Journal of Theoretical Biology 258 (2009) 53–59

Contents lists available at ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/yjtbi

Basic networks: Definition and applications

Ignacio Marín [a,*], Sergio Hoyas [b]

[a] Instituto de Biomedicina de Valencia, Consejo Superior de Investigaciones Científicas (IBV-CSIC), Calle Jaime Roig, 11, Valencia 46010, Spain
[b] Departamento de Informática, Universidad de Valencia, Burjassot, Spain

ARTICLE INFO

Article history:
Received 18 June 2008
Received in revised form
21 January 2009
Accepted 21 January 2009
Available online 3 February 2009

Keywords:
Graph
Module
Interactome
Steiner tree

ABSTRACT

We define basic networks as the undirected subgraphs with minimal number of units in which the distances (geodesics, minimal path lengths) among a set of selected nodes, which we call seeds, in the original graph are conserved. The additional nodes required to draw the basic network are called connectors. We describe a heuristic strategy to find the basic networks of complex graphs. We also show how the characterization of these networks may help to obtain relevant biological information from highly complex protein–protein interaction data.

1. Introduction

In biology, as in other sciences, many types of complex data can be expressed as undirected graphs, in which units (nodes) are connected by bidirectional edges. Among the most significant examples are protein–protein interaction networks, networks based on cooccurrence of protein domains or graphs generated by assigning edges to genes with a given level of coexpression (Aittokallio and Schwikowski, 2006; Sharan et al., 2007). All these graphs have emerged very recently, with the advent of genomic and proteomic high-throughput technologies that have generated huge amounts of data. How to obtain useful information from this kind of data is one of the main challenges in modern biology.

A significant problem occurs when a set of units of one of those graphs is selected by any given criteria and we are interested in determining its functional or structural meaning. There are two ways of tackling this situation. First, we can exclusively use information not included in the graph. For example, if we are trying to understand the function of a set of proteins, we can simply explore the available literature or, if possible, we can use standard classifications, such as gene ontologies (GOs), in which gene products are classified according to the biological processes in which they act (Ashburner et al., 2000). These approaches are very useful if the available information is sufficient, but may fail if the set contains many proteins of unknown function. A second alternative is to explore the graph to determine the context in which the selected units are working. For example, in a protein–protein interaction network, we can determine whether most of the selected proteins are located in a given region of that network. If this is the case, we can then use the information provided by proteins not included in our set, but closely linked to ours, to infer the functions that characterize our dataset. Conceptually, this second strategy can be schematized as follows. First, our set defines a group of *seed* nodes from which to start the analysis. Second, we characterize a second group of nodes which are sufficiently closely linked to the seed nodes, which may be called *connectors*. Third, we use the information provided by the connectors to infer information about the seeds. The problem then consists in defining the best way to determine the connectors. In the current literature, this has been generally solved by devising strategies to define modules (e.g. Del Rio et al., 2001; Bader, 2003; Asthana et al., 2004; Hashimoto et al., 2004; Krauthammer et al., 2004; Arnau et al., 2005; Scott et al., 2005; Lubovac et al., 2006; Lucas et al., 2006; Li and Horvath, 2007; see review by Sharan et al., 2007) A module is loosely defined as a group of closely linked nodes, with an internal cohesion that allows its separation from the rest of units in the network. The problem is that the characterization of modules is based on conventional, a priori criteria of unknown efficiency. The ability of any module definition to efficiently characterize connectors will depend on the features of the graph (size, connectivity) and the proximity among the seeds, and different definitions may lead to quite different results (e.g. Luo et al., 2007). In fact, to define modules may be difficult if the seeds are very distant.

The starting point of our work is the appreciation that any strategy able to determine significant connectors which is based on non-conventional criteria could be a significant advance in network exploration. We introduce here the concept of *basic*

* Corresponding author. Tel.: +34 963393770; fax: +34 963690800.
E-mail addresses: imarin@ibv.csic.es (I. Marín), sergio.hoyas@uv.es (S. Hoyas).

0022-5193/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jtbi.2009.01.022

network and we show that it allows for a non-ambiguous definition of significant connectors. We describe a heuristic strategy to characterize the significant connectors and also, using real examples, we show the usefulness of the characterization of basic networks to explore complex biological graphs.

2. Concept of basic network and some theoretical considerations

We define *basic network* of an undirected, fully connected graph as the fully connected, undirected subgraph which includes a set of preselected units (seeds) plus the minimal number of connectors required to conserve the distances (geodesics, minimal path lengths) found among the seeds in the original graph. We define as *basic units* the set of seeds plus the set of connectors that are required to generate the basic network. This simple definition lacks any ambiguity: for any starting graph and set of seeds, there is at least one basic network (see e.g. Fig. 1A). However, it will occur often that more that one basic network, all with identical number of connectors, can be obtained. A typical example of *tied basic networks* is shown in Fig. 1B. A tied basic network can be further simplified (*pruned*; see below) to obtain several basic networks (Fig. 1B). It is very simple to demonstrate that randomly eliminating nodes from the graph will not necessarily lead to obtaining the basic network. On the contrary, random elimination of nodes will generally lead to subgraphs that cannot be further minimized but are much larger than the basic network of the graph (Fig. 1C). We call these results *local minimal networks*. Thus, any strategy to determine the basic network(s) of a graph will have to effectively deal with the presence of local minimal networks.

A significant theoretical consideration is the relationship among Steiner trees and basic networks. In the context of undirected graphs in which all edges are equivalent, a Steiner tree for a given set of seeds is defined as the tree that minimizes the number of edges (i.e. the total distance) to connect those seeds (Chartrand and Zhang, 2004). Although the goal of both Steiner trees and basic networks is to establish minimal subgraphs determined by the disposition of certain seeds in a starting graph, it is easy to demonstrate that they are often unrelated. The main reason is that, to obtain the basic networks, we demand the minimal distances among seeds to be conserved, while this is not the case for Steiner trees. Thus, although the basic network of a graph can coincide with the Steiner tree (Fig. 2A), it will most often occur that the basic network is larger and totally different from the Steiner tree (Fig. 2B). As can be seen in Fig. 2B, and can be also easily visualized will occur in larger graphs, Steiner trees are based on finding "internal" units (i.e. units that are quite away from the seeds and more or less equidistant from all of them) to minimize the total path length. Thus, in Steiner trees the seeds may end up at large distances. On the contrary, basic networks select "external" units, which are as closely linked to the seeds as possible.

We may now ask why the definition of basic networks may be significant in the exploration of complex graphs. There are three main reasons. First, the close proximity among seeds and connectors just mentioned is obviously a useful feature, if we wish to obtain information about the seeds. Second, as we already

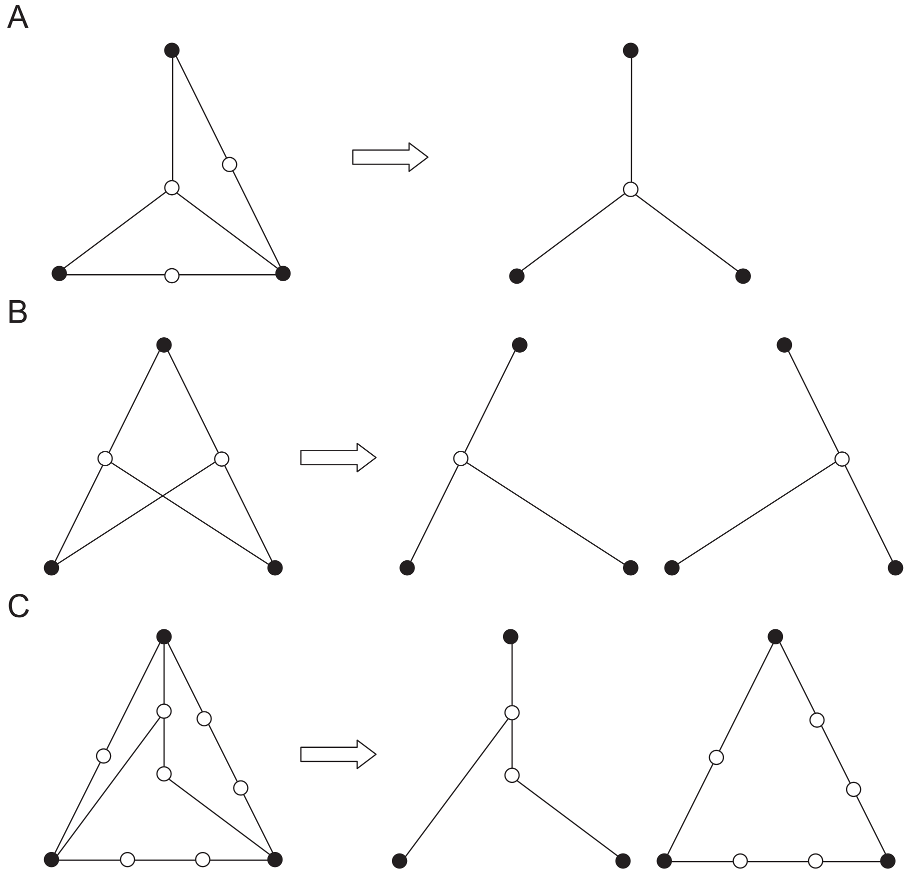

Fig. 1. Three situations that may occur when searching for basic networks: (A) in the simplest case, a single basic network exists, and all connectors are basic units; (B) a tied basic network cannot be further simplified without pruning (left). When pruning is performed, multiple alternative basic networks are found (right). (C) Randomly eliminating nodes may lead to the basic network (left) but, most often, will lead to local minimal networks (right; this would be the optimal solution if any of the two internal nodes is eliminated).

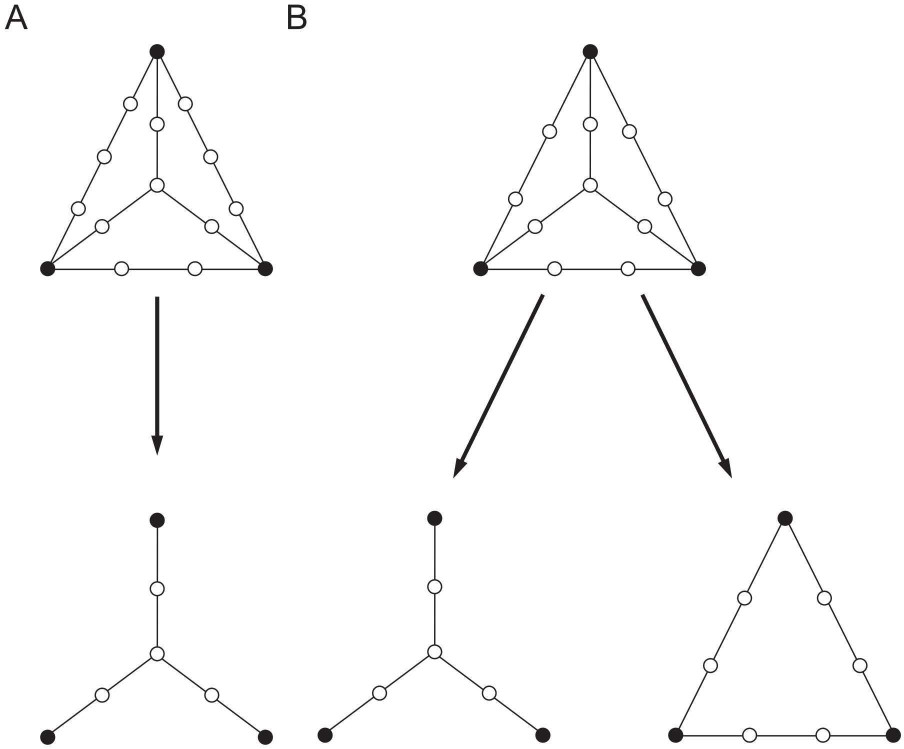

Fig. 2. Relationships between Steiner trees and basic networks: (A) when the only geodesic among the seeds is through the Steiner tree, the basic network and the Steiner tree are identical; (B) a typical example in which the Steiner tree (left) is totally different from the basic network (right), for the same starting graph and starting seeds (black dots). The shortest paths do not include the nodes in the Steiner tree.

discussed above, the definition of basic network is not ambiguous, avoiding the potential pitfalls of all module definitions. This has another direct advantage: the fact that basic networks have a given, clear-cut number of seeds allows for their statistical comparison with *random basic networks*, which are those obtained by randomly taking units from the same starting graph. Thus, we can demonstrate that some seeds are unrelated if the basic network obtained starting with them is not significantly different in size from a set of random basic networks. Finally, by determining the minimal number of basic units, we focus on the nodes that best explain the connections among the seeds. These must necessarily be hubs, highly connected units, given that only hubs may contribute to minimize its size.

3. A heuristic strategy to characterize basic networks

It is obvious that it is impossible to test for all the possible combinations of connectors to obtain the basic network of any large graph. Thus, a heuristic strategy must be devised to obtain the basic network, or at least to get a good approximation to it, avoiding falling in local minimal networks. We have devised a strategy which is implemented in a program that we have called *Netbasic*. Although the algorithmic details of the program are complicated, and will be described elsewhere, the strategy is actually quite simple to understand. The basic pseudocode of the program is as follows:

```
(1) Mark seeds as basic units.
(2) Compute the distances among the seeds.
(3) For each seed, define seed+1 units as the second node in any
    geodesic that starts in that seed and ends in a different seed.
(4) Determine all the basic seed+1 units, which are those that
    have no alternative seed+1 units and therefore are essential to
    connect two or more seeds with a a minimal number of steps.
    Mark them as basic units.
(5) Remove paths among seeds, following these rules:
    (a) If a seed+1 unit is basic, none of the paths in which it is
        included can be eliminated.
    (b) Count paths among seeds for each seed+1 unit. If two or
        more seed+1 units have alternative paths, which connect
        the same two seeds, but one of them has assigned more
        total paths than the rest, the paths that pass through the
        rest of seed+1 units are eliminated.
    (c) In case of ties in the number of paths, all paths are kept
        unless one of the tied seed+1 units is basic. If that occurs,
        alternative paths through non-basic units are eliminated.
    (d) Determine again whether seed+1 units are basic.
    (e) Repeat steps 5a–c until there are no more eliminations.
(6) Remove nodes that lack paths.
(7) If needed, define seed+2 units (and, if needed, seed+3 units,
    seed+4 units,…, seed+n units) and repeat the process
    described in steps 4–6 until all nodes are either marked as
    basic or no additional node can be eliminated.
```

Steps 1–4 are easy to follow. Most critical is step 5. In it, seed+1 units are compared and paths that go through basic or highly connected units (i.e. those that are included in many paths among seeds) are conserved, while those that go through non-basic, poorly connected units are eliminated. The same applies to step 7, for seed+2 units, seed+3 units, etc. These steps very effectively eliminate most units that are very unlikely to be part of the basic network, given its scarce participation in connecting the seeds. Thus, this strategy is based on keeping hubs and eliminating less connected units. A simple example of how this strategy works is shown in Fig. 3.

Even though this heuristic search is logical, it is obviously not certain it will obtain the basic network. As we already described

1. Mark seeds (**A**, **B**, **C**, and **D**) as basic. (Panel A).
2. Define seeds+1 (all the rest of nodes in this particular case). Detect seeds+1 included in the minimal paths among seeds: A-B: **1, 3**, **5**; A-C: **3** ; A-D: **4**, **5** ; B-D: **5**, **C**
3. Mark basic seeds+1: in this case, **3** (needed for path A3C) (Panel B)
4. Compute number of minimal paths for each seed+1: **5**: 3; **3**: 2; **1**, **4**: 1
5. For each pair of seeds, find the seed+1 with more paths. Remove all the paths associated to other, non-basic seeds+1. In case of tie, remove paths from non-basic seeds, if any. In this case, A3B eliminates A1B, A5B. A5D eliminates A4D; BCD eliminates B5D.
6. Go to step 3 and repeat until all the seeds+1 are basic (in this case, this is not needed).
7. Remove pathless seeds and non-used links. In this case, we are done: all the units left are basic, so we have determined the basic network (Panel C) If this is not the case, we should start characterizing the seed+2 nodes and go again to step 3.

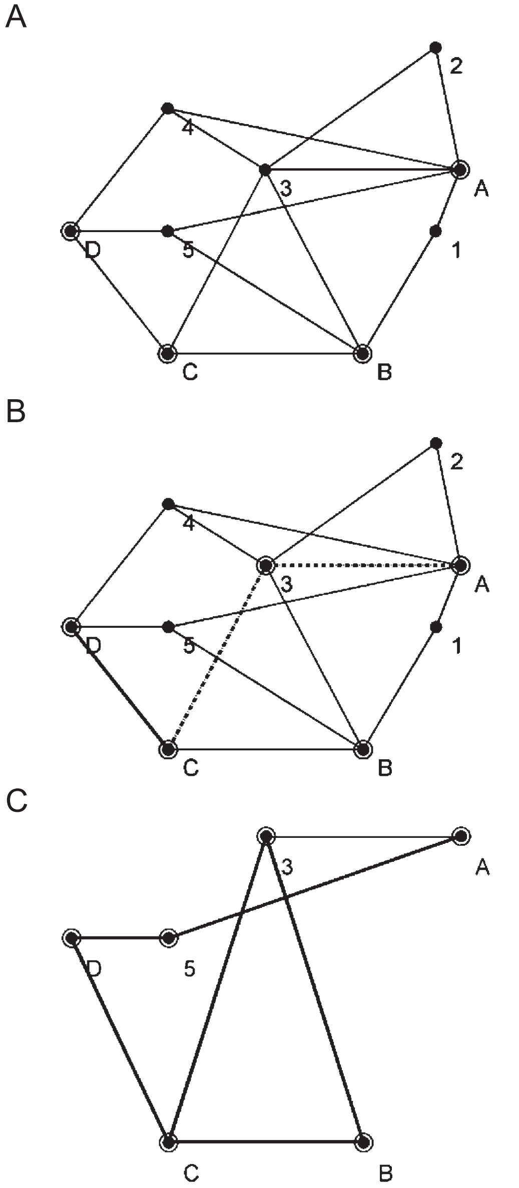

Fig. 3. An example of how to obtain basic networks. Units are named in bold. **A**, **B**, **C**, **D** are the seed units. The whole graph contains only nine units. Basic units are indicated with double circles.

above, the two complications are (1) ties and (2) local minimal networks. To deal with them, we have included refinements to the basic strategy. First, after the program goes through steps 1–7, it determines whether all nodes left are basic. If not, this may be caused by tied solutions as those that we shown in Fig. 1B. If this is the case, the program can, if the user is interested, to further minimize the graph by checking in turn which connectors can be eliminated without affecting the distances among the seeds, in a process that we have called *pruning*. Once the pruning has been completed, ties are eliminated. However, the pruning process also eliminates the information about alternative solutions of the basic graph and, in many cases, it may be interesting to consider the full solution (see below). The second complication is much more difficult to cope with, given that to avoid the problem of local minimal networks with any heuristic search is intrinsically impossible, except in trivial cases. Thus, we have chosen to minimize the problem by using a permutation method. In steps 5 and 7, it is obviously critical the order in which seed+*n* units are considered. The process of path elimination depends on counting paths, and that count varies each time that a seed+*n* unit has been processed. Thus, we can easily envisage a situation in which taking the seed+*n* units in a given order will lead to a local minimal network, while considering them in a different order will lead to obtaining the basic network. By randomly taking the seeds and performing many alternative runs of the program (10^3–10^5), we can avoid as much as possible this problem. The multiple runs also give us a clear idea of the variation among the solutions obtained, and which is the set of proteins that appear in all the solutions and what proteins appear less frequently.

4. Searching for the basic networks in real data

To demonstrate the potential of this method, we have explored the basic network obtained when some sets of proteins of the yeast *Saccharomyces cerevisiae* are taken as seeds. The starting graph is the whole interactome of this species. This interactome, the largest known for any eukaryote, currently (April 2008) contains 4939 proteins connected by 38 645 links (data from the Biogrid database; release 2.0.36; Stark et al., 2006).

The first set of seeds included 43 proteins, which were chosen because they are included in the *Mitochondrial large ribosomal subunit* Gene Ontology term (obtained from the *Saccharomyces genome database* (SGD); http://www.yeastgenome.org/). Fig. 4 shows the best solution obtained without pruning the data, which contains 20 connectors in addition of the 43 original seeds. The average number of connectors in our set of solutions, without pruning, was 22±1. Given that the seed proteins belong to an organule and thus often directly interact (see Fig. 4), the solutions obtained were expected to be much better than those obtained with random starting seeds. Indeed, the average number of connectors in analyses with 43 random seeds, again without pruning, was 207±27 (Table 1). The pruned results were very similar (see also Table 1).

Using this example, we can show how to use the concept of basic network to obtain significant biological information, based on the connectors found. Out of the 20 connectors detected in this analysis, four stand out as having many direct connections with multiple large subunit proteins. However, only three of them (MRP4, YLH47 and MDM38; white dots in Fig. 4) are annotated in SGD as being related with the large subunit of the mitochondrial ribosome. The case of MRP4 is the easiest to understand. MRP4 is a protein of the small subunit of the mitochondrial ribosome which has been successfully used to capture many proteins of both the small and large ribosomal subunits (Gan et al., 2002). Also not surprising is to find the very similar proteins YLH47 and MDM38, which are related to human Letm1, a protein encoded by a gene associated to Wolf–Hirschhorn syndrome. These proteins are located in the inner membrane of the mitochondria and both are known to interact with mitochondrial ribosomes. MDM38 interacts with nascent proteins and is involved in the transport of proteins across the mitochondrial inner membrane (Frazier et al., 2006). The recovery of YLH47 and MDM38 as connectors thus confirms their close functional relationship with the yeast ribosome.

Taking into account those results, it is logical to predict that the fourth connector that directly interacts with many of the seed proteins, MHR1, must also have a significant role in mitochondrial ribosome function. However, the SGD database does not report any relationship of this protein with ribosomal function in *S. cerevisiae*. It only includes that it is involved in homologous recombination in mitochondria, transcriptional regulation in the nucleus and recombination-dependent mtDNA partitioning. This summary turns out to be incomplete. In fact, the protein orthologous to MHR1 in another ascomycete fungus, *Neurospora crassa*, was found to be part of the large subunit of the mitochondrial ribosome (Gan et al., 2006). This finding led to a reexamination of whether MHR1 could be found in the ribosome in *Saccharomyces* and, as a result, I was determined that MHR1 is an integral part of the large subunit (Gan et al., 2006). Thus, we can conclude from these four examples that detection by basic network analyses of connector proteins with large numbers of

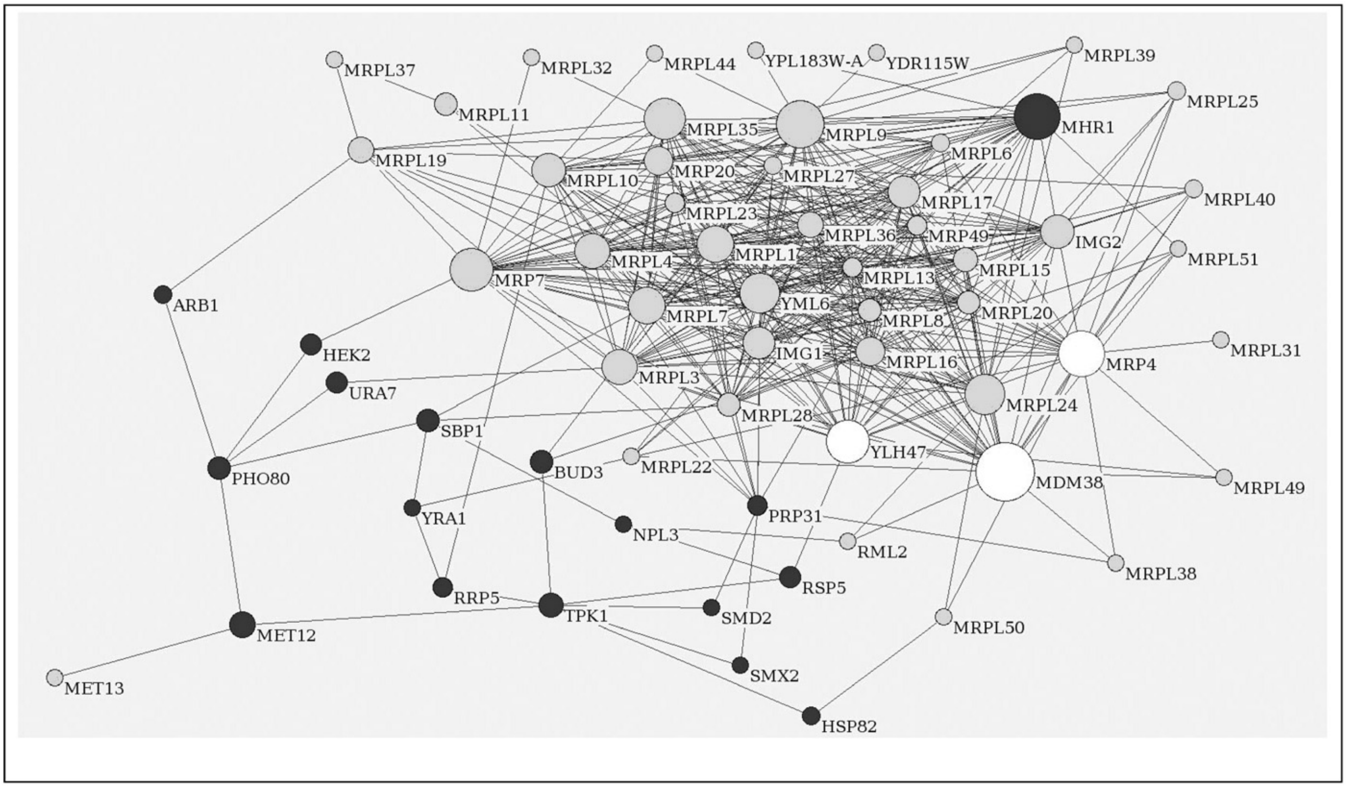

Fig. 4. Tied basic network for the *mitochondrial large ribosomal subunit* GO term. Gray dots: seeds, i.e. proteins included in the term. Black and white: connectors. White nodes are mitochondrial proteins known to associate to the ribosome (YLK47, MDM38) or part of the small subunit of the mitochondrial ribosome (MRP4). Black nodes had, according to SGD, no known relationship with mitochondrial ribosome function. All but one of the black nodes are included to connect MET13 with the rest of proteins, suggesting that MET13 should not be included in the GO term (see text). The size of the circles is proportional to the number of minimal paths among seeds which include the nodes. However, in the case of seeds, minimal paths that start or end in a seed have not been counted. The figure was drawn with Pajek (Batagelj and Mrvar, 2003).

Table 1
Characterization of the basic networks obtained when the proteins annotated to several GO terms are used as seeds and comparison with random basic networks obtained starting from the same number of random seeds.

GO term	GO domain	No. of seeds	No. of connectors in the basic networks tied/pruned	No. of connectors in random basic networks tied/pruned	*Z*-scores tied/ pruned
Mitochondrial large ribosomal subunit	Cellular component	43	22±1 (20)/19 (19)	207±27 (163)/206±27 (163)	6.85/6.92
Mitochondrial small ribosomal subunit	Cellular component	33	11±1 (10)/10 (10)	150±23 (109)/149±22 (109)	6.04/6.32
Mitochondrial ribosome	Cellular component	76	36±2 (34)/34±1 (34)	382±37 (298)/381±37 (298)	9.35/9.38
Spliceosome	Cellular component	79	19±2 (17)/19±1 (17)	402±37 (339)/401±36 (339)	10.35/10.61
Phosphoprotein phosphatase activity	Molecular function	46	151±2 (149)/151±2 (149)	220±27 (171)/219±26 (171)	**2.56/2.58**
Protein kinase activity	Molecular function	127	381±1 (380)/381±1 (380)	637±36 (551)/637±36 (551)	7.11/7.11
Protein folding	Biological process	72	224±2 (221)/224±2 (221)	362±32 (301)/361±31 (301)	4.31/4.42
Sexual reproduction	Biological process	110	403±3 (399)/403±3 (399)	560±33 (499)/560±33 (499)*	4.76/4.76
Sporulation	Biological process	110	525±3 (522)/525±3 (522)		**1.06/1.06**

In all cases, 1500 permutations of the seeds were performed. Values are expressed as mean ± standard deviation and, in parenthesis, the minimum number (i.e. the optimal solution for the essential network characterized after obtaining all replicates). *Z*-scores (calculated as the absolute value of the difference between the average sizes of the random and experimental networks divided by the standard deviation of the random networks) are also indicated. The asterisk refers to the fact that, the number of seeds was identical in the two last GO terms, a single characterization of random basic networks, with number of seeds equal to 110, was performed. Thus, the *Z*-scores obtained for the GO terms *sexual reproduction* and *sporulation* are based on the same random basic networks. Values in bold are not significant, after Bonferroni′s correction.

links is a good evidence for strong functional relationships of those connectors with the seed proteins.

Very interestingly, the other 16 connectors are both unrelated to ribosome function (according to SGD) and are only linked to at most a few of the large subunit proteins (Fig. 4). These results can be explained noticing that there is an outlier among the seeds: the protein MET13, which is far apart from the rest of the proteins in this GO term (Fig. 4). The fact that MET13 is not closely linked to the rest of seeds explains the need to include those 16 additional connectors. MET13 is a methylenetetrahydrofolate reductase involved in methionine biosynthesis and therefore its relationship with the mitochondrial ribosome is unclear. According to SGD, MET13 was purified together with units of the mitochondrial large ribosomal subunit by Kitakawa et al. (1997). However, a direct interaction with any of those units has not been yet described. Its purification may thus have been artifactual.

To generalize the results obtained with the large subunit of the mitochondrial ribosome, we performed related searches with other sets of proteins, defined according to diverse GO terms. Results are summarized in Table 1. Significantly, in all cases that we studied, the tied and pruned basic networks were very similar (Table 1). This is due to the fact that most units in the tied basic networks were basic units and the number of ties was therefore very low. If we now consider each GO term in detail, we can easily conclude that they can be divided into two classes. Some of them, especially those that are included in the *cellular component* GO domain (as the *mitochondrial large ribosomal subunit* term just described), are known to have a strong structural basis, that is, many of the proteins in a given term interact. They all generated basic networks which are much smaller than those generated by random seeds (Table 1; ribosome- and spliceosome-related terms). On the contrary, other GO terms, especially some of those included in the *molecular function* and *biological process* GO domains, generate basic networks that are either larger or not much smaller than those obtained starting with random seeds (see data in brackets in Table 1). In two cases, the sizes of the random basic networks were statistically not significantly different from those of the GO-based basic networks (see *Z*-scores in Table 1). These results demonstrate that these GO terms are not characterized by including gene products which are closely linked in the protein-protein interaction network.

5. Discussion

We think that our definition of basic network is conceptually significant. First, it is an intuitively simple concept. Second, it provides a natural way to obtain groups of nodes related to a set of predefined seeds that escapes from any, more or less controversial, module definition. Finally, the results described above show that it is useful in very different contexts. On one hand, basic networks can be used to determine whether units are related or not. We have shown that this can be done by comparing the minimal graphs obtained with those units as seeds with the minimal graphs obtained with the same number of seeds, but randomly taken from the graph. On the other hand, basic networks point out proteins which are very closely linked to the seeds, for which a function related to the function of the seed proteins is likely (as we have discussed above for the connectors MHR1, MRP4, YLH47 and MDM38). Alternatively, by defining groups of nodes that are closely linked to the selected seeds, the determination of basic networks may contribute to understand what those seeds have in common. A final aspect is that basic networks may provide clues about potentially false connections in the graphs. When adding a single unit to the set of seeds involves a substantial increase in the number of connectors, we may suspect that the unit is unrelated to the rest of seeds. This can be quite easily observed by depicting the basic network: the offending unit stands out as very distant from the rest (e.g. the case of MET13 in Fig. 4).

Our solution to the problem of obtaining basic networks, the strategy described above is, given the impossibility of analyzing all possible combinations of units in large graphs, a heuristic search based on conserving the nodes with the highest number of paths traveling among seeds. As any other heuristic search, it does not guarantee finding the true basic network. However, the cases examined suggest that, in general, the solutions obtained will be very similar if the seeds are close in the graph (see the very low standard deviations in Table 1). In any case, the user may perform a large number of trials, in order to obtain a progressively improving approximation to the basic network. Saturation, i.e. lack of further minimization of the size of the graphs after a large number of trials, may indicate that the basic network has been already obtained. Further exploration of complex graphs may determine the usefulness and limitations of this and other potentially competing heuristic searches for basic network characterization.

Acknowledgments

I.M. developed the concept of basic networks and devised the heuristic strategy to characterize them and wrote the manuscript. S.H. wrote the Netbasic program to implement that strategy and performed the analyses. This project was supported by grant 200720I021 (Proyectos intramurales especiales, CSIC, Spain) and project BIO2008-05067 (Programa Nacional de Biotecnología; Ministerio de Ciencia e Innovación. Spain) awarded to I.M. We would like to thank Antonio Marco for his help in the development of the examples shown in this study.

References

Aittokallio, T., Schwikowski, B., 2006. Graph-based methods for analysing networks in cell biology. Briefings in Bioinformatics 7, 243–255.

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M., Sherlock, G., 2000. Gene ontology: tool for the unification of biology. Nat. Genet. 25, 25–29.

Asthana, S., King, O.D., Gibbons, F.D., Roth, F.P., 2004. Predicting protein complex membership using probabilistic network reliability. Genome Res. 14, 1170–1175.

Arnau, V., Mars, S., Marín, I., 2005. Iterative cluster analysis of protein interaction data. Bioinformatics 21, 364–378.

Bader, J.S., 2003. Greedily building protein networks with confidence. Bioinformatics 19, 1869–1874.

Batagelj, V., Mrvar, A., 2003. Pajek-analysis and visualization of large networks. In: Jünger, M., Mutzel, P. (Eds.), Graph Drawing Software. Springer, Berlin.

Chartrand, G., Zhang, P., 2004. Distance in graphs. In: Gross, J.L., Yellen, J. (Eds.), Handbook of Graph Theory. CRC Press, Boca Raton, FL.

Del Rio, G., Bartley, T.F., del Rio, H., Rao, R., Jin, K.L., Greenberg, D.A., Eshoo, M., Bredesen, D.E., 2001. Mining DNA microarray data using a novel approach based on graph theory. FEBS Lett. 509, 230–234.

Frazier, A.E., Taylor, R.D., Mick, D.U., Warscheid, B., Stoepel, N., Meyer, H.E., Ryan, M.T., Guiard, B., Rehling, P., 2006. Mdm38 interacts with ribosomes and is a component of the mitochondrial protein export machinery. J. Cell Biol. 172, 553–564.

Gan, X., Kitakawa, M., Yoshino, K., Oshiro, N., Yonezawa, K., Isono, K., 2002. Tag-mediated isolation of yeast mitochondrial ribosome and mass spectrometric identification of its new components. Eur. J. Biochem. 269, 5203–5214.

Gan, X., Arita, K., Isono, S., Kitakawa, M., Yoshino, K., Yonezawa, K., Kato, A., Inoue, H., Isono, K., 2006. Identification and comparative analysis of the large subunit mitochondrial ribosomal proteins of *Neurospora crassa*. FEMS Microbiol. Lett. 254, 157–164.

Hashimoto, R.F., Kim, S., Shmulevich, I., Zhang, W., Bittner, M.L., Dougherty, E.R., 2004. Growing genetic regulatory networks from seed genes. Bioinformatics 20, 1241–1247.

Kitakawa, M., Graack, H.R., Grohmann, L., Goldschmidt-Reisin, S., Herfurth, E., Wittmann-Liebold, B., Nishimura, T., Isono, K., 1997. Identification and characterization of the genes for mitochondrial ribosomal proteins of *Saccharomyces cerevisiae*. Eur. J. Biochem. 245, 449–456.

Krauthammer, M., Kaufmann, C.A., Gilliam, T.C., Rzhetsky, A., 2004. Molecular triangulation: bridging linkage and molecular-network information for identifying candidate genes in Alzheimer's disease. Proc. Natl. Acad. Sci. USA. 101, 15148–15153.

Li, A., Horvath, S., 2007. Network neighborhood analysis with the multi-node topological overlap measure. Bioinformatics 23, 222–231.

Lubovac, Z., Gamalielsson, J., Olsson, B., 2006. Combining functional and topological properties to identify core modules in protein interaction networks. Proteins 64, 948–959.

Lucas, J.I., Arnau, V., Marín, I., 2006. Comparative genomics and protein domain graph analyses link ubiquitination and RNA metabolism. J. Mol. Biol. 357, 9–17.

Luo, F., Yang, Y., Chen, C.F., Chang, R., Zhou, J., Scheuermann, R.H., 2007. Modular organization of protein interaction networks. Bioinformatics 23, 207–214.

Scott, M.S., Perkins, T., Bunnell, S., Pepin, F., Thomas, D.Y., Hallett, M., 2005. Identifying regulatory subnetworks for a set of genes. Mol. Cell. Proteomics 4, 683–692.

Sharan, R., Ulitsky, I., Shamir, R., 2007. Network-based prediction of protein function. Mol. Syst. Biol. 3, 88.

Stark, C., Breitkreutz, B.J., Reguly, T., Boucher, L., Breitkreutz, A., Tyers, M., 2006. BioGRID: a general repository for interaction datasets. Nucleic Acids Res. 34, D535–D539.