measure the distance among any two proteins in a protein–protein interaction network according to the minimum number of steps required to connect them, where each step is a known, physical protein–protein interaction. Second, we use those distances to classify the proteins in groups using hierarchical clustering. This approach is thus in principle very similar to those used for other types of data and would seem quite easy to implement. However, protein–protein interaction data have special features that make the use of hierarchical clustering methods particularly problematic.

*To whom correspondence should be addressed.

An often overlooked aspect of hierarchical cluster analysis is the fact that it has serious intrinsic problems when used with datasets in which the distances among many elements are identical (the 'ties in proximity' problem; Backeljau *et al.*, 1996; Takezaki, 1998; MacCuish *et al.*, 2001). Ties generate multiple mathematically equivalent solutions when hierarchical clustering is performed. In favorable cases, when ties are very rare, we would anyway expect all alternative clustering solutions to be very similar. The simplest solution in those cases is to solve the ties arbitrarily, for example, in ways that are dependent on the order of data input. However, when ties are frequent, the number of alternative solutions and the differences among those alternatives increase. Backeljau *et al.* (1996) summarized the effect of ties on the performance of several commonly used computer programs, concluding that none of them was able to correctly confront this problem. The relative importance of ties was further analyzed by Takezaki (1998), which determined the likelihood of ties and their effect on bootstrap tests for some small simulated datasets, concluding that ties have only occasionally a significant influence. However, these two works analyzed small distance matrices based on nucleotide sequences, in which ties are relatively rare and often caused by the effect of rounding decimal numbers. Ties are much more frequent in other types of data, most especially when distances are constrained to a narrow range of values. Most especially, protein–protein interaction data generate one of the datasets in which ties are most prominent.

Currently, the most complete interactome data available in eukaryotes involves the protein interaction network of the yeast *Saccharomyces cerevisiae*. In addition to results derived from a large number of small-scale, directed studies, massive non-directed protein–protein interaction data for *S.cerevisiae* were generated either by using two-hybrid assays (Uetz *et al.*, 2000; Ito *et al.*, 2001) or by pulling down protein complexes using a tagged subunit as bait (Gavin *et al.*, 2002; Ho *et al.*, 2002). Analysis of the large datasets generated using these approaches demonstrated that the *S.cerevisiae* interactome has 'small world' properties (Wagner, 2001; see review by Albert and Barabási, 2002). We can define the distance between two proteins as the minimum number of direct interactions among proteins in the dataset that are required in order to connect them (a parameter often called 'minimal path length' or simply 'path length'; see Barabási and Oltvai, 2004). For a 'small world' interaction dataset, it is found that many distances are identical because most proteins are either directly connected (i.e. they are part of the same protein complex or interact in two-hybrid assays; distance $= 1$) or separated by just a few steps, each one involving a physical interaction between two proteins or protein complexes. That interactome data have small world properties was already anticipated using a small fraction of the information currently available (just 899 interactions) by Wagner (2001), which described in the *S.cerevisiae* protein network a group of 466 proteins connected by an average distance (average path length) of 7.14. This distance should become smaller when more interactions are determined. Thus, recently Wilhelm *et al.* (2003) described a *S.cerevisiae* dataset involving about 20 000 interactions and with an average distance for the proteins of the largest connected component equal to only 2.57. Similarly, in a recent analysis in which 15 210 interactions were considered [data obtained from the January 2004 release of the DIP database; Xenarios *et al.* (2002)], we found that all proteins that can be connected have distances that range from 1 to 12. When we assigned a conventional value of distance equal to 24 (i.e. twice the largest detected for any connected proteins) to all those pairs of proteins that cannot be connected by any path, we found an average distance of 4.97 for 4721 *S.cerevisiae* proteins. This value is of course an overestimate, because in the future more/all proteins are expected to be found connected, and with shorter distances. In any case, for DIP data, we found that there are $4721 \times 4720/2 = 11.1$ millions of distances among proteins that are constrained to a short range of possible distance values, mostly between 1 and 12, so it can be deduced that the *S.cerevisiae* interactome dataset includes millions of identical distances. Very recent data for animal species, such as *Drosophila melanogaster* or *Caenorhabditis elegans* are more limited (i.e. the average number of interactions per protein is lower than in *S.cerevisiae*). However, the general pattern of small world properties with constrained distance values is similarly observed (Giot *et al.*, 2003; Li *et al.*, 2004).

The ties in proximity problem has led most authors interested in analyzing protein interaction data to focus their efforts on generating tools, which are based on graph theory and designed to detect clusters of highly or similarly connected proteins (e.g. Bader and Hogue, 2003; Bu *et al.*, 2003; Goldberg and Roth, 2003; Spirin and Mirny, 2003; Gagneur *et al.*, 2004; Pereira-Leal *et al.*, 2004; Przulj *et al.*, 2004). These approaches are very interesting because they may efficiently detect functionally relevant modules. However, they have the obvious limitation of not allowing a general view of the relationships among proteins selected with criteria different from their degree of connection. In this study, we describe a novel, alternative strategy of analysis that allows to explore the characteristic type of complex data that suffers the ties in proximity problem using hierarchical clustering. In a significant precedent to our work, Rives and Galitski (2003) circumvented the problem by using not the distances among elements but the correlation coefficient obtained when the distances of two proteins are measured against all the members in a database (see also Prinz *et al.*, 2004 for a related procedure). Our method is different, being based on obtaining multiple, equally valid solutions and evaluating the distances among elements according to those multiple solutions. This strategy is implemented in a program that we have called UVCLUSTER, which is extremely simple to use and fast enough to analyze large datasets involving hundreds of proteins on standard PC computers. UVCLUSTER allows the determination of the relative degree of proximity of groups of proteins that can be defined by the user in several ways, generating outputs that can be easily explored or converted into dendrograms. We demonstrate that UVCLUSTER-based analyses have obvious advantages over conventional hierarchical clustering and that UVCLUSTER can be used to explore interactome data in order to detect new proteins relevant to any chosen biological process.

2 SYSTEM AND METHODS

The UVCLUSTER program (available upon request from the authors, free for academic users) is written in C and its versions have been compiled and tested on Windows and Linux operating systems. The flow chart of the program is shown in Figure 1. UVCLUSTER analyses begin by importing a text file containing a dataset of direct protein–protein interactions. This dataset can be then filtered in two different ways. First, the user can create a list of proteins and then select one of two options: (1) use only those interactions between pairs of proteins in the list; or (2) exclude all interactions involving any protein in the list. After this first filter, a table with the selected group of interactions can be saved for further analyses. Then the user can apply a second filter by selecting a cutoff value for the maximum/minimum number of interactions that a protein may have to be included in the analysis. This second filter allows to exclude proteins with a number of interactions lower than a particular threshold (i.e. poorly connected proteins) or higher than the cutoff value (i.e. it eliminates 'hubs', highly connected proteins). After these two filters, the UVCLUSTER program generates with the remaining proteins a matrix of *primary distances* (d; equivalent to their shortest path or minimal path length) among them. If two proteins cannot be connected directly or indirectly, we have followed the convention of assigning a distance that is twice the longest distance among connected proteins in the dataset (modifying this convention does not alter any of the results that we will present below). The tables of primary distances generated by UVCLUSTER can be saved for further analyses. Thus, in other rounds of analysis, the user may decide whether to import a new file to generate another primary distance table or to directly use a table that is already available. This is useful, because a primary distance table obtained from all the interaction data that exist for a species is generated that can be used many times in different analyses.

Once a primary distance table has been generated or simply loaded, UVCLUSTER allows the user to further select groups of proteins to be analyzed, again in two different ways. A first option is to choose a single protein ('network center') while establishing a cutoff distance value. In this case, the program explores the primary distance table, by selecting all the proteins with primary distances from the selected one that are equal or lower than the cutoff value and generates a subtable containing only those proteins together with their distances.

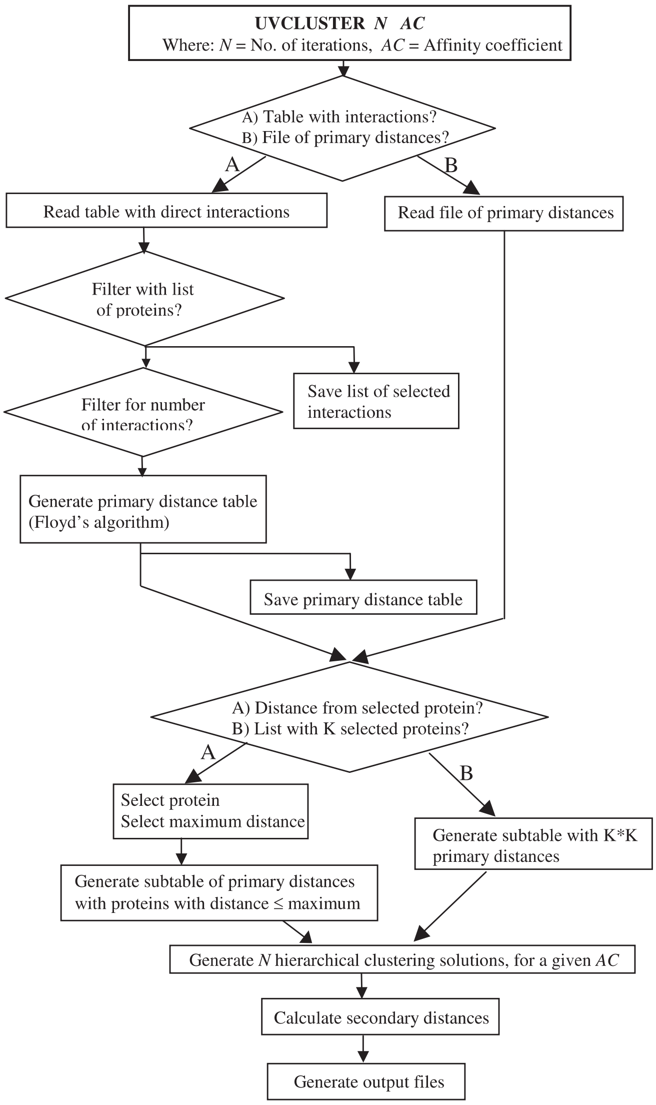
UVCLUSTER N AC
Where: N = No. of iterations, AC = Affinity coefficient
A) Table with interactions?
B) File of primary distances?
A
B
Read table with direct interactions
Read file of primary distances
Filter with list of proteins?
Save list of selected interactions
Filter for number of interactions?
Generate primary distance table (Floyd's algorithm)
Save primary distance table
A) Distance from selected protein?
B) List with K selected proteins?
A
B
Select protein
Select maximum distance
Generate subtable with K*K primary distances
Generate subtable of primary distances with proteins with distance ≤ maximum
Generate N hierarchical clustering solutions, for a given AC
Calculate secondary distances
Generate output files

Fig. 1. UVCLUSTER flow chart.

The second alternative involves providing the program with a list of proteins. Then, the program generates a subtable of primary distances, but only including the proteins in the list for which interaction data are available.

Once the subtable of primary distances has been generated by any of these two methods, UVCLUSTER iteratively uses agglomerative hierarchical clustering on the primary distances dataset to generate N alternative, equally valid, clustering solutions. The value of N must be chosen by the user before starting the analysis after considering the speed of the computer, number of proteins to analyze and the precision of analysis required (see below for a discussion). The solutions are generated in three steps: (1) random sampling of the elements of the dataset; (2) elements in the dataset are clustered according to group average linkage (Everitt *et al*., 2001); and (3) the agglomerative process is finished at a certain point, defined by the user that must select the value for a global 'stopping rule' parameter, the *Affinity Coefficient* (AC). AC is defined as follows:

$$AC = 100[(P_m - C_m)/(P_m - 1)], \quad (1)$$

where C_m (cluster mean) is the average of the distances for all elements included in the clusters and P_m (partition mean) is the average value of distances for the whole set of selected proteins. If $AC = 100$, then $C_m = 1$, meaning that only proteins with distance equal to 1 can be clustered together. Using $AC < 100$ relaxes the conditions, allowing that proteins separated by distances higher than 1 be put together in the same cluster. It therefore favors the generation of larger clusters, although including proteins often indirectly connected. We generally use values of AC ranging from 50 (highly relaxed) to 100 (maximally strict) in our analyses. If many proteins are directly connected, it may be useful to increase the AC value, while if distances among proteins are large, AC values may be decreased to facilitate the detection of potentially interesting clusters.

Once the dataset of N alternative solutions is obtained, UVCLUSTER evaluates in how many of them each pair of elements appear together in the same cluster and generates an output file containing a table with *secondary distances* (d') among the elements. The secondary distance between two elements is defined as the number of solutions in which those two elements do not appear together in the same cluster divided by the total number of solutions (N). Therefore, iterative resampling of the primary distance data allows to establish how likely it is for each pair of elements to be clustered together when many alternative, equally good, clustering solutions are generated. This is the key of the strategy implemented in UVCLUSTER. Secondary distances establish the strength of the connection between two elements, *relative to all the elements in the analyzed dataset* [an idea first developed in Arnau and Marín (2003), although using a less appropriate clustering method]. Ties in secondary distances will be very rare, due to the complex nature of the connections among elements. This means that the secondary distance dataset may then be analyzed by conventional (i.e. non-iterative) methods to obtain a rigorous classification of the elements.

To facilitate the exploration of the results, UVCLUSTER generates four output files. The first one contains the tables of primary and secondary distances among the chosen elements plus the values of several significant parameters used in the analyses, such as AC, C_m and P_m. This first file also contains a table of secondary distances suitable to be copied to a text file and directly imported into MEGA 2.1 (Kumar *et al*., 2001). In MEGA, the secondary distance data can be used to generate dendrograms, using conventional methods of clustering such as UPGMA or Neighbor-joining. In the second UVCLUSTER output file, the results of an agglomerative hierarchical clustering using UPGMA performed with the secondary distance data are detailed. This file may be useful for a preliminary exploration of the results, especially for those users that are not familiar with MEGA or other similar packages. Finally, the third UVCLUSTER output file contains a graphical representation of the data in PGM (Portable GreyMap) format. To generate the PGM file, proteins are ordered according to the results described in the second UVCLUSTER output file. To facilitate the analysis of the PGM figure, a fourth output file is generated that contains a list of names that corresponds to the order in which proteins are shown in that figure. The PGM representation (see Arnau and Marín, 2003) is a square formed by K^2 smaller color-coded squares, where K is the number of proteins analyzed. Shades of gray indicate the degree of interaction between each pair of proteins, with light gray corresponding to close proteins and black to distantly connected proteins. PGM format files can be read using freeware programs such as IrfanView 3.85 (www.irfanview.com).

3 ALGORITHM

The dataset of direct protein–protein interactions must be written as a text file, in such a way that each line of the file contains the names of two interacting proteins separated by a tab. This is the format most commonly used to express protein–protein interaction data in public databases. Therefore, massive amounts of data can be very quickly imported into UVCLUSTER. Primary distances are calculated from this file of direct interactions using Floyd's algorithm (Floyd, 1962).

The main algorithm of the program, that characterizes the iterative clustering method, can be described as follows:

```
Import table with $d$ values
Select $N$, AC values
Repeat_from $k = 1$
    Random ordering of elements;
    Agglomerative hierarchical clustering using group
        average linkage ($d$, AC);
    Increment counters according to the solution found;
    $k = k + 1$;
To $k = N$
Generate $d'$ values corrected by $N$
Export files containing: (a) tables of $d$, $d'$ values;
(b) UPGMA clustering with $d'$ values and
(c) graphical output in PGM format.
```

4 IMPLEMENTATION

4.1 Speed and performance

No matter how complex or extensive the raw interaction data are, they can be quickly converted into primary distance data by UVCLUSTER and then saved for later analyses. In addition, the speed of generation of primary distance tables allows us to easily update any comprehensive species-specific table, every time a new information is available. For example, the *S.cerevisiae* data available in the January 2004 release of the DIP database (4721 proteins, 15 210 interactions) can be converted into a primary distance table in about 14 min on a standard PC computer (Intel Pentium IV 2.8 GHz processor with 512 MB RAM memory).

Once a table with primary distances has been generated, secondary distance data are also obtained very quickly. Time depends on the selected AC value. Smaller AC values that prolong the clustering process, require longer time. Therefore, to establish the speed of UVCLUSTER analyses, we have performed tests with AC values ranging from 50 to 100. First, by using a set of 34 elements and 561 primary distances (see Section 4.3), and running on the same standard computer detailed above, UVCLUSTER obtained the values of secondary distances with a number of iterations, $N = 10\,000$ in less than 2 s. Time increased linearly with the number of iterations, so a similar analysis with $N = 100\,000$ required 9 (AC $=$ 100) to 13 (AC $=$ 50) s. Similarly, for a set of 150 randomly chosen elements (11 175 primary distance values), an analysis with $N = 10\,000$ took from 9 (AC $=$ 100) to 125 (AC $=$ 50) s. Finally, the largest dataset that we have used composed of 500 randomly chosen proteins (i.e. 124 750 distances), which was analyzed ($N = 10\,000$), in 23 to 160 min, again depending on the AC value used.

UVCLUSTER results can be directly explored, considering the files that the program automatically generates containing either a UPGMA cluster analysis of secondary distances or a graphical PGM representation. However, for advanced users, and especially for analyses including up to a few hundreds of proteins, we think that the simplest and most efficient way to explore the results involves the generation of dendrograms based on secondary distances. Ties in secondary distances are very rare, facilitating the generation of non-ambiguous trees. For this reason, UVCLUSTER also generates outputs compatible with MEGA 2.1 (Kumar *et al.*, 2001). This is one of the most used program packages in the molecular evolution and phylogenetics fields and includes several standard methods to build dendrograms. Moreover, it can be obtained free from the authors (www.megasoftware.net).

4.2 UVCLUSTER analysis of synthetic graphs

A simple model (Figure 2) demonstrates the advantages of the iterative strategy implemented in UVCLUSTER. Figure 2A shows a graph with 11 elements connected by a total of 16 interactions. Two clusters (units 1–4 and 8–11) are obvious. Figure 2B shows a UPGMA tree obtained using primary distances. Input order to obtain this tree followed the numeric value $(1, 2, \ldots, 11)$ assigned to the elements. The solution shown in Figure 2B clearly fails to detect the two clusters, closely connecting units 4 and 5. This type of error is caused by the ties. If ties are solved in such a way that, by chance, units 4 and 5 (or, alternatively, 7 and 8) become clustered, the solution generated will necessarily fail to detect the two natural clusters. Figure 2C shows a UPGMA tree obtained using the secondary distances generated by UVCLUSTER with $N = 10\,000$ and AC $=$ 100. In this case, distances among units 1–3 or 9–11 are equal to zero. It can be seen that Figure 2C closely corresponds to Figure 2A: the two clusters appear as discrete entities with units 4 and 8 being the ones most closely connected to the rest.

It is interesting to develop an index to summarize the validity of the clusters detected, in order to be able to compare different cluster results. Such an index would help in determining that the solution shown in Figure 2C is better than that shown in Figure 2B. Many rules have been proposed to evaluate the different partitions of a dataset into clusters [see review by Gordon, 1999, pp. 60–65 and 185–204). However, we think that none of these methods is fully convenient when applied to proteomic interaction data. They all use the whole distance dataset for cluster validation (i.e. they take into account all distances, irrespectively of them implying direct interactions or simply shortest paths among distant elements). Evaluation of the whole distance dataset is essential for clustering, but validation of the clusters should take into account two features of this particular type of information, which demonstrate that a distance equal to one is qualitatively different from the other distances. First, distances equal to one imply a strong functional link: two proteins directly interact in the cell, and therefore must be at the same time in the same place and often functioning together. On the contrary, when larger than one, two identical distances may be radically different from a cellular point of view. For example, a distance of two may mean that two proteins are part of a complex together with a third protein that physically interacts with both of them, or, alternatively, may mean that in different moments of the cell cycle (or in different tissues, if we refer to a multicellular organism), there are transient interactions of both proteins with a third one. In the first case, all three proteins may function as a single unit, while in the second, the cellular roles of the two proteins may be radically different. A second consideration is that the protein interaction datasets are incomplete. As we already commented above for *S.cerevisiae*, each new experiment diminishes the distance among proteins. In fact, comparisons among the experiments performed so far in *S.cerevisiae* found a low degree of congruence (Bader and Hogue, 2002; von Mering *et al.*, 2002), suggesting that we are still quite far away from having a complete dataset of the protein interactions for any species. This means that it is reasonable to expect that, in the future, many more direct interactions will be found. In this sense, distances larger than one may be considered to be overestimates of the true distances.

All these considerations lead us to suggest that cluster results from proteomic interaction data may be conveniently evaluated considering just the number of intracluster and intercluster direct interactions. If we have a dataset in which K elements are connected by direct interactions, thus forming

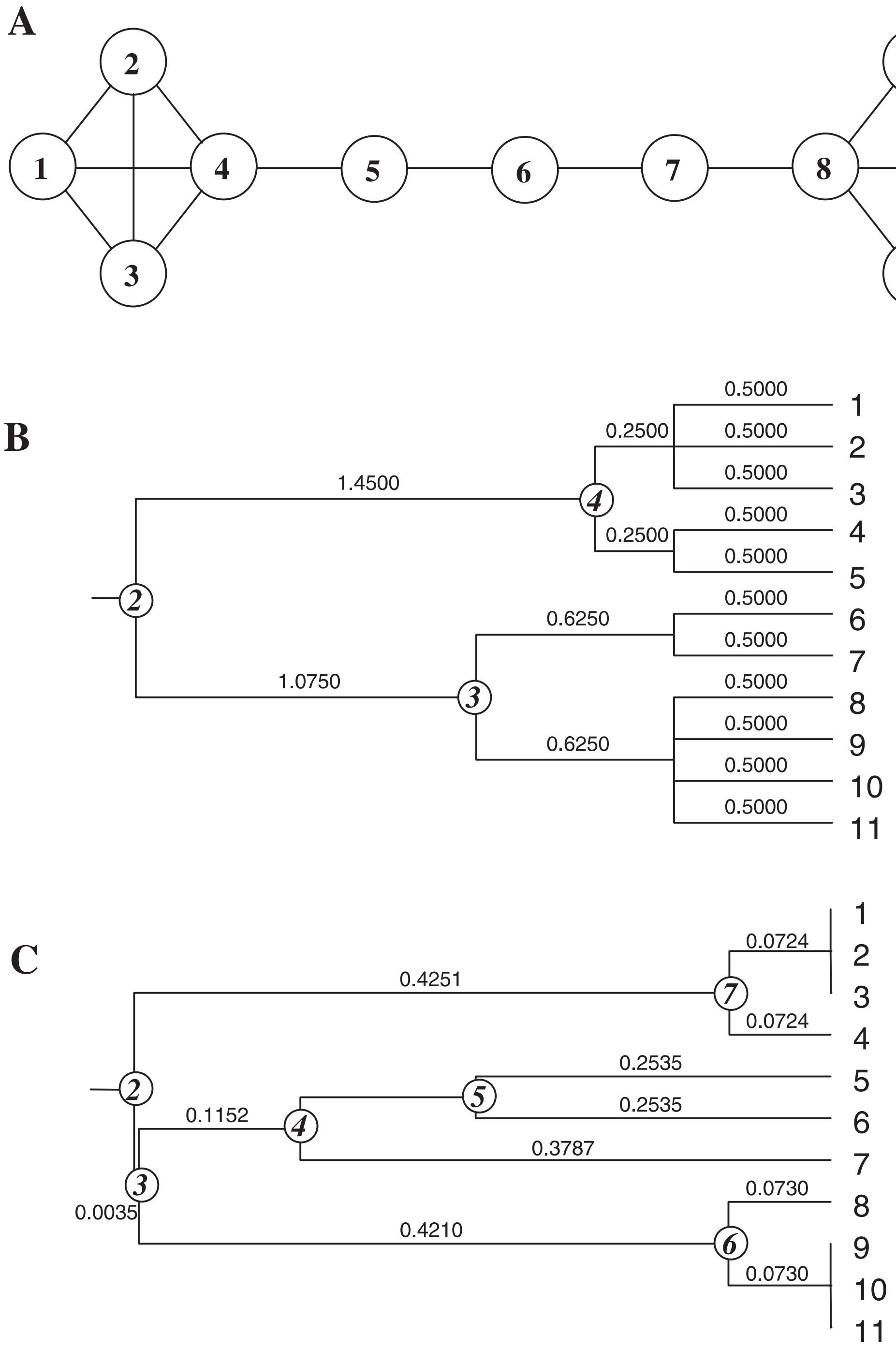

Fig. 2. A synthetic example. (**A**) Graph of 11 elements connected by 16 interactions. Two clusters (units 1–4 and 8–11) are observed. (**B**) UPGMA-based tree using primary distances derived from the graph in (A). Notice the lack of correspondence with the graph. (**C**) UPGMA-based tree using the secondary distances obtained using UVCLUSTER with AC = 100 and 10 000 iterations. The topology very closely corresponds to the graph shown in (A).

an undirected graph, we can define the following parameters. First, F will be the maximum possible number of direct interactions among elements [$F = K(K-1)/2$]. Let us also call n the number of actual direct interactions discovered among the K elements. For a particular partition into clusters, we can find a number M that corresponds to the maximum possible number of intracluster direct interactions:

$$M = \sum_{i-1}^{c} k_i(k_i-1)/2, \qquad (2)$$

where c is the number of clusters in the partition and k_i is the number of elements in each cluster. Finally, we will call p the total number of direct intracluster interactions actually known. We suggest to evaluate the partitions found in proteomic data by selecting as best the one that minimizes the cumulative hypergeometric distribution that follows:

$$\sum_{j=p}^{\mathrm{Min}\,(M,n)} \frac{\binom{M}{j}\binom{F-M}{n-j}}{\binom{F}{n}}. \tag{3}$$

This is equivalent to select the partition that generates a distribution that maximizes the proportion of intracluster direct interactions while minimizing the proportion of intercluster direct interactions.

We can apply this rule to the trees shown in Figure 2, by using their topology in order to establish partitions with a progressive number of clusters ($c \geq 2$) (see Levenstien *et al*., 2003 for a similar approach). Circles in the dichotomic nodes have been numbered in Figure 2B and C according to the number of clusters generated when we progress from left (largest distances) to right (smallest distances). We found that the tree in Figure 2C has a minimum value of Equation (3) in the particular case when $c = 3$. In that case, we find that out of 16 total direct interactions, 14 are found within clusters and only 2 (units 4 and 5; units 7 and 8) are found between clusters. According to the cumulative hypergeometric distribution described in Equation (3), the likelihood of finding by chance a distribution at least as asymmetric as the one defined by such partition is 3.17×10^{-9}. None of the partitions obtained from the tree in Figure 2B has such a small value (lowest value: 3.30×10^{-6}). We can conclude, as intuitively was evident, that UVCLUSTER analysis using secondary distances has provided a partition that is superior to those obtained by using the tree topology generated from the direct analysis of the primary distances.

4.3 UVCLUSTER exploration of the actin cytoskeleton of *Saccharomyces cerevisiae*

To demonstrate the advantages that our program provides when applied to real biological data, we first chose as a model a set of 34 *S.cerevisiae* proteins characterized by Drees *et al*. (2001), which they described as 26 proteins participating in actin patch assembly and patch-mediated endocytosis together with 8 proteins involved in other related processes, such as cytokinesis (BNI1, BNR1), endocytosis (SVL3), control of the morphogenesis checkpoint (SWE1, HSL7) or the CDC42 signaling pathway (CDC42, CLA4, GIC2) (Drees *et al*., 2001). Results shown in that study were updated by analyzing the DIP database, which we found contained all the information available in the literature. The graph of protein–protein interactions shown in Figure 3 summarizes all currently (February 2004) available information for that set of proteins.

This particular set of proteins was selected for two main reasons. First, it comprises a highly explored (i.e. most likely without false positive interactions) and very compact set of elements, in which obvious clusters or groups cannot be detected. This is shown in Figure 3, obtained using PIVOT (Orlev *et al*., 2004), a program that provides a minimal-energy-based layout that highlights clustering among units. However, whether clusters are present in Figure 3 is unclear. This group of proteins may thus be a good test to determine whether standard methods of clustering may detect any particular organization in the data, and how UVCLUSTER compares with these standard procedures. Second, because it contains a few elements that stand out as participating in processes that are related (through connections with actin function) but not part of the two main processes in which most elements participate (actin patch assembly and patch-mediated endocytosis), we can test whether we can discriminate these groups using UVCLUSTER.

Figure 4 shows the results obtained with two different methods. First, we show in Figure 4A the tree generated by MEGA 2.1 that was obtained with UPGMA and the primary distances calculated from the graph shown in Figure 3. This first option therefore corresponds to the analysis that can be typically performed using standard clustering tools. Second, we show the tree obtained when the UPGMA algorithm is applied to the set of secondary distances obtained using UVCLUSTER, with $N = 10\,000$ and AC $= 100$ (Figure 4B). Both trees are quite different. As was obtained in the synthetic example shown above, the first tree is just one of the many possible solutions that can be obtained by applying hierarchical clustering to the primary distance dataset. When we evaluated the topologies shown in Figure 4A and B using the hypergeometric-based parameters described above, we found a partition (with $c = 12$) in the tree obtained using UVCLUSTER that is much more extreme ($p = 1.79 \times 10^{-27}$) than the best value obtained from the direct analysis of the primary distances ($c = 10$; p value $= 1.80 \times 10^{-23}$). This optimal partition is shown also in Figure 4B. Because it is known that one important problem with protein–protein interaction data is the presence of false positives, we tested whether the partition shown in Figure 4B is robust by randomly generating false protein–protein interactions that were added to the DIP database. Even when 7605 false interactions were added (making the randomly generated interactions 33% of all present in the modified database), the clusters were identically recovered when $N = 10\,000$, AC $= 100$. Only when 15 210 false interactions were added (50% of all interactions are then spurious) we found a variation: clusters 2 and 7 appeared together. These results suggest that, when provided a significant core of true interactions, UVCLUSTER analyses may provide robust topologies even in the presence of a substantial number of false positive interactions.

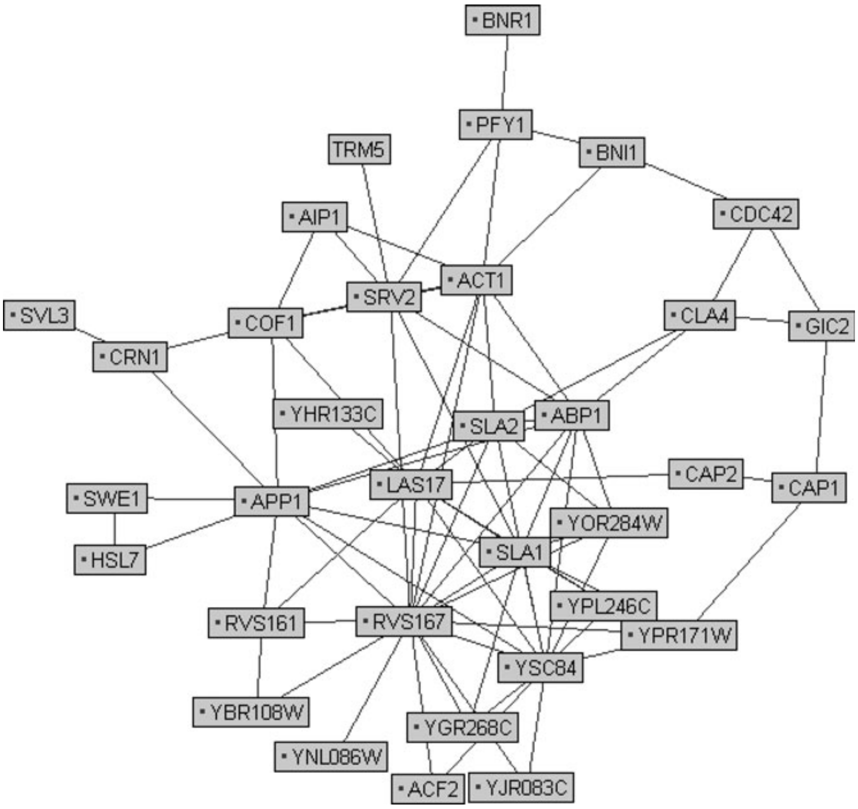

Fig. 3. Set of proteins involved in actin patch assembly and endocytosis according to Drees *et al*. (2001). Lines indicate direct interactions among them as found in the DIP database (January 2004 release). The figure was drawn using PIVOT (Orlev *et al*., 2004).

We also wanted to check whether the partition obtained was biologically relevant. First, we considered the information provided by Drees *et al*. (2001). The proteins involved (according to these authors) in actin patch assembly and function form clusters 1–5, 7, 11 and 12. Cluster 6 corresponds to the three proteins described as involved in CDC42 signaling. Cluster 8 includes the two proteins described by Drees *et al*. (2001) as involved in the morphogenesis checkpoint together with a third protein (APP1, called YNL094w in the study by Drees *et al.*) whose relationship with that process has not been determined. Cluster 9 contains the endocytosis protein SVL3 plus a protein involved in actin polymerization and crosslinking to microtubules (CRN1). Finally, cluster 10 contains the two proteins involved in cytokinesis (BNI1, BNR1) plus PFY1, which plays several roles in actin organization and polymerization. These 'hybrid' clusters were expected because the actin patch proteins that closely connect with proteins involved in other processes may contribute to linking these functions together in the cell (Drees *et al*., 2001). As a second biological validation of the clusters, we performed searches using the SGD Gene Ontology Term Finder (http://db.yeastgenome.org/cgi-bin/SGD/GO/goTermFinder) that allows, among other features, to determine the most likely process in which a group of proteins are involved according to Gene Ontology (GO) terms. For the whole set of proteins, we found, as expected, that the general GO process term defined as 'actin cytoskeleton organization and biogenesis' included 20 out of the 34 proteins analyzed, with a probability of those proteins being together by chance equal to 1.03×10^{-29}. For the particular clusters including two or more proteins (because a limitation of the program is that requires at least two proteins in a cluster, thus eliminating clusters 3, 4 and 12 in Figure 4B) we found the following results (considering only the GO terms that gave the lowest probabilities of the cluster occurring by chance): clusters 1 and 7—actin cytoskeleton organization and biogenesis [$p(\text{cluster 1}) = 2.65 \times 10^{-11}$; $p(\text{cluster 7}) = 1.95 \times 10^{-6}$]; cluster 6—Rho protein signal transduction (equivalent to the definition by Drees *et al*. of 'involved to CDC42 signaling'; $p = 1.51 \times 10^{-8}$); cluster 8—G2/M transition of mitotic cell cycle (again related to the assignation of this group of proteins to the morphogenesis checkpoint by Drees *et al*.; $p = 5.78 \times 10^{-5}$); cluster 9—cell growth and/or maintenance ($p = 0.091$); cluster 10—response to osmotic stress ($p = 5.06 \times 10^{-7}$) and cluster 11—actin filament depolymerization ($p = 7.55 \times 10^{-7}$). Only clusters 2 and 5 did not generate significant assignations according to GO terms. These results are interesting in two different ways.

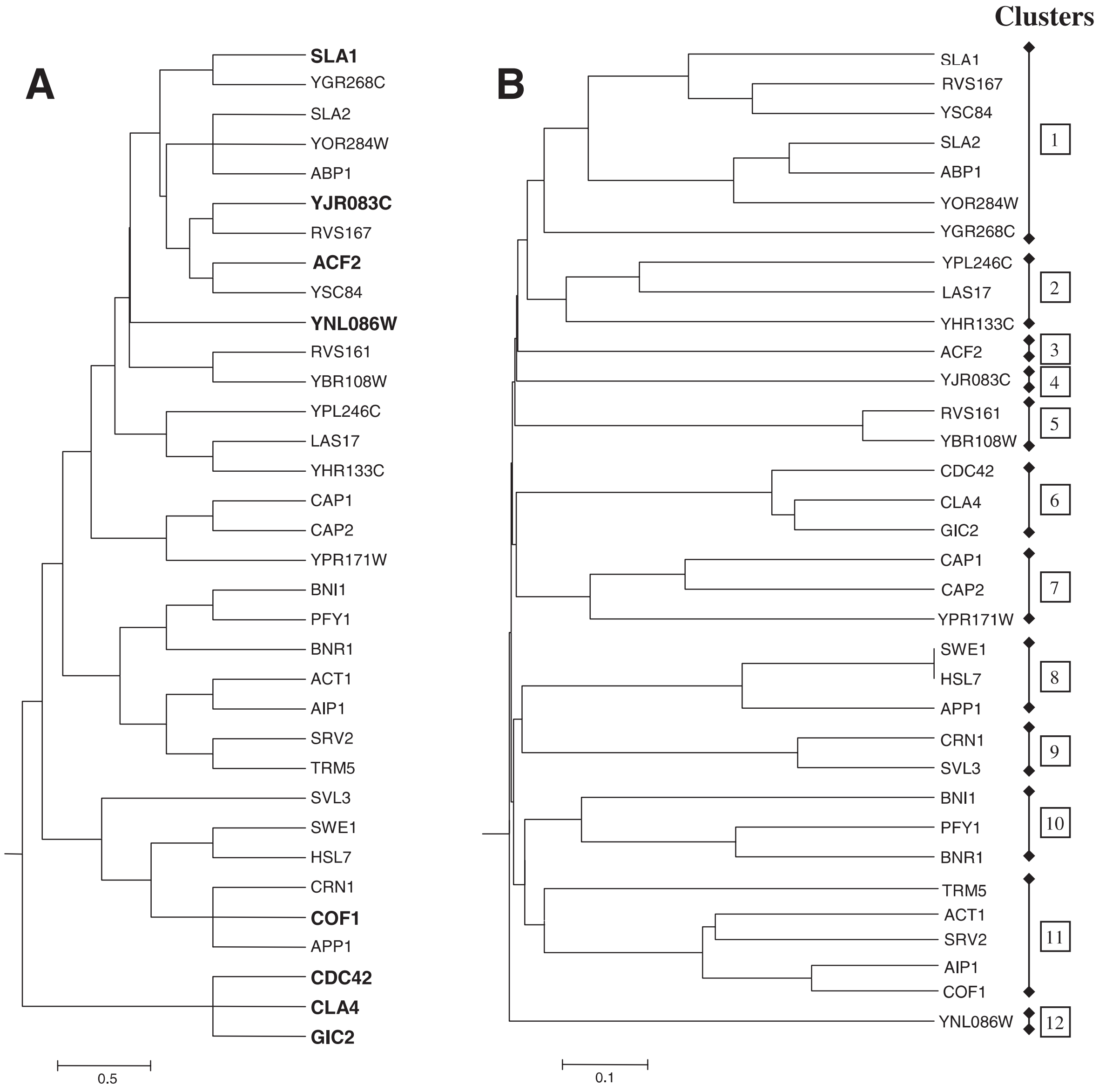

Fig. 4. Comparison of analyses using primary versus secondary distances from the dataset described in Figure 3. (**A**) UPGMA-based tree using the primary distances derived from Figure 3 data. Names in bold indicate deviations with respect to the tree in Figure 4B. (**B**) UPGMA-based tree using the secondary distances estimated using UVCLUSTER (AC $= 100$; $N = 10\,000$). The optimal solution, according to the hypergeometric distribution of direct interactions, consists of 12 clusters, that are detailed on the right side.

First, they demonstrate that proteins in most clusters can be significantly associated to particular cellular processes, thus confirming the biological significance of our results. Second, the fact that the processes assigned are generally different for each cluster suggests that these clusters are composed of proteins involved in closely related but still different processes *in vivo* and that GO terms are precise enough so as to discriminate among those processes.

The possibility of including all available data for a particular species using UVCLUSTER analyses allowed us to ask the question of whether there are other proteins that may be significantly involved in actin patch assembly and function that did not appear in the description by Drees *et al.* (2001). In order to do so, we calculated the average primary distance of each of the 4721 proteins in the DIP database against the 26 proteins that, according to Drees *et al.* (2001) were involved in these processes (the other 8, as we already described above, often showed protein–protein interactions with these 26, but functionally were less related). We found that 19 of those 26 proteins were among the 40 proteins with lowest average distances in the whole dataset (average distance values ranging from 1.31 to 2.23) and even the worst connected protein (TRM5; average distance with the rest equal to 2.65) was in position 199 in our list. These results confirm the results obtained by Drees *et al*. (2001) using the updated information currently available: all the proteins considered by those authors indeed are in close proximity in the *S.cerevisiae* interactome graph. However, we concluded that there are other proteins that are similarly or even better connected. Figure 5 contains a new UPGMA tree obtained using also the whole DIP interaction data, with $\text{AC} = 100$ and $N = 10\,000$, but including 38 other proteins potentially involved in actin patch assembly and function. These 38 proteins have average distances lower than 2.27 to the 26 proteins in the original dataset. The close proximity to actin and interspersion with the proteins of the original dataset of many of those newly added proteins can be easily appreciated in Figure 5. It is also significant that duplicating the number of proteins only alters the relative position of a few proteins of the original dataset. If we compare Figures 4B and 5, we can see that only YPR171W, SLA1, YHR133C, RVS167 and RVS161 appear in very different positions in both the trees. This result can be interpreted as a new confirmation that the delimited clusters were quite reliable. Adding more data in general only contributes to make the groups that were already observed larger. Similar results were obtained using $\text{AC} = 50$ (not shown). As a secondary biological validation of our results, we have also included in Figure 5 some additional data. First, we add information for protein localization from Huh *et al*. (2003) and other sources, as compiled in the MIPS database (Mewes *et al*., 2002; http://mips.gsf.de/). Fifty-six proteins are described in MIPS as being localized to the actin cytoskeleton. Of them, 24 are found in Figure 5, being 16 part of the original data from Drees *et al*. (2001) while the other 8 are among the ones we have secondarily added to our tree using UVCLUSTER analyses (see details in Figure 5). Moreover, another five UVCLUSTER-suggested proteins (LSB3, LSB5, BEM1, ARP2, END3) are localized according to MIPS to the yeast cytoskeleton in a broader class that may also imply interaction with actin. Second, we include also information about the GO term 'actin cytoskeleton organization and biogenesis' that, as we described above, includes most of the proteins in our original dataset. The SGD Gene Ontology Term Finder assigns to that GO term the lowest probability among all terms ($p = 4.3 \times 10^{-34}$) when all the proteins in Figure 5 are included. Moreover, all other terms with low probability are related to it. In total, there are 28 proteins in Figure 5 that are included in the 'actin cytoskeleton organization and biogenesis' GO term. Of them, 8 are from the secondary dataset obtained using UVCLUSTER. These results demonstrate that a significant fraction of the proteins suggested by the UVCLUSTER study of the interactome are related to actin, or at least cytoskeleton, function. Thus, we can conclude that UVCLUSTER-based analyses may significantly contribute to delineate groups of closely integrated proteins in the cell, suggesting members that could be missed when using other approaches.

4.4 UVCLUSTER global analysis correlating gene expression and protein–protein interaction

To demonstrate that UVCLUSTER can be used at a genomic scale, we decided to compare UVCLUSTER-generated results based on protein–protein interaction data with those derived from gene coexpression data obtained using microarrays. There is good evidence, at least in yeast, that the products of highly coexpressed genes interact with a probability much higher than expected by chance (e.g. Kemmeren *et al*., 2002). Thus, we can expect UVCLUSTER to recover, at least in part, clusters of coexpressed genes using protein–protein interaction data. To analyze coexpression, we used as a starting database the yeast 'refined modules' defined by (Bergmann *et al*., 2004). These modules are groups of genes that show a significant level of coexpression and that were obtained starting with a core of evolutionary conserved, coexpressed genes to which other genes with similar expression patterns were added Bergmann *et al*. (2004). These authors described eight of those modules, including a total of 548 genes, as detailed in Table 1. We decided to check to which extent UVCLUSTER analyses may unearth those same modules using the available protein–protein interaction data. We therefore took the list of 548 genes (that was reduced to 543 after eliminating 5 genes that appeared in two or more modules) and analyzed whether the DIP database contains interaction data for their products. We found that the products of 376 of those genes indeed were found in DIP. However, only 163 were involved in more than one protein–protein interaction. These results demonstrate that the amount of information for the protein products of this set of genes is quite limited. We then used UVCLUSTER ($\text{AC} = 100$, $N = 10\,000$) to determine clusters of proteins based on interactome data. Results are shown in Figure 6. A total of 162 proteins formed several well-defined clusters while the other 214 appeared isolated, due to the fact that they did not present direct interactions with any other proteins in the dataset (these last ones have been grouped together in a single branch in Figure 6). As shown in the figure, several of the clusters closely follow the results obtained by Bergmann *et al*. (2004). Thus, a cluster contains 56 rRNA processing proteins plus 14 'contaminant' proteins involved in other processes. Another has 29 proteasome proteins with a single 'contaminant', a third one includes 17 MRP proteins with two 'contaminants', etc. We became aware when considering these results that the information in DIP is indeed quite fragmentary, which explains why some very characteristic clusters are missing in Figure 6. For example, DIP does not contain interaction data corresponding to cytoplasmic ribosomes (explaining why a large ribosomal cluster does not appear), or to the small subunit of mitochondrial ribosomes (therefore, the 17 MRP proteins that appear together are all part of the large subunit). Several of the modules (e.g. glycolysis,

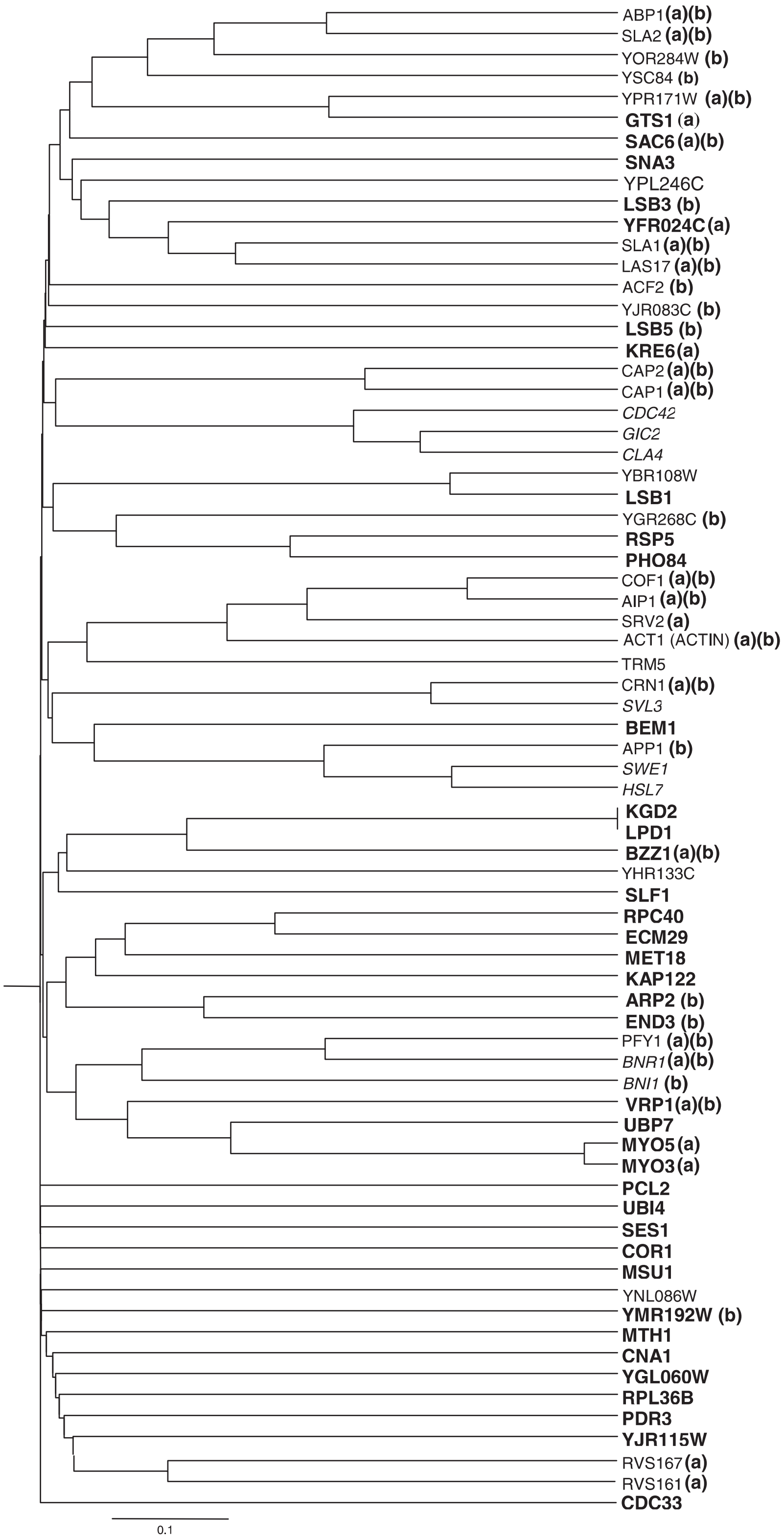

Fig. 5. Results of the analysis to discover new actin patch assembly and endocytosis proteins. In bold, the new proteins discovered. Italics refer to the eight proteins that Drees *et al*. (2001) considered as belonging to processes other than patch assembly and patch-mediated endocytosis. (a): Proteins that are localized to the actin cytoskeleton according to Huh *et al*. (2003). (b): Proteins assigned to the GO process ‘actin cytoskeleton organization and biogenesis’ according to the SGD database (see http://www.yeastgenome.org/GOContents.shtml).

Table 1. UVCLUSTER results using the refined modules by Bergmann *et al.* (2004)

Module	No. of genes in module	No. of proteins in clusters defined by UVCLUSTER	Size of main cluster defined by UVCLUSTER (% of total genes in module)
rRNA processing	192	74	56 (29.2)
Ribosomal protein	153	20	5 (3.3)
Mitochondrial ribosomal protein	76	19	17 (22.3)
Proteasome	40	31	29 (72.5)
Peroxide	29	2	0 (0.0)
Secreted protein	28	8	4 (14.3)
Glycolysis	15	2	0 (0.0)
Heat shock	15	6	5 (33.3)

peroxide) are moreover formed in part by proteins that are not expected to physically interact. Those results explain why only some of the modules defined by Bergmann *et al.* (2004) actually are found in UVCLUSTER analyses (see Figure 6 and Table 1 for the details). In any case, we can conclude that UVCLUSTER can be used for analysis involving hundreds of proteins and, even in those complex cases, it recovers interesting functional information based on interactome data. On the other hand, the analysis just shown casts doubts on whether interaction data alone will ever provide a detailed picture of cell function. The relative incompleteness of the results we have just described can be conservatively explained suggesting that the available information is very fragmentary and will significantly improve in the future. However, it could also be explained by some/many highly integrated cellular processes, which do not require direct protein–protein interactions. This would hinder the processes to be identified, no matter how exhaustive the interactome data are.

5 DISCUSSION

UVCLUSTER is a flexible tool for global exploration of protein function using interactome data that is based on iterative hierarchical clustering. The strategy implemented in our program is related to permutation tests commonly used in other contexts (reviewed by Felsenstein, 2004, pp. 359–363), which are applied for the first time to the resolution of the 'ties in proximity' problem that arises in clustering methods when many distances are equal. We have shown that UVCLUSTER has four main strengths. First, it may easily discover and define sets of closely linked proteins. Secondary distance data among the elements of a set delineate their relationships in a way that their direct, primary distances cannot do (see Figures 2 and 4). Second, UVCLUSTER may be used to discover proteins involved in a particular process, when provided with some preliminary information. For example, if we know that a group of proteins are acting on a relevant process, we can use them as 'seeds' to delimit, using UVCLUSTER, all the proteins that are closely connected to them in the interactome. Therefore, we can obtain a clearer picture of the functions of the set of interesting proteins (see Figure 5 and the related analyses explained above). This approach can be seen as complementary to the one based on the detection of highly connected graphs that include a protein of interest developed by Bader and Hogue (2003; 'directed mode'). Third, UVCLUSTER can be also used to establish groups of connected proteins even when some information is unavailable, by being able to predict potential interactions. This can be done by using AC values lower than 100, that is, allowing proteins that do not directly interact, but that are still quite close in the interactome, to be clustered together. We have observed that diminishing the AC value has the effect of generating topologies that are worse than the one obtained with AC = 100 when evaluated with the hypergeometric distribution-based parameter described above. This is due to the effect that the increased permissiveness of the clustering process has on the secondary distances, often allowing directly connected proteins to appear in different clusters. However, this relaxation of the conditions may be of great interest if we assume that data are incomplete and some distances may be actually lower than those available. In those cases, relaxed analyses may suggest potential clusters of interacting proteins that would be missed by the strictest ones. It is significant that UVCLUSTER can also be used in the opposite direction, that is, in order to detect false positive interactions. For example, promiscuous proteins, which are able to interact with many different partners, can be easily detected using UVCLUSTER (with AC = 100) when they appear as being equally distant to many other, totally unrelated, proteins (Arnau and Marín, 2003). This may be another significant application of our program, because, as we commented above, the number of false positives is considered to be high for interaction data generated by non-directed, large-scale experiments (e.g. von Mering *et al.*, 2002). As a final potential application, UVCLUSTER can be used to quickly explore whether proteins encoded by orthologous genes retain related functions in two species by comparing the relative positions in both of their the interactomes. We have already generated relevant information for some conserved genes by comparing the *S.cerevisiae* and *D.melanogaster* interactomes, which will be presented elsewhere.

UVCLUSTER has been designed keeping in mind the needs of groups working in functional biology. Potential users of UVCLUSTER are those researchers interested in obtaining in a short period of time significant information for parts of the interactome of a species, for example, including all the proteins involved in the particular biological process that they are studying. Therefore, analyses involving tens to hundreds of elements, similar to the ones we have shown above are

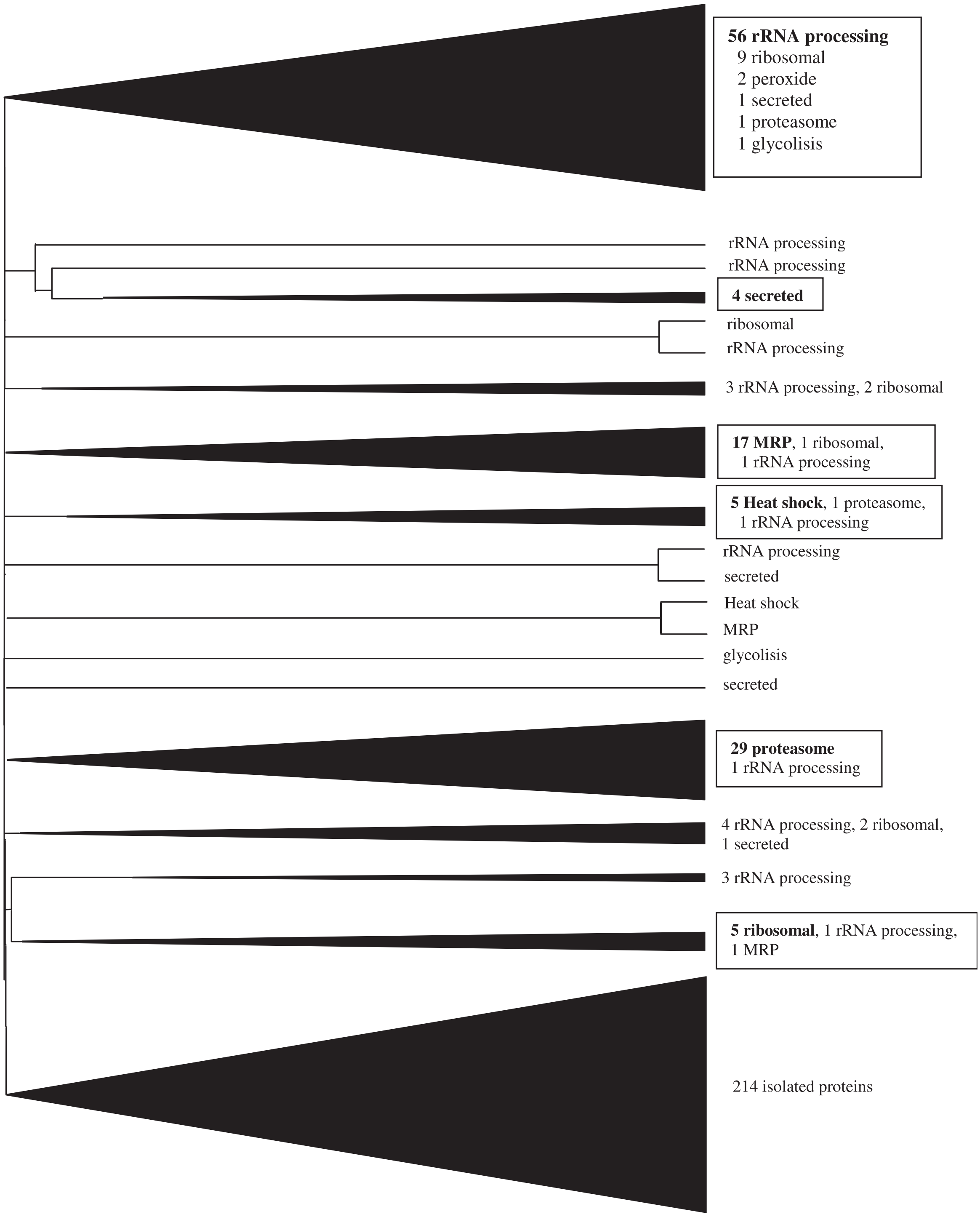

Fig. 6. UVCLUSTER results for the refined modules defined by Bergmann *et al*. (2004). See text for details.

expected to be the most frequently performed. It is important that, for our datasets, we have found that secondary distance results do not significantly change by increasing N above a certain threshold. Our experience with the program suggests that it is reasonable to use a value of N of at least ten times the number of elements to generate reliable secondary distance tables. For example, when we repeatedly analyzed the actin dataset with $N = 100$, the topology shown in Figure 4B (obtained when $N = 10\,000$) appeared only in 7 out of 10 cases (in the other three, clusters 1 and 2 appeared fused together). However, 10 repetitions with $N = 500$ generated the same clusters shown in Figure 4B. This recommendation of using at least 10 times the number of elements sharply contrasts with the suggestions of several well-known computer programs, which suggest to perform about 10 replicates when there are ties [summarized in Backeljau *et al*. (1996) and recently confirmed by our group].

UVCLUSTER analyses involving up to 1000 elements (e.g. 1000 proteins) can be, as the times described previously demonstrate, easily performed on standard computer equipment. However, our general guideline to select the number of iterations suggests that UVCLUSTER, in its current implementation, has severe limitations to analyze whole-proteome data. For example, for the whole *S.cerevisiae* dataset, which is defined by about 11 millions of primary distances, our guideline suggests performing analyses with *N* values of at least 50 000. Such a huge analysis is evidently not feasible on standard PC equipment (estimated time: 6100 h, with AC = 100), showing that a parallel implementation of UVCLUSTER, whose development is already in progress by our group, is essential for analyses involving very large datasets.

Finally, it is obvious that UVCLUSTER may also be used to analyze information other than protein interaction data. Any type of information that can be converted into primary distances and that suffers from the ties in proximity problem can be advantageously analyzed using UVCLUSTER. Typical examples in genomics are the analyses of connections among protein domains, in which distances are defined depending on the combination of domains found in the set of proteins of one or multiple species (e.g. Mott *et al*., 2002; Ye and Godzik, 2004) or the analysis of distances estimated from genetic interaction screens (Tong *et al*., 2004). Other fields, as the analysis of paper citations or coauthorships (Albert and Barabási, 2002) could also benefit from the use of UVCLUSTER.

ACKNOWLEDGEMENTS

Research supported by the Spanish Ministry of Science and Technology (MCYT; projects GEN2001-4851-C06-02 [*NEUROGENOMICA*/CAGEPEP] and SAF2003-09506), Fundació *La Caixa* (01/080-00) and Generalitat Valenciana (GV04B-141). S.M. is the recipient of an FPU predoctoral fellowship from the MCYT.

REFERENCES

Albert,R. and Barabási,A.L. (2002) Statistical mechanics of complex networks. *Rev. Modern Phys.*, **74**, 47–97.

Arnau,V. and Marín,I. (2003) A hierarchical clustering strategy and its application to proteomic interaction data. *Lect. Notes Comput. Sci.*, **2652**, 62–69.

Backeljau,T., De Bruyn,L., De Wolf,H., Jordaens,K., Van Dongen,S. and Winnepenninckx,B. (1996) Multiple UPGMA and Neighbor-joining trees and the performances of some computer packages. *Mol. Biol. Evol.*, **13**, 309–313.

Bader,G.D. and Hogue,C.W.V. (2002) Analyzing yeast protein–protein interaction data obtained from different sources. *Nat. Biotechnol.*, **20**, 991–997.

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

Bader,G.D., Heilbut,A., Andrews,B., Tyers,M., Hughes,T. and Boone,C. (2003) Functional genomics and proteomics: charting a multidimensional map of the yeast cell. *Trends Cell Biol.*, **13**, 344–356.

Barabási,A.L. and Oltvai,Z.N. (2004) Network biology: understanding the cell's functional organization. *Nat. Rev. Genet.*, **5**, 101–113.

Bergmann,S., Ihmels,J. and Barkai,S. (2004) Similarities and differences in genome-wide expression data of six organisms. *PLoS Biol.*, **2**, 0085–0093.

Bu,D., Zhao,Y., Cai,L., Xue,H., Zhu,X., Lu,H., Zhang,J., Sun,S., Ling,L., Zhang,N., Li,G. and Chen,R. (2003) Topological structure analysis of the protein–protein interaction network in budding yeast. *Nucl. Acids Res.*, **31**, 2443–2450.

Drees,B.L., Sundin,B., Brazeau,E., Caviston,J.P., Chen,G.C., Guo,W., Kozminski,K.G., Lau,M.W., Moskow,J.J., Tong,A. *et al*. (2001) A protein interaction map for cell polarity development. *J. Cell Biol.*, **154**, 549–571.

Everitt,B.S., Landau,S. and Leese,M. (2001) *Cluster Analysis*, 4th edn. Arnold, London.

Felsenstein,J. (2004) *Inferring Phylogenies*. Sinauer Associates, Inc. Sunderland, MA.

Floyd,R.W. (1962) Algorithm 97—Shortest path. *Commun. ACM*, **5**, 345.

Gagneur,J., Krause,R., Bouwmeester,T. and Casari,G. (2004) Modular decomposition of protein–protein interaction networks. *Genome Biol.*, **5**, R57.

Gavin,A.C., Bösche,M., Krause,R., Grandi,P., Marzioch,M., Bauer,A., Schultz,J., Rick,J.M., Michon,A.M., Cruciat,C.M. *et al*. (2002) Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, **415**, 141–147.

Gibbons,F.D. and Roth,F.P. (2002) Judging the quality of gene expression-based clustering methods using gene annotation. *Genome Res.*, **12**, 1574–1581.

Giot,L., Bader,J.S., Brouwer,C., Chaudhuri,A., Kuang,B., Li,Y., Hao,Y.L., Ooi,C.E., Godwin,B., Vitols,E. *et al*. (2003) A protein interaction map of *Drosophila melanogaster*. *Science*, **302**, 1727–1736.

Goldberg,D.S. and Roth,F.P. (2003) Assessing experimentally derived interactions in a small world. *Proc. Natl Acad. Sci., USA*, **100**, 4372–4376.

Gordon,A.D. (1999) *Classification*, 2nd edn. Chapman and Hall/CRC, Boca Ratón, FL.

Ho,Y., Gruhler,A., Hellbut,A., Bader,G.D., Moore,L., Adams,S.L., Millar,A., Taylor,P., Bennett,K., Boutllier,K. *et al*. (2002) Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry. *Nature*, **415**, 180–183.

Huh,W.K., Falvo,J.V., Gerke,L.C., Carroll,A.S., Howson,R.W., Weissman,J.S. and O'Shea,E.K. (2003) Global analysis of protein localization in budding yeast. *Nature*, **245**, 686–691.

Ito,T., Chiba,T., Ozawa,R., Yoshida,M., Hattori,M. and Sakaki,Y. (2001) A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc. Natl Acad. Sci., USA*, **98**, 4569–4574.

Kemmeren,P., van Berkum,N.L., Vilo,J., Bijma,T., Donders,R., Brazma,A. and Holstege,F.C. (2002) Protein interaction verification and functional annotation by integrated analysis of genome-scale data. *Mol. Cell*, **9**, 1133–1143.

Kumar,S., Tamura,K., Jakobsen,I.B. and Nei,M. (2001) MEGA2: molecular evolutionary genetics analysis software. *Bioinformatics*, **17**, 1244–1245.

Levenstien,M.A., Yang,Y and Ott,J. (2003) Statistical significance for hierarchical clustering in genetic association and microarray expression studies. *BMC Bioinformatics*, **4**, 62.

Li,S., Armstrong,C.M., Bertin,N., Ge,H., Milstein,S., Boxem.M., Vidalain,P.O., Han,J.D.J., Chesneau,A., Hao,T. *et al*. (2004) A map of the interactome network of the metazoan *C.elegans*. *Science*, **303**, 540–543.

MacCuish,J., Nicolaou,C. and MacCuish,N.E. (2001) Ties in proximity and clustering compounds. *J. Chem. Inf. Comput. Sci.*, **41**, 134–146.

Mewes,H.W., Frishman,D., Guldener,U., Mannhaupt,G., Mayer,K., Mokrejs,M., Morgenstern,B., Munsterkotter,M., Rudd,S. and Weil,B. (2002) MIPS: a database for genomes and protein sequences. *Nucl. Acids Res.*, **30**, 31–34.

Mott,R., Schultz,J, Bork,P. and Ponting,C.P. (2002) Predicting protein cellular localization using a domain projection method. *Genome Res.*, **12**, 1168–1174.

Nei,M. and Kumar,S. (2000) *Molecular Evolution and Phylogenetics*. Oxford University Press, New York.

Orlev,N., Shamir,R. and Shiloh,Y. (2004) PIVOT: protein interactions visualization tool. *Bioinformatics*, **20**, 424–425.

Pereira-Leal,J.B., Enright,A.J. and Ouzounis,C.A. (2004) Detection of functional modules from protein interaction networks. *Proteins*, **54**, 49–57.

Prinz,S., Avila-Campillo,I., Aldridge,C., Srinivasan,A., Dimitrov,K., Siegel,A.F. and Galitski,T. (2004) Control of yeast filamentous-form growth by modules in an integrated molecular network. *Genome Res.*, **14**, 380–390.

Przulj,N., Wigle,D.A. and Jurisica,I. (2004) Functional topology in a network of protein interactions. *Bioinformatics*, **20**, 340–348.

Quackenbush,J. (2001) Computational analysis of microarray data. *Nat. Rev. Genet.*, **2**, 418–427.

Rives,A.W. and Galitski,T. (2003) Modular organization of cellular networks. *Proc. Natl Acad. Sci., USA*, **100**, 1128–1133.

Salwinski,L. and Eisenberg,D. (2003) Computational methods of analysis of protein–protein interactions. *Curr. Opin. Struct. Biol.*, **13**, 377–382.

Schwikowski,B., Uetz,P. and Fields,S. (2000) A network of protein–protein interactions in yeast. *Nat. Biotechnol.*, **18**, 1257–1261.

Spirin,V. and Mirny,L.A. (2003) Protein complexes and functional modules in molecular networks. *Proc. Natl Acad. Sci., USA,* **100**, 12123–12128.

Takezaki,N. (1998) Tie trees generated by distance methods of phylogenetic reconstruction. *Mol. Biol. Evol.*, **15**, 727–737.

Tong,A.H.Y., Lesage,G., Bader,G.D., Ding,H., Xu,H., Xin,X., Young,J., Berriz,G.F., Brost,R.L., Chang,M. *et al*. (2004) Global mapping of the yeast genetic interaction network. *Science*, **303**, 808–813.

Uetz,P., Giot,L., Cagney,G., Mansfield,T.A., Judson,R.S., Knight,J.R., Lockshon,D., Narayan,V., Srinivasan,M., Pochart,P. *et al*. (2000) A comprehensive analysis of protein–protein interactions in *Saccharomyces cerevisiae*. *Nature*, **403**, 623–627.

von Mering,C., Krause,R., Snel,B., Cornell,M., Oliver,S.G., Fields,S. and Bork,P. (2002) Comparative assessment of large-scale data sets of protein–protein interactions. *Nature*, **417**, 399–403.

Wagner,A. (2001) The yeast protein interaction network evolves rapidly and contains few redundant duplicate genes. *Mol. Biol. Evol.*, **18**, 1283–1292.

Wilhelm,T., Nasheuer,H.P. and Huang,S. (2003) Physical and functional modularity of the protein network in yeast. *Mol. Cell. Proteom.*, **2**, 292–298.

Xenarios,I., Salwinski,L., Duan,X.J., Higney,P., Kim,S.M. and Eisenberg,D. (2002) DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions. *Nucl. Acids Res.*, **30**, 303–305.

Ye,Y. and Godzik,A. (2004) Comparative analysis of protein domain organization. *Genome Res.*, **14**, 343–353.