Journal Subline

LNBI 4220

Transactions on Computational Systems Biology VI

Corrado Priami

Editor-in-Chief

Springer

Corrado Priami   Gordon Plotkin (Eds.)

Transactions on Computational Systems Biology VI

Springer

Volume Editor

Gordon Plotkin
The University of Edinburgh
Division of Informatics
Mayfield Road, Edinburgh EH9 3JZ, UK
E-mail: gdp@inf.ed.ac.uk

Library of Congress Control Number: 2006935913

CR Subject Classification (1998): J.3, F.1, F.4, I.6

LNCS Sublibrary: SL 8 – Bioinformatics

ISSN 1861-2075
ISBN-10 3-540-45779-8 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-45779-4 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11880646 06/3142 5 4 3 2 1 0

Preface

This issue of Transactions on Computational Systems Biology contains a fully-refereed selection of papers from the Fourth International Conference on Computational Methods in Systems Biology, held in Edinburgh, Scotland, April 3–5, 2005. I would like to thank both the referees for all their hard work and also the CMSB 2005 programme committee for their help in choosing which papers to invite for submission.

June 2006

Gordon Plotkin
Program Chair
CMSB 2005

LNCS Transactions on Computational Systems Biology – Editorial Board

Lecture Notes in Bioinformatics

Vol. 4230: C. Priami, A. Ingólfsdóttir, B. Mishra, H.R. Nielson (Eds.), Transactions on Computational Systems Biology VII. VII, 185 pages. 2006.

Vol. 4220: C. Priami, G. Plotkin (Eds.), Transactions on Computational Systems Biology VI. VII, 247 pages. 2006.

Vol. 4216: M.R. Berthold, R. Glen, I. Fischer (Eds.), Computational Life Sciences II. XIII, 269 pages. 2006.

Vol. 4210: C. Priami (Ed.), Computational Methods in Systems Biology. X, 323 pages. 2006.

Vol. 4205: G. Bourque, N. El-Mabrouk (Eds.), Comparative Genomics. X, 231 pages. 2006.

Vol. 4175: P. Bücher, B.M.E. Moret (Eds.), Algorithms in Bioinformatics. XII, 402 pages. 2006.

Vol. 4146: J.C. Rajapakse, L. Wong, R. Acharya (Eds.), Pattern Recognition in Bioinformatics. XIV, 186 pages. 2006.

Vol. 4115: D.-S. Huang, K. Li, G.W. Irwin (Eds.), Computational Intelligence and Bioinformatics, Part III. XXI, 803 pages. 2006.

Vol. 4075: U. Leser, F. Naumann, B. Eckman (Eds.), Data Integration in the Life Sciences. XI, 298 pages. 2006.

Vol. 4070: C. Priami, X. Hu, Y. Pan, T.Y. Lin (Eds.), Transactions on Computational Systems Biology V. IX, 129 pages. 2006.

Vol. 3939: C. Priami, L. Cardelli, S. Emmott (Eds.), Transactions on Computational Systems Biology IV. VII, 141 pages. 2006.

Vol. 3916: J. Li, Q. Yang, A.-H. Tan (Eds.), Data Mining for Biomedical Applications. VIII, 155 pages. 2006.

Vol. 3909: A. Apostolico, C. Guerra, S. Istrail, P. Pevzner, M. Waterman (Eds.), Research in Computational Molecular Biology. XVII, 612 pages. 2006.

Vol. 3886: E.G. Bremer, J. Hakenberg, E.-H.(S.) Han, D. Berrar, W. Dubitzky (Eds.), Knowledge Discovery in Life Science Literature. XIV, 147 pages. 2006.

Vol. 3745: J.L. Oliveira, V. Maojo, F. Martín-Sánchez, A.S. Pereira (Eds.), Biological and Medical Data Analysis. XII, 422 pages. 2005.

Vol. 3737: C. Priami, E. Merelli, P. Gonzalez, A. Omicini (Eds.), Transactions on Computational Systems Biology III. VII, 169 pages. 2005.

Vol. 3695: M.R. Berthold, R. Glen, K. Diederichs, O. Kohlbacher, I. Fischer (Eds.), Computational Life Sciences. XI, 277 pages. 2005.

Vol. 3692: R. Casadio, G. Myers (Eds.), Algorithms in Bioinformatics. X, 436 pages. 2005.

Vol. 3680: C. Priami, A. Zelikovsky (Eds.), Transactions on Computational Systems Biology II. IX, 153 pages. 2005.

Vol. 3678: A. McLysaght, D.H. Huson (Eds.), Comparative Genomics. VIII, 167 pages. 2005.

Vol. 3615: B. Ludäscher, L. Raschid (Eds.), Data Integration in the Life Sciences. XII, 344 pages. 2005.

Vol. 3594: J.C. Setubal, S. Verjovski-Almeida (Eds.), Advances in Bioinformatics and Computational Biology. XIV, 258 pages. 2005.

Vol. 3500: S. Miyano, J. Mesirov, S. Kasif, S. Istrail, P. Pevzner, M. Waterman (Eds.), Research in Computational Molecular Biology. XVII, 632 pages. 2005.

Vol. 3388: J. Lagergren (Ed.), Comparative Genomics. VII, 133 pages. 2005.

Vol. 3380: C. Priami (Ed.), Transactions on Computational Systems Biology I. IX, 111 pages. 2005.

Vol. 3370: A. Konagaya, K. Satou (Eds.), Grid Computing in Life Science. X, 188 pages. 2005.

Vol. 3318: E. Eskin, C. Workman (Eds.), Regulatory Genomics. VII, 115 pages. 2005.

Vol. 3240: I. Jonassen, J. Kim (Eds.), Algorithms in Bioinformatics. IX, 476 pages. 2004.

Vol. 3082: V. Danos, V. Schachter (Eds.), Computational Methods in Systems Biology. IX, 280 pages. 2005.

Vol. 2994: E. Rahm (Ed.), Data Integration in the Life Sciences. X, 221 pages. 2004.

Vol. 2983: S. Istrail, M.S. Waterman, A. Clark (Eds.), Computational Methods for SNPs and Haplotype Inference. IX, 153 pages. 2004.

Vol. 2812: G. Benson, R.D.M. Page (Eds.), Algorithms in Bioinformatics. X, 528 pages. 2003.

Vol. 2666: C. Guerra, S. Istrail (Eds.), Mathematical Methods for Protein Structure Analysis and Design. XI, 157 pages. 2003.

Table of Contents

Property-Driven Statistics of Biological Networks

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

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

Abstract. An analysis of heterogeneous biological networks based on randomizations that preserve the structure of component subgraphs is introduced and applied to the yeast protein-protein interaction and transcriptional regulation network. Shuffling this network, under the constraint that the transcriptional and protein-protein interaction subnetworks are preserved reveals statistically significant properties with potential biological relevance. Within the population of networks which embed the same two original component networks, the real one exhibits simultaneously higher bi-connectivity (the number of pairs of nodes which are connected using both subnetworks), and higher distances. Moreover, using restricted forms of shuffling that preserve the interface between component networks, we show that these two properties are independent: restricted shuffles tend to be more compact, yet do not lose any bi-connectivity.

Finally, we propose an interpretation of the above properties in terms of the signalling capabilities of the underlying network.

1 Introduction

The availability of genome-scale metabolic, protein-protein interaction and regulatory networks [25,7,3,5,21] —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 [23,11,6,28] (*i.e.*, how the connectivity structure of the biological process reflects its dynamics), on their evolution [10,30,27] (*i.e.*, likely scenarios for the evolution of a network exhibiting the observed property or properties), or both [9,14,15]. 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 [17,18,19,22,4], the distribution of vertex-vertex distances [22], and more recently the distribution of network motifs occurrences [15].

C. Priami and G. Plotkin (Eds.): Trans. on Comput. Syst. Biol. VI, LNBI 4220, pp. 1–15, 2006.

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 [20,1,24]. For the simplest random graph models, such as the classical Erdös-Rényi model (where each pair of vertices is connected with constant probability p, [2]), analytical derivations of the simplest of the above graph properties are known [20,1].

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 [23,14,29]. 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. Another, somewhat more empirical, category of approaches reverses the process: variants are generated from the network of interest using a random rewiring procedure. The procedure selects and moves edges randomly, preserving the global number of edges, and optionally their type, as well as local properties such as the degree of each vertex. Rewirings are thus heuristic procedures which perform a sequence of local modifications on the structure of the network.

The specific focus of the present paper is on measuring the degree of cooperation between the two subgraphs of the yeast graph of interactions induced by the natural partition of edges as corresponding either to transcriptional interaction (directed) or to protein protein interaction (undirected). To evaluate a potential deviation with respect to such a measure, one needs as a first ingredient a suitable notion of random variation of the original graph. The goal is here, as in many other cases, to contrast values of a given observable on the real graph, against the distribution of those same values in the population of variants. We define *shuffles* of the original graph as those graphs that are composed exactly of the original two subgraphs of interest, the variable part being the way these are 'glued' together.

From the probabilistic point of view, this notion of randomisation coincides with a traditional Erdös-Renyi statistics, except that it is conditioned by the preservation of the original subgraphs. Designing a generative random graph model that would only yield networks preserving this very precise property seems to be a hard endeavor ; it is not as easy as in the unconditional Erdös-Renyi model to draw edges step by step yet ensure that component subgraphs will be obtained in the end. Shuffling might also be seen as rewiring, except the invariant is large-scale and extremely precise: it is not edges that are moved around but entire subgraphs. Moving edges independently would break the structure of the subnetworks, and designing a sequential rewiring procedure that eventually recovers that structure is not an obvious task. Moreover, it would be in general difficult to ensure the uniformity of the sample ; see [16] for a thorough analysis of rewiring procedures. This choice of an invariant seems rather natural in that one is interested in qualifying the interplay between the original subgraphs in the original graph. Now, it is not enough to have a sensible notion of randomisation, it is also crucial to have a computational handle on it. Indeed, whatever the observable one wants to use to mark cooperation is, there is little hope of obtaining an analytic expression for its distribution, hence one needs sampling. Fortunately, it turns out it is easy to generate shuffles uniformly, since these can be described by pairs of permutations over nodes, so that one can always sample this distribution for want of an exact expression. As explained below in more details, the analysis will use two different notions of subgraph-preserving sampling: *general* shuffles, and *equatorial* ones that also preserve the interface between our two subgraphs. Equatorial shuffles are feasible as well, and in both cases the algorithms for sampling and evaluating our measures turn out to be fast enough so that one can sweep over a not so small subset of the total population of samples.

Regarding the second necessary ingredient, namely which observable to use to measure in a meaningful way the otherwise quite vague notion of cooperation, there are again various possibilities. We use two such observables in the present study: the *connectivity*, defined as the percentage of disconnected pairs of nodes, and a refined quantitative version of connectivity, namely the full distance distribution between pairs of nodes. The latter is costlier, requiring about three hours of computation for each sample on a standard personal computer.

Once we have both our notion of randomisations and our observables in place, together with a feasible way of sampling the distribution of the latter, we can start. Specifically we run four experiments, using general or equatorial shuffling, and crude or refined connectivity measures. The sampling process allows us to compare the values of these measures for the original graph with the mean value for the sample, and, based on the assumption that those values follow a normal distribution over the sample, one can also provide a p-value that gives a rough estimate of the statistical deviation of the observable in the given graph.

The general shuffle based experiments show with significant statistical confidence that shuffling reduces connectivity (1), and at the same time contracts distances (2). More precisely, both bi-connectivity (the amount of pairs of nodes which are connected using both subgraphs) and distances are higher than average in the real network. A first interpretation might be that the real graph is trading off compactness for better bi-connectivity. In order to obtain a clearer picture and test this interpretation, we perform two other experiments using equatorial shuffles. Surprisingly, under equatorial shuffles connectivity hardly changes, while the global shift to shorter distances is still manifest. It seems therefore there is actually no trade-off, and both properties (1) and (2) have to be thought of as being independently captured by the real graph. With appropriate caution, we may try to provide a biological interpretation of this phenomenon. Since all notions of connectivity and distances are understood as directed, we propose to relate this to signalling, and interpret bi-connectivity as a measure of the capability to convey a signal between subgraphs. With this interpretation, the above properties may be read as:(1) signal flows better than average and (2) signal is more specific than average. The second point requires explanation. At constant bi-connectivity, longer average distances imply that upon receipt of a signal, the receiver has a better chance of guessing the emitter. In other words, contraction of distances (which can be easily achieved by using hubs) will anonymise signals, clearly not a desirable feature in a regulatory network. Of course this is only part of the story, since some hubs will also have an active role in signal integration and decision making. The latter is probably an incentive for compactness. If our reading of the results is on track, we then may think of the above experiments as showing that the tropism to compactness due to the need for signal integration, is weaker than the one needed for signal specificity.

Beyond the particular example we chose to develop here because of the wealth of knowledge available on the yeast regulatory and protein interaction networks, one can think of many other applications of the shuffling methodology for heterogeneous networks. The analyses performed here rely on edges corresponding to different types of experimental measurements, but edges could also represent different types of predicted functional links. Indeed, there are many situations where a biological network of interactions can be naturally seen as heterogeneous. Besides, the notions of shuffle we propose can also accomodate the case where one would use a partition of nodes, perhaps given by clustering, or localisation, or indeed any relevant biological information, and they may therefore prove useful in other scenarios.

The paper is organised as follows: first, we set up the definitions of edge-based general and equatorial shuffles based, and also consider briefly node-based shuffles though these are not used in the sequel; then we describe the interaction network of interest and the way it was obtained; finally we define our observables and experiments, and interpret them. In the conclusion, we discuss generalization and potential applications of the method. The paper ends with an appendix on the algorithmical aspects of the experiments, and a brief recall of the elementary notions of statistics we use to assert their significance.

2 Shuffles

Let $G = (V, E)$ be a directed graph, where V is a finite set of nodes, and E is a finite set of directed edges over V. We write M for the incidence matrix associated to G. Since G is directed, M may not be symmetric. In the absence of parallel edges M has coefficients in $\{0, 1\}$, where parallel edges are allowed.

Given such a matrix M and a permutation σ over V, one writes $M\sigma$ for the matrix defined as for all u, v in V:

$$M\sigma(u, v) := M(\sigma^{-1}u, \sigma^{-1}v)$$

Note that $M\sigma$ defines the same abstract graph as M does, since all σ does is changing the nodes names.

2.1 Shuffles Induced by Properties on Edges

We consider first shuffles induced by properties on edges. Suppose given a partition of $E = \sum E_i$; this is equivalent to giving a map $\kappa : E \to \{1, \ldots, p\}$ which one can think of as colouring edges.

Define M_i as the incidence matrix over V containing the edges in E_i (of colour i).

Define also V_I, where $I \subseteq \{1, \ldots, p\}$, as the subset of nodes v having for each $i \in I$ at least one edge incident to v with colour i, and no incident edge coloured j for $j \notin I$. We abuse notation and still write $\kappa(u) = I$ when $u \in V_I$. This represents the set of colours seen by the nodes.

Clearly $V = \sum V_I$, $V_\varnothing$ is the set of isolated nodes of G, and the set of nodes of G_i is the union of the graphs generated by V_I, for $i \in I$.

Given $\sigma_1, \ldots, \sigma_p$ permutations over V, define the *global shuffle* of M as:

$$M(\sigma_1, \ldots, \sigma_p) := \textstyle\sum_i M_i \sigma_i$$

The preceding definition of $M\sigma$ is the particular case where $p = 1$ (one has only one colour common to all edges). Each G_i (the abstract graph associated to M_i) is preserved up to isomorphism under this transformation. However the way the G_is are glued together is not, since one uses a different local shuffle on each.

For moral comfort, we can check that any means of glueing together the G_is is obtainable using a general shuffle in the following sense: given G' and $\sum q_i : \sum G_i \to G'$ where the disjoint sum $\sum_i q_i$ is an isomorphism on edges, one has that G' is a general shuffle of G. To see this, define $\sigma_i(u) := q_i p_i^{-1}(u)$ if $u \in \kappa^{-1}(i)$, $\sigma_i(u) = u$ else (we have written p_i for the inclusion of G_i in G), one then has $G' = \sum G_i \sigma_i = G(\sigma_1, \ldots, \sigma_p)$.

Note also that $(M(\sigma_1, \ldots, \sigma_p))\tau := \sum_i M_i(\tau\sigma_i)$, and so in particular, without loss of generality one can take any the σ_i's to be the identity (just take $\tau = \sigma_i^{-1}$). This is useful when doing actual computations, and avoids some redundancy in generating samples.

An additional definition will help us refine the typology of shuffles. One says a shuffle $M\sigma$ is *equatorial* if in addition for all I, and all i, V_I is closed under σ_i. Equivalently, one can ask that $\kappa \circ \sigma_i = \kappa$. An *equatorial shuffle* preserves the set of colours associated with each node and in particular preserves for a given pair of nodes (u, v) the fact that (u, v) is heterochromatic, *i.e.*, $\kappa(u) \cap \kappa(v) = \varnothing$. This in turn implies that the distance between u and v must be realised by a path which uses edges of different colours. In the application such paths are mixing different types of interaction, and are therefore of particular interest; without preserving this attribute, an observable based on path with different colours would not make sense. In the particular case of two colours, nodes at the 'equator', having both colours, will be globally preserved, hence the name.

2.2 Shuffles Induced by Properties on Vertices

One can also consider briefly shuffles induced by properties on nodes. Suppose then given a partition of nodes $V = \sum_i V_i$, again that can be thought of as a colouring of nodes $\kappa : V \to \{1, \ldots, p\}$, and extended naturally to the assignment of one or two colours to each edge.

A node shuffle is defined as a shuffle associated to σ which can be decomposed as $\sum_i \sigma_i$, σ_i being a permutation over each cluster V_i. Clearly each graph G_i generated by V_i is invariant under the transformation: only the inter-cluster connectivity is modified.

The equivalent of the equatorial constraint would be to require in addition $\sigma(u) \in \partial V_i$ if $u \in \partial V_i$, where ∂V_i is defined as those nodes of V_i with an edge to some V_j, $i \neq j$. Other variants are possible and the choice of the specific variant will likely depend on the particular case study. We now turn to the description of the network the shuffle experiments will be applied to.

3 A Combined Network of Regulatory and Protein-Protein Interactions in Yeast

With our definitions in place, we can now illustrate the approach on a heterogeneous network obtained by glueing two component networks.

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 [3,5], 8531 physical interactions generated using high-throughput Y2H assays [26], and 7455 direct regulatory interactions compiled from literature and from ChIP-Chip experiments [4,12], connecting a total of 6541 yeast proteins. A subnetwork of high-reliability interactions was selected, using a threshold on the confidence levels associated to each inferred interaction. For the ChIP-Chip data produced by Lee et al. [12], interactions with a p-value inferior to 3.10^{-2} were conserved ; for the Y2H data produced by Ito et al. [26], a threshold of 4.5 on the Interaction Sequence Tag was used (see [8]). The PPI network was built by connecting two proteins, in both directions, whenever there was 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 simplify the discussion, we will refer in the rest of the paper to the TRI graph as TRI, and to the PPI graph as PPI. With some more precision, define G as the real graph, *TRI* as the subgraph induced by the set of TRI nodes, *i.e.*, nodes such that $TRI \in \kappa(u)$, and *PPI* as the subgraph induced by the set of PPI nodes.

Their respective sizes are:

$$TRI = 3387, PPI = 2517, TRI \cup PPI = 4489, TRI \cap PPI = 1415$$

The set of nodes $TRI \cap PPI$ of both colours is also referred to in the sequel as the *equator* or the *interface.* Since the object of the following is to discuss the interplay between the *TRI* and *PPI* subgraphs, the interface naturally plays an important role. A qualitative measure of the connectivity between *TRI* and *PPI* which will be useful later in the discussion, is the number of bi-connected pairs in G (these are the pairs which are connected in G, but not connected in either *TRI* or *PPI*), which is roughly $p_{bi} = 23\%$. To complete this statistical portrait of the data, we provide in figure 1 the histograms of degree distributions in the PPI and TRI networks, with in and out degrees pictured separately for the latter. Figure 1 also shows the hub size distribution for the TRI network (the PPI network has no non-trivial hubs). Note that hubs are defined as sets of nodes connected to a single node. The TRI network (here considered as unoriented) has 124 such hubs ; the histogram of the distribution of their sizes is given in figure 1.

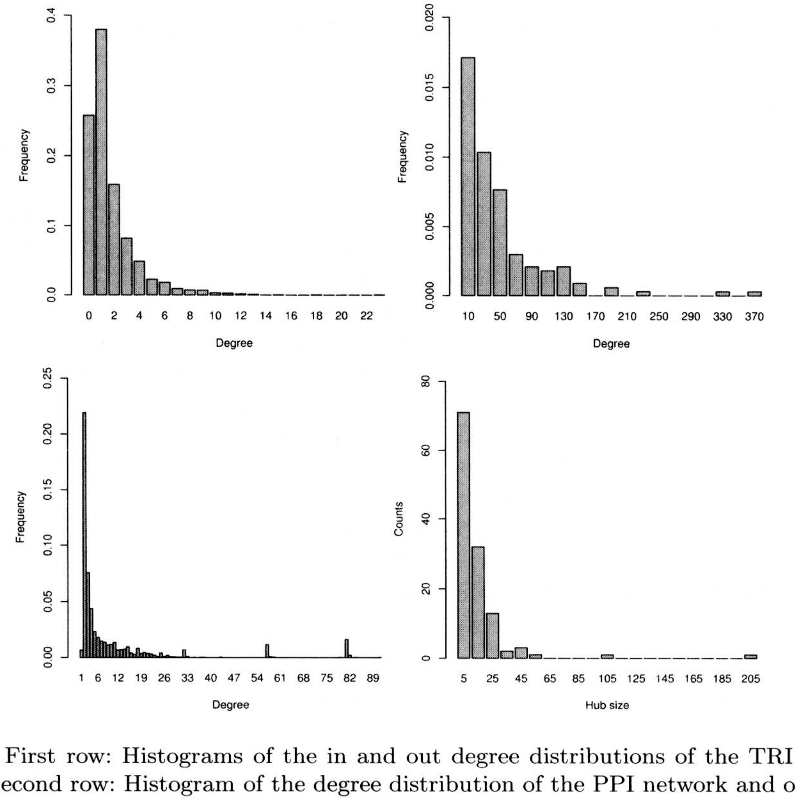

Fig. 1. First row: Histograms of the in and out degree distributions of the TRI network. Second row: Histogram of the degree distribution of the PPI network and of the distribution of the hub size in the TRI network.

4 Results and Interpretations

Hereafter, notions of connectivity, distance, etc. should be understood as *directed* unless explicitly stated otherwise. We now turn to the various shuffle experiments and consecutive observations.

4.1 General Shuffle vs Connectivity

We take here as a rough measure of the connectivity of a graph the percentage of unconnected pairs. Comparing first the real graph with the randomised versions under the general shuffle, one finds that in the average 4% of the population pairs are disconnected under shuffle. So general shuffle disconnects, or in other words G maximises bi-connectivity.

Clearly mono-connected pairs (pairs connected in either PPI or TRI) cannot be disconnected under general shuffle; a pair is 'breakable' only if bi-connected in G; therefore a more accurate measure of the connectivity loss under general shuffle is that about 17.5% of the breakable pairs are actually broken (this obtained by dividing by p_{bi}), a rather strong deviation with a p-value below 10^{-11}.

Inasmuch as a directed path can be thought of as a signal-carrying pathway, one can interpret the above as saying that the real graph connects PPI and TRI so as to maximise the bandwidth between the subgraphs.

4.2 Equatorial Shuffle vs Connectivity

Keeping with the same observable, we now restrict to equatorial shuffles. One sees in this case that no disconnection happens, and actually about 1% more pairs are connected *after* shuffling. The default of connected pairs of the real graph has a far less significant p-value of 3%. However the point is that equatorial shuffles leaves bi-connectivity rather the same.

This complements the first observation and essentially says that the connectivity maximisation seen above is a property of the set of equatorial nodes ($\{$TRI,PPI$\}$ nodes) itself, and not of the precise way TRI and PPI edges meet at the equator.

Both observations can be understood as saying that the restriction of G to the equator is a much denser subgraph than its complement (as evidenced by the connectivity loss under general shuffle), and dense enough so that equatorial shuffling does not impact connectivity.

Note that so far the observable is somewhat qualitative, being only about whether a pair is connected or not. Using a refined and quantitative version of connectivity, namely the distribution of distances (meaning for each n the proportion of pairs at distance n), will reveal more.

4.3 Impact of Equatorial Shuffles on Distance Distribution

Using this refined observable, one sees that the whole histogram shifts to the left, so equatorial shuffle contracts the graph (Fig. 2). This is confirmed by the equality between the number of lost pairs at distance 7 to 9 and the number of

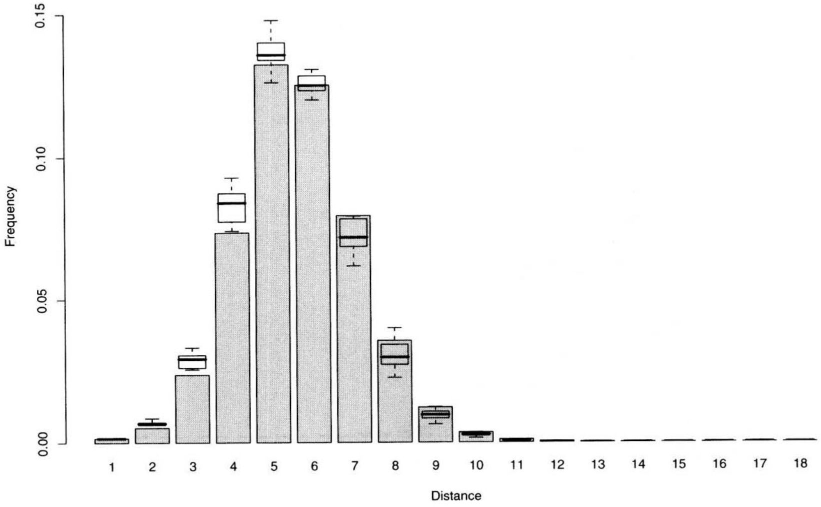

Fig. 2. Equatorial shuffle distance histogram: grey boxes stand for the real graph; one sees that shuffles have more pairs at shorter distance, and consequently (because the number of connected pairs is about the same) less such at higher distances

new ones at distance 3 to 5. In accordance with the preceding experiment, one also does not see any disconnection under equatorial shuffle.

This is to be compared with the general shuffle version (Fig. 3) where both effects are mixed, and the cumulated excess of short pairs does not account for the loss of long pairs (indeed we know 4% are broken, *i.e.*, disappear at infinity and are not shown on the histogram).

To summarize the distance distribution results in a single number, one can compute the deviation of the real graph mean distance under both shuffles. As expected the mean distance is higher in the real graph with respective *p*-values of 0.2% and 2% in the general and equatorial shuffles (see Appendix for details). We conclude that while the real graph does maximise bi-connectivity, it does not try to minimise the associated distances.

To provide an intuition on the potential interpretation of the above result, let us again consider paths as rough approximations of signalling pathways. Now compare a completely linear chain-shaped graph and star-shaped one, with the same number of nodes and edges. In the star case, any two nodes are close, at constant distance 2, while in the chain distances are longer. As said, compactness comes with a price, namely that in a star graph all signals go through the hub and are anonymised, *i.e.*, there may be a signal, but there is no information whatsoever in the signal about where the signal originated from. Quite the opposite happens in a linear graph. Of course this is an idealized version of the real situation; nevertheless it is tempting to interpret this last observation as an indication that the real graph is trading off fast connectivity against specificity of signals. The heterogeneous network is likely to result from a trade-off between causality and signal integration.