Property-driven statistics of biological networks

Vincent Schächter[1], Pierre-Yves Bourguignon[1], Vincent Danos[2], Serge Smitdas[1], and François Képes[3]

[1] Genoscope
[2] CNRS & Université Paris VII
[3] CNRS

Abstract. An analysis of biological networks based on property-preserving randomizations is presented. Unlike any other method proposed yet, ours assumes no preliminary notion of random network and is not limited to the detection of local properties.
The feasibility and relevance of the method are demonstrated in the yeast protein-protein interaction and transcriptional regulation interaction network. A measure of modularity is proposed revealing a significant deviation between the real network and the randomized versions.

1 Introduction

The availability of genome-scale metabolic, protein-protein interaction and regulatory networks [23,8,4,6,19] —following closely the availability of large graphs derived from the Internet hardware and software network structure, from social or collaborative relationships— has spurred considerable interest in the empirical study of the statistical properties of these 'real-world' networks. As part of a wider effort to reverse-engineer biological networks, recent studies have focused on identifying *salient* graph properties that can be interpreted as 'traces' of underlying biological mechanisms, shedding light either on their dynamics [21,11,7,26] (*i.e.*, how the connectivity structure of the biological process reflects its dynamics), on their evolution [10,28,25] (*i.e.*, likely scenarios for the evolution of a network exhibiting the observed property or properties), or both [9,13,14]. The statistical graph properties that have been studied in this context include the distribution of vertex degrees [10,9], the distribution of the clustering coefficient and other notions of density [15,16,17,20,5], the distribution of vertex-vertex distances [20], and more recently the distribution of network motifs occurrences [14].

Identification of a salient property in an empirical graph —for example the fact that the graph exhibits a unexpectedly skewed vertex degree distribution— requires a prior notion of the distribution of that property in a class of graphs relatively to which saliency is determined. The approach chosen by most authors so far has been to use a *random graph model*, typically given by a probabilistic graph generation algorithm that constructs graphs by local addition of vertices and edges [18,1,22]. For the simplest random graph models, such as the classical

Erdös-Rényi model (where each pair of vertices is connected with constant probability p, [3]), analytical derivations of the simplest of the above graph properties are known [18,1].

In the general case, however, analytical derivation is beyond the reach of current mathematical knowledge and one has to retort to numerical simulation. The random graph model is used to generate a sample of the corresponding class of graphs and the distribution of the graph property of interest is evaluated on that sample, providing a standard against which the bias of the studied graph can be measured [21,13,27]. Some of these random graph models are not only designed to emulate one or more network properties —such as the small-world property (the average distance between nodes is logarithmic in the total number of nodes), or the power-law nature of vertex degree distribution (the probability of having n immediate neighbours varies as $1/n^{\gamma}$ with $2 < \gamma \leq 3$)— but also understood as *constructive explanations* for the evolution of the class of biological networks under scrutiny [10,9,25,20]. Again, perhaps because of the local nature of the random graph generation process, it is mostly simple *local* network properties that have been successfully reproduced in that fashion.

Our general approach, in essence, is to reverse the process. Rather than contrast the network of interest against an independent random graph model, one generates the model from the original graph itself. Specifically, one selects a property of the graph and generates a sample of randomized graphs within the class of graphs sharing this property. This sample is then compared with the original graph. Whichever property is lost in the shuffling process can then be construed as a system property that has to be understood and can serve as a means of statistically validating and classifying other networks, perhaps by homology considerations. That said, there are obvious constraints bearing on the choice of properties both for shuffling the original graph and for comparing it with the obtained sample. First, the associated randomization has to be feasible. Second, new properties, one could not obtain in the traditional bottom-up approach have to be made accessible by the new method, else why bother with a new method ?

The specific contribution of this article is to rigorously define a category of global properties where the shuffling procedure is easily implemented and does lead to something new, in that properties may describe at some level of detail how a biological attribute of vertices or edges unfolds on the graph structure, and may include detailed topological structures of subgraphs. Clearly such investigations are not within reach of the usual methods. To be a bit more concrete, suppose a heterogeneous network, *i.e.*, with several types of edges (relationships) on the same set of vertices (biological entities, *e.g.*, proteins), is given. One can then demand that the shuffling preserve homogeneous component subnetworks —each corresponding to one type of edge— and allow them to be glued back randomly. Various criteria can be used to measure the relative strength of the chunk cooperativity in the true graph compared to a randomized one.

To illustrate the method, we investigated the network obtained by combining protein-protein interactions (PPI) and direct transcriptional influences (protein-

DNA interactions, thereafter abbreviated as TRI) in yeast, using the protein-protein distance distribution as a means of measuring cooperativity. One would expect, and the authors did, to observe a significant statistical deviation, showing more cooperation between the PPI and the TRI chunks in the true graph. One would think that the way both subnetworks are glued in the true network makes it an even smaller or more compact world that the randomized variants. It turns out it is the exact opposite. Distances in the real network are larger in average. A reasonable guess is that the network seen as an information processor has more definite functions (in other words is more expressive) if it is modular, and being modular means probably losing on the side of having short paths. However, we don't have at the moment any solid explanation for this phenomenon but it is certainly a good point for the method itself to generate new observations from data that demand a biological explanation, and most likely one of an unusual kind.

The rest of this article is organized as follows. Section 2 defines more precisely what we mean by graph property, invariants and shuffles. Section 3 describes the application to the combined PPI—TRI network in yeast. Finally, we list several directions for future research on the identification of relevant properties in biological networks.

2 Properties, invariants and shuffles

The notion of graph property can be used within a variety of contexts and thrives without the umbrella of an overarching definition. Existing work on the analysis of biological networks and other types of large networks modeling 'real-world' phenomena, however, is actually very focused on a small set of graph properties, such as the distribution of vertices degrees, the clustering coefficient, or more recently the distribution of small 'network motifs' within the graph. One common point of these properties is that they describe a statistical distribution of *local* characteristics of the graph structure.

In the case of biological networks, another layer of information is often associated with the graph: attributes of vertices and edges bearing biological information. These are also local characteristics of the graph in the sense that they are associated with individual graph elements, and their distribution on the graph can also be studied.

Here, we will take the view that an attribute of graph elements (vertices or edges) is a mapping from these elements to a discrete set of values. While this view extends to continuous sets of values, for instance via discretization, the discrete nature of a property is precisely what allows its study in the present context.

An attribute of vertices or edges may represent a biological piece of information (*e.g.*, the functional class associated to a protein), but it may also reflect some local property of the graph structure (*e.g.*, the degree of a vertex, or inclusion of a vertex within a cluster, where the clustering was performed by using some distance on the graph). The point is that we do not focus here on how

the property is obtained. Rather, we focus on how the *distribution of the attribute values on the graph structure* can be imposed as an *invariant* on a class of graphs, in order to study how it influences the distribution of other observable properties.

Several types of such invariants can be envisioned, based either on vertices or edges attributes:

1. the complete graph structure of one or more subgraphs of the initial graph, subgraphs which may be defined on the basis of edge or vertices attributes.
2. the partition of vertices (or edges) into a set of equivalence classes defined on the basis of a given attribute. This can also be seen as the distribution of vertices (or edges) into clusters, and described formally as a map from the initial graph into an abstract graph of clusters, with no edges; the invariant to be preserved is then the morphism onto that specific abstract graph.
3. the structure of the partition into equivalence classes along with constraints on the edges between these classes. This corresponds to a morphism into an abstract graph of *interconnected* clusters.

A *shuffle* relative to an invariant is then defined as a special case of permutation on the graph that preserves that invariant.

For the rest of this paper, we will focus on the first type of invariant: preservation of the complete connectivity structure of subgraphs defined on the basis of an edge attribute. In order to give a rigorous definition of this type of invariant, let us define a notion of heterogeneous graph as a graph obtained by combining several graphs by 'glueing them together' on the same set of vertices.

2.1 Graphs and morphisms

Graphs can be defined in many ways depending on whether loops, multi edges and directed edges are allowed or not; we take here the view that a graph is directed, include loops and multi-edges (two vertices may be connected more than once).

Definition 1 *A* graph *consists of:*

- *two finite sets E, V of* edges *and* vertices *(or nodes),*
- *and two maps t, s, called* target *and* source*, mapping E to V.*

Equally important here, is the notion of a *morphism* between two graphs.

Definition 2 *Suppose two graphs (V, E, t, s) and (V', E', t', s') given, then a map f from V to V' and E to E' is said to be a morphism if for all $e \in E$:*

$$f(s(e)) = s'(f(e)) \text{ and } f(t(e)) = t'(f(e))$$

Morphisms compose, that is given $f_1 : G_1 \to G_2$ and $f_2 : G_2 \to G_3$, one can form their composite $f_2 \circ f_1 : G_1 \to G_3$. This composition endows the set of graphs with the structure of a *category* [2].

An invertible morphism $f : G \to H$ is called an *isomorphism*, and an *automorphism* if $G = H$. Two graphs related by an isomorphism share all the same graph-theoretical properties, average degree, number of cycles, etc. Let us remember this for later use.

2.2 Glueing

Definition 3 *Given a set V, a tuple of graphs $G_1 = (V_1, E_1, t_1, s_1)$, ..., $G_n = (V_n, E_n, t_n, s_n)$, and a tuple of maps $p_1 : V_1 \to V$, ..., $p_n : V_n \to V$, one may define a new graph $G = (V, E, t, s)$ as follows:*

- *$E = \sum_{1 \leq i \leq n} E_i$ (disjoint sum);*
- *for all i and $e_i \in E_i$, $t(e_i) = p_i(t_i(e_i))$ and $s(e_i) = p_i(s_i(e_i))$.*

This graph G is said to be obtained by *glueing* the G_i along the maps p_i, and is denoted by $[(G_1, p_1), \ldots, (G_n, p_n)]$. The G_is will be refered to as the *components* of the glueing and the maps p_i as the *glueing maps*.

The glueing maps represent instructions explaining how the vertices in $\sum_i V_i$ should be glued together. By construction, they extend to graph morphisms from G_i to G by setting $p_i(e_i) = e_i$ (indeed a morphism, since for each e_i, $p_i(t_i(e_i)) = t(e_i) = t(p_i(e_i))$). Those morphisms are injective on edges, and in the application will also be injective on vertices.

An example of glueing For instance, the following component graphs:

$$G_1 = v \xrightarrow{a} u \quad G_2 = v \xleftarrow{b} u \quad G_3 = u \circlearrowright c$$

together with the glueing maps defined by $p_1(u) = p_2(u) = p_3(u) = u$ and $p_1(v) = p_2(v) = v$, obtain the glueing:

$$G = \; c \circlearrowright u \underset{b}{\overset{a}{\rightleftarrows}} v$$

Incidentally, one sees that glueing may result in multiple edges connecting the same two vertices, even though no components has multi edges.

2.3 Shuffling

When a graph is obtained as a glueing, one can define transformations that preserve each of the components.

Definition 4 *Given a graph $G = [(G_1, p_1), \ldots, (G_n, p_n)]$ and a tuple of bijections $\alpha_1 : V_1 \to V_1$, ..., $\alpha_n : V_n \to V_n$, one defines the* shuffle *of G as:*

$$G^{\alpha_1, \ldots, \alpha_n} = [(G_1, p_1 \circ \alpha_1), \ldots, (G_n, p_n \circ \alpha_n)] \tag{1}$$

In words, $G^{\alpha_1,\ldots,\alpha_n}$ is obtained by glueing back together the component graphs after applying to each an automorphism. Of course, the result depends both on the glueing and the choice made of the automorphisms.

Although a shuffle preserves the component graphs, by modifying the way in which these are interconnected, it may result in an overall graph which is not isomorphic to the original one.

An example of shuffling We can see this by going back to the example and choosing the shuffle defined by $\alpha_1(u) = v$, $\alpha_1(v) = u$, and both α_2 and α_3 are the identity, then:

$$G^{\alpha_1,\alpha_2,\alpha_3} = \quad c \circlearrowleft \; u \underset{b}{\overset{a}{\rightleftarrows}} v$$

which indeed is not isomorphic to G.

Glueing and shuffling can be presented in a more abstract and algebraic way by using *amalgamated sums* (also known as *pushouts*) in the category of graphs. While not directly relevant for our development, this more abstract view may prove useful with richer notion of graphs. It is a warrant of the robustness of the approach and will be developed in future work.

3 Application to the combined PPI TRI network in yeast

With our definitions in place, we can now illustrate the approach on a biologically meaningful example using a graph obtained by glueing two components.

It is known that regulatory influences, including those inferred from expression data analysis or genetic experiments, are implemented by the cell through a combination of direct regulatory interactions and protein-protein interactions, which propagate signals and modulate the activity level of transcription factors. The detailed principles underlying that implementation are not well understood, but one guiding property is the fact that protein interaction and transcriptional regulation events take place in the regulatory network at different time-scales.

In order to clarify the interplay between these two types of interactions, we have combined protein-protein (PPI) and protein-DNA (TRI, for 'transcriptional regulation interaction') interaction data coming from various sources into a heterogeneous network by glueing together these two networks on the underlying set of yeast proteins.

The data from which the composite network was built includes: 1440 protein complexes identified from the literature, through HMS-PCI or TAP [4,6], 8531 physical interactions generated using high-throughput Y2H assays [24], and 7455 direct regulatory interactions compiled from literature and from ChIP-Chip experiments [5,12], connecting a total of 6541 yeast proteins. A subnetwork of high-reliability interactions was defined, using a reliability scale founded on assay type, and quality indicators provided by the authors, if any. The PPI network is

built by connecting two proteins, in both directions, whenever there is a protein-protein or a complex interaction between the two corresponding proteins. In the case of the TRI network, an edge connects a regulator protein with its regulatee.

To assess the role of PPI in the TRI network, one needs a measure of the cooperativity between the two associated subgraphs. The property we choose to observe in the combined network, and its randomized variants, is the distribution of the distance between proteins, *i.e.*, the length of the shortest directed path connecting any given pair of proteins. As explained in the preceding section, randomized variants are obtained by shuffling the TRI network (in this case it is enough to shuffle one of the two subgraphs to get all possible results up to isomorphism). We then estimate the distance distribution both for the actual and the shuffled networks.

At this point, we have to compromise to make the computation feasible. Indeed, we only consider a small sample (700) of shuffled variants. This is a necessary evil, since an enumeration of the full set of shuffled networks is unfeasible, and on the other hand there is clearly no analytical expression for the distribution of the shortest path lengths in the shuffled graphs. However, and for the statistical observations we are interested in here, the empirical standard deviation strongly indicates that the PPI- and TRI-subgraphs are glued together so that the distribution of distances is biased in a significant way.

The first basic statistics with respect to which we observe a significant deviation between the real graph and the population of shuffled graphs is the *average distance* computed over the set of all connected pairs. Figure 1 reveals the value of 3.95 computed over the real network is significantly larger than those obtained from shuffled networks. Assuming that the average distance between connected pairs in the population of shuffled graphs is a gaussian distribution, the p-value of this deviation is 8×10^{-134} (more details are given in the appendix).

Looking only at the mean distance may seem to suggest that shuffles contract the graph. However the situation turns out to be more more subtle when one refines this first observation and compares the histogram of distances in the real and shuffled networks (see the appendix for details about algorithms and computational costs). Figure 2 shows the distance histogram of the real graph and summarises the distribution of the shuffled histograms. This summary is obtained by indicating with central horizontal bars the average over the shuffled networks, together with lower and upper vertical bars indicating the 1^{st} and 99^{th} percentiles (meaning for any n less than 2% of the shuffled graphs have a number of pairs at distance n which is not whithin these vertical bars). The last bar on the right indicates the amount of disconnected node pairs (hence at distance ∞). Clearly the real graph is below the average histogram for $n = 3, \infty$ and above for $n = 4, 5, 6$.

Let us comment first on the largest difference obtained for $n = \infty$. One sees that there is in average of 21.2% disconnected pairs in the shuffled graphs whereas only 2.7% are disconnected in the real graph, which is therefore outstandingly more connected. These 18.5% additional disconnected pairs are accounting for part for the deficit over the real pairs observed at distance $n = 4, 5, 6$ (which

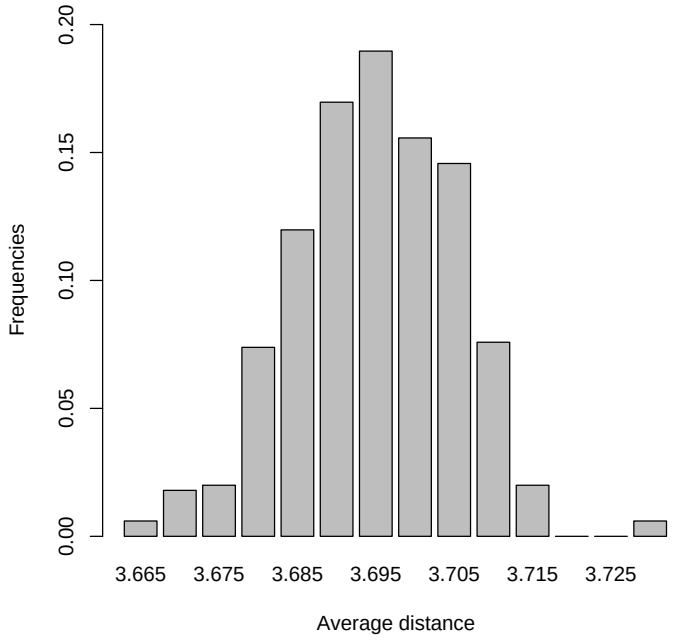

Fig. 1. Distribution of the average shortest path lengths in the shuffled networks. The average in the real graph is 3.94. Averaging is done on the set of connected pairs.

sums up to 20.9% if one takes also into account $n = 7$), while the remaining 2.4% missing pairs account for the surplus over the real pairs observed at distance $n = 3$. Shuffling the graph contracts it at short distances and expands at longer ones.

The statistical picture we have so far is consistent with an idealisation of the real graph as a series of PPI modules which are TRI connected in a fragile and precise way —an idealized view which would explain both why shuffles are less connected and more compact at short distances. To examine whether the statistical evidence collated at this stage might indeed be construed as a measure of a greater modularity of the real network, we have counted its number of connected components, and compared it with the number of connected components in the shuffled networks. Indeed, as Figure 3 shows, there is an average of 500 such components in the shuffled networks whereas the number of connected components in the real network is 37. It seems that when shuffled, the TRI does not link properly the PPI components anymore, and breaks the long paths running from a component to another. This is further confirmed by the distribution of

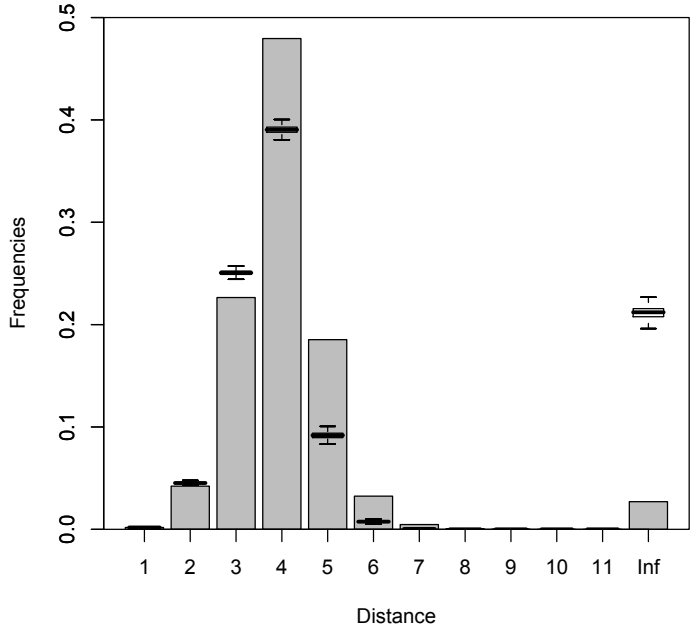

Fig. 2. Distance histograms in the real graph and sample of shuffled graphs. Rightmost column corresponds to disconnected pairs.

distances in the PPI and TRI subgraphs (Figure 4) which shows that the PPI network is made mostly of tightly connected components with diameter ≤ 4, whereas the TRI network exhibits much longer distances.

One could summarise the above observations by saying that the numerous small PPI connected components are carefully interfaced with the TRI subgraph in the real network so as to obtain a highly connected combined graph. To further examine the properties of the real graph, one could think of observing refined properties such as the histogram of length of shortest paths going through both the TRI and PPI-subgraphs, with a view to eliminate the independent contributions of the subgraphs and better measure their cooperativity.

4 Conclusion and future work

We have introduced a methodology for the identification of potentially meaningful properties in biological networks, based on randomizations that preserve other classes of properties. We have also provided the definition of a specific category of such invariants, reflecting the decomposition of a network into sev-

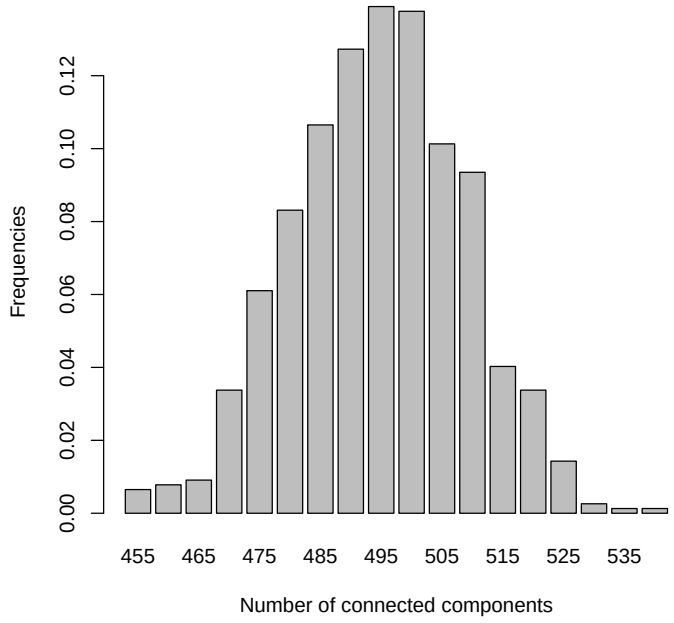

Fig. 3. Histogram of the number of connected components in the shuffled networks. There are 37 connected components in the real network.

eral subnetworks corresponding to different types of edges, and a procedure to generate the corresponding class of networks with uniform probability. Finally, we have given an illustration of the method on a network composed of PPI and TRI interactions, where we show that the actual network exhibits a markedly biased distribution of protein-protein distances relatively to the class of networks obtained by shuffling the TRI network in all possible ways.

The extant 'bottom-up' approaches typically proceed by first selecting a statistical property thought to be characteristic of the class of networks against which the network of interest is to be contrasted. That class of networks is then obtained through a random generation procedure that provides no guarantee of capturing precisely that property (*i.e.*, there may be 'false positive' or 'false negative' networks), and little control over the probability of occurrence of networks in that class.

In contrast, our 'top-down' approach allows for both a rigorous definition of the invariant property and of the class of networks which obey it. This methodology can be driven by any biological property of interest that one may attribute to interactions (edges) or species (nodes). As we have shown in the preceding

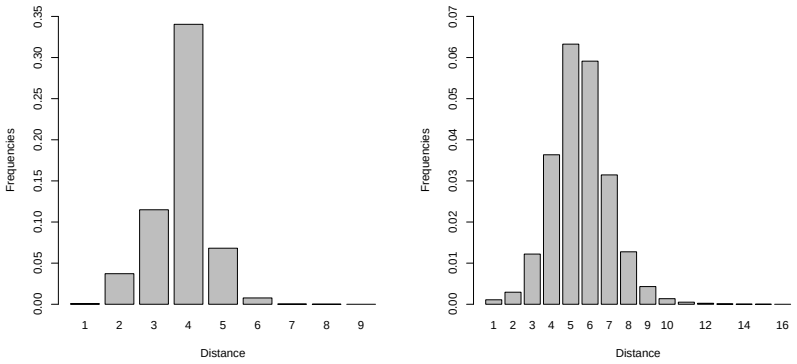

Fig. 4. Histogram of the distance between pairs of nodes in the PPI network (left) and in the TRI network (right). Frequencies are relative to the overall number of pairs. 52% of the paires are connected in the PPI network, while only 21.5% are connected in the TRI network. The average distance, computed over the set of connected pairs, is 3.8 in the PPI and 5.7 in the TRI.

section, interactions can be partitioned in the PPI and TRI class. Different kinds of biological information could also be put to use. For instance, species could be sorted according to either clustering information, localisation within the various cell organelles and membranes, biochemical specificities (*e.g.*, length of the amino-acid sequence, hydrophoby), or combinations thereof. Then, other categories of top-down invariants, such as the projection of a network onto a given network of abstract clusters could be explored. In addition, there is also room for using other observable properties. In fact, we believe that the approach sketched in this paper is quite general.

Another line of investigation, perhaps more theoretical but potentially very fruitful in terms of applied insights, is the design of a solid unified theoretical framework for the definition of top-down invariant properties of networks, using category theory. The 'glueing' invariant presented enjoys a natural definition in this framework, and it seems likely that other properties that are related to the preservation of (partial) connectivity structures will, too.

Finally, a longer-term promise of the above extensions is to provide the foundations of an iterative search process for relevant properties in biological networks: at stage n, a class of networks is given by a set of invariants (biological constraints/hypotheses), and one searches for a property that discriminates the network of interest against the current class. Such a property is then considered as a candidate for 'biologically meaningful' status, and subjected to finer scrutiny, theoretical and perhaps experimental. If that status is confirmed, then the property is added to the set of invariants, thus defining a more restrictive class of biological networks, against which the same or another network can be pitted at the next iteration. In other words, the principle would be to search

for properties that are *not yet explained* by the existing store of invariants —a form of conditional probability assessment— and to elevate some of them to explanation/invariant status.

References

1. William Aiello, Fan Chung, and Linyuan Lu. A random graph model for massive graphs. In *32nd Annual ACM Symposium on Theory of Computing*, pages 171–180, 2000.
2. Francis Borceux. *Handbook of Categorical Algebra*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, Cambridge, 1994.
3. P. Erds and A. Rnyi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci.*, 5:1761, 1960.
4. AC Gavin, M Bosche, R Krause, P Grandi, M Marzioch, A Bauer, J Schultz, JM Rick, AM Michon, CM Cruciat, M Remor, C Hofert, M Schelder, M Brajenovic, H Ruffner, A Merino, K Klein, M Hudak, D Dickson, T Rudi, V Gnau, A Bauch, S Bastuck, Huhse, C Leutwein, MA Heurtier, RR Copley, A Edelmann, E Querfurth, V Rybin, G Drewes, M Raida, T Bouwmeester, P Bork, B Seraphin, B Kuster, G Neubauer, and G. Superti-Furga. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868)(Jan 10):141–7., 2002.
5. N. Guelzim, S. Bottani, P. Bourgine, and F. Kepes. Topological and causal structure of the yeast transcriptional regulatory network. *Nat Genet*, 31(1):60–3, 2002.
6. Y Ho, A Gruhler, A Heilbut, GD Bader, L Moore, SL Adams, A Millar, P Taylor, K Bennett, K Boutilier, L Yang, C Wolting, I Donaldson, S Schandorff, J Shewnarane, M Vo, J Taggart, M Goudreault, B Muskat, C Alfarano, D Dewar, Z Lin, K Michalickova, AR Willems, H Sassi, PA Nielsen, KJ Rasmussen, JR Andersen, LE Johansen, LH Hansen, H Jespersen, A Podtelejnikov, E Nielsen, J Crawford, V Poulsen, BD Sorensen, J Matthiesen, RC Hendrickson, F Gleeson, T Pawson, MF Moran, D Durocher, M Mann, CW Hogue, D Figeys, and M Tyers. Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry. *Nature*, 415(6868)(Jan 10):180–3, 2002.
7. J. Ihmels, G. Friedlander, S. Bergmann, O. Sarig, Y. Ziv, and N. Barkai. Revealing modular organization in the yeast transcriptional network. *Nat Genet*, 31(4):370–7, 2002.
8. T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc Natl Acad Sci U S A*, 98(8):4569–74, 2001.
9. H. Jeong, S. P. Mason, A. L. Barabasi, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–2, 2001.
10. H. Jeong, B. Tombor, R. Albert, Z.N. Oltvai, and A.-L. Barabasi. The large-scale organization of metabolic networks. *Nature*, 407:651–654, 2000.
11. N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, 2004.
12. T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannett, C. T. Harbison, C. M. Thompson, I. Simon, J. Zeitlinger, E. G. Jennings, H. L. Murray, D. B. Gordon, B. Ren, J. J. Wyrick, J. B. Tagne, T. L. Volkert, E. Fraenkel, D. K. Gifford, and R. A. Young. Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 298(5594):799–804, 2002.

13. R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer, and U. Alon. Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–42, 2004.
14. R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–7, 2002.
15. M. E. Newman. Assortative mixing in networks. *Phys Rev Lett*, 89(20):208701, 2002.
16. M. E. Newman. Properties of highly clustered networks. *Phys Rev E Stat Nonlin Soft Matter Phys*, 68(2 Pt 2):026121, 2003.
17. M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys Rev E Stat Nonlin Soft Matter Phys*, 69(2 Pt 2):026113, 2004.
18. M. E. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys Rev E Stat Nonlin Soft Matter Phys*, 64(2 Pt 2):026118, 2001.
19. N. D. Price, J. A. Papin, C. H. Schilling, and B. O. Palsson. Genome-scale microbial in silico models: the constraints-based approach. *Trends Biotechnol*, 21(4):162–9, 2003.
20. E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A. L. Barabasi. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586):1551–5, 2002.
21. S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon. Network motifs in the transcriptional regulation network of escherichia coli. *Nat Genet*, 31(1):64–8, 2002.
22. S. H. Strogatz. Exploring complex networks. *Nature*, 410(6825):268–76, 2001.
23. P. Uetz, L. Giot, G. Cagney, T. A. Mansfield, R. S. Judson, J. R. Knight, D. Lockshon, V. Narayan, M. Srinivasan, P. Pochart, A. Qureshi-Emili, Y. Li, B. Godwin, D. Conover, T. Kalbfleisch, G. Vijayadamodar, M. Yang, M. Johnston, S. Fields, and J. M. Rothberg. A comprehensive analysis of protein-protein interactions in saccharomyces cerevisiae. *Nature*, 403(6770):623–7, 2000.
24. C. von Mering, R. Krause, B. Snel, M. Cornell, S. G. Oliver, S. Fields, and P. Bork. Comparative assessment of large-scale data sets of protein-protein interactions. *Nature*, 417(6887):399–403, 2002.
25. A. Wagner. The yeast protein interaction network evolves rapidly and contains few redundant duplicate genes. *Mol Biol Evol*, 18(7):1283–92, 2001.
26. D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–2, 1998.
27. E. Yeger-Lotem, S. Sattath, N. Kashtan, S. Itzkovitz, R. Milo, R. Y. Pinter, U. Alon, and H. Margalit. Network motifs in integrated cellular networks of transcription-regulation and protein-protein interaction. *Proc Natl Acad Sci U S A*, 101(16):5934–9, 2004.
28. S. H. Yook, H. Jeong, A. L. Barabasi, and Y. Tu. Weighted evolving networks. *Phys Rev Lett*, 86(25):5835–8, 2001.

A Algorithms

This section is devoted to a brief description of the algorithms and methods used to derive the various statistics used in the study of the yeast regulation network (see Section 3).

A.1 Adjacency matrices

The various graphs were represented using adjacency matrices, the adjacency matrix $M(G)$ of a graph $G = (E, V, s, t)$ being defined as:

$$M(i,j) = \begin{cases} 1 & \text{if } \exists e \in E, s(e) = i \text{ and } t(e) = j \\ 0 & \text{otherwise} \end{cases}$$

for $(i,j) \in V^2$. All computations were done under the simplifying assumption that graphs were undirected, hence working with a symmetrised form of the matrix M. It is worth noting that working with directed graphs could refine the statistical picture.

A.2 Shuffle of the TRI network

Shuffling consists in permuting the indices of the matrix. That is, writing σ for a permutation of V, the permuted adjacency matrix $\sigma(M)$ is defined as:

$$\sigma(M)(i,j) = M(\sigma(i), \sigma(j))$$

This permutation has to be drawn uniformly among the $|V|!$ such possible permutations (this is done using a classical algorithm [**?**]). Once the TRI network has been shuffled, it is glued together with the PPI network by applying the `OR` operator entry by entry on the adjacency matrices (an operation which forgets which subgraph edges came from).

A.3 Computation of the shortest path lengths distribution

Clearly the (i,j) coefficient of M^n is the number of paths of length n connecting i to j in the graph underlying M. Since we are only interested in knowing whether two nodes are connected by a path of a given length we may use a simplified matrix product defined as:

$$M^n(i,j) = \begin{cases} 1 & \text{if } \exists k \in V : M^{n-1}(i,k) = M(k,j) = 1 \\ 0 & \text{otherwise} \end{cases}$$

which is forgetting the numbers of connecting paths, only to remember whether there is at least one.

Furthermore, the addition of the identity matrix I to the adjacency matrix before the computation of the products gives an immediate access to the value of the cumulative distribution function of the distances in the network. Indeed, writing $\widehat{M} = M + I$:

$$\widehat{M^n}(i,j) = \begin{cases} 1 & \text{if } \exists k \in V, M^{n-1}(i,k) = M(k,j) = 1 \\ & \text{or } M^{n-1}(i,j) = 1 \\ 0 & \text{otherwise} \end{cases}$$

Thus the number of 1s in $\widehat{M}(G)^n$ is the number of ordered pairs connected by at least one path of length $\leq n$, and the whole distribution is obtained when the computation reaches a fixpoint. This algorithm can be optimised by being run separately on each connected component (these components can be obtained by a prior and faster computation). Computing the distribution on the real PPI-TRI graph takes about 30' on a recent computer; the distribution for the 700 shuffles were computed down in 12 hours on a grid of 41 computers hosted by the Genoscope.

B Statistics

This section details the definition and computation of p-values.

In order to compute p-values for the deviation of the average distance in the real network from its distribution over the sample of shuffled networks, we need to approximate this distribution by a Gaussian one, whose mean and standard deviation are fixed to the empirical values computed on the sample. This is necessary, since a direct estimation of the p-value as the proportion of shuffled networks with a larger average distance than in the real network would yield a degenerately optimistic estimate of 0.

The empirical values observed are $m = 3.7$ for the mean, and $\sigma = 1.02 \times 10^{-2}$ for the standard deviation.

Assuming this average distance is a Gaussian random variable A with those parameters, the p-value of the deviation of the average distance in the real network from its distribution over the sample of shuffled networks is defined as:

$$p = \mathbb{P}(A > m_c), \quad with\ A \sim \mathcal{N}(m, \sigma)$$

where $m_c = 3.94$ is the observed average distance in the real network. In this case, this yields $p = 8.1 \times 10^{-134}$.