Modularity in PPI Networks: Characteristics of Existing Networks and Models of Evolution

Min Ye[1], Xiuwei Zhang[2], and Bernard M.E. Moret[1]
[1]School of Computer and Communication Sciences, EPFL
Lausanne, 1015, Switzerland
[2]Simons Institute, UC Berkeley
Berkeley, 94720, USA
(minye.epfl, zhangxiuwei03, bmemoret)@gmail.com

Abstract

Networks are often used to represent interactions in biology. Databases store many such networks, observed or inferred. Generative models have been proposed for various biological networks; for PPI networks, current models are based on duplication and divergence (D&D): a node is duplicated and inherits some subset of the connections of the original node. An early finding about biological networks was modularity: these networks present a higher-level structure consisting of well connected subgraphs with lower connectivity to other subgraphs. While D&D models generate networks with a modular structure, there has been no comparison of these modular structures with those in the databases.

We study the PPI networks of six model organisms across six databases to uncover commonalities in network structure so as to compare D&D models with our module-aware model NEMo. By restricting our data to high-confidence interactions, some shared structural characteristics (beyond the presence of modules) can be identified among the six species and six databases. When comparing these characteristics with those of the networks produced by D&D models and NEMo, we further find that NEMo-generated networks come much closer to the PPI networks stored in databses. We conclude that modularity in PPI networks takes a particular form, one that is better approximated by the module-aware NEMo model than by models that do not take modularity into consideration.

1 Introduction

Networks are commonly used to represent key processes in biology. The network is typically a graph, directed or undirected, where edges or arcs represent interactions and vertices represent actors or targets. Current methods for building such models range from experimental determination of specific interactions (expensive and time-consuming), through high-throughput experimental methods such as affinity-purification mass spectrometry (AP MS)) [25] (which suffer from large error rates, such as large numbers of false positives for AP MS), to inference through computational methods, ranging from data mining the literature (see, e.g., [23, 13, 1]) to inferring the evolutionary history of networks from present observations [10, 46, 47, 34]. (Makino and McLysaght [22] present a thorough discussion of evolutionary approaches in the case of PPI networks.) A variety of databases, with vastly different levels of curation and annotation, store these networks, some with the aim of gathering all plausible interactions, others focused on interactions obtained through specific methods.

An early finding about regulatory and PPI networks was the presence of modularity [14]: these networks do not have similar connectivity at every vertex, but instead present a higher-level structure consisting of well connected subgraphs with less substantial connectivity to other such subgraphs. Modularity is now viewed as one of the main characteristics of living systems [37]. While some of the models devised for networks lead automatically to the emergence of modules within the network [39], these models are purely generative—increasing the size of the network at each step; moreover, the types of modular structure they create have not been compared to those found in biological networks. Recently, we proposed an evolutionary model for PPI networks, NEMo, that explicitly takes modularity into account [45].

In this paper, we take two additional steps to further our understanding of modularity in PPI networks and of suitable evolutionary models that support this modularity. In the first step, we conduct an extensive study of PPI networks for six model species across many public databases, in order to identify any commonalities in network structure across the databases and, as appropriate, across model species. We show that, after filtering out interactions (edges) of lower confidence, we can identify a number of structural features, both at the level of the entire network and at the level of individual modules, that extend across both species and databases. These structural features can then be taken as references in our second step, in which we compare them with comparable features produced by existing network models

978-1-943436-07-1 / copyright ISCA, BICOB 2017
March 20-22, 2017, Honolulu, Hawaii, USA

as well as by our NEMo model, in order to characterize how well these various models do in generating the type of structure and modularity observed in PPI networks. We show that NEMo, a model that explicitly takes modularity into account, comes much closer to producing these same structural features than current models (all of which operate strictly at the node and edge level).

2 Background: Evolutionary Models

All evolutionary models to date are based on the addition or removal of the basic constituent elements of the network: vertices (proteins) and edges (pairwise interactions). Most of the recent models are based on duplication followed by divergence, denoted D&D [32, 4], in which a vertex (gene) is duplicated and inherits some randomly chosen subset of the connections of the original vertex. (Since the copy of the gene could initially produce much the same proteins as the original, it would enter into many of the same interactions.) D&D models are favored because most evolutionary biologists view gene duplication (single gene, a segment of genes, or even the entire genome) as the most important source of diversification in genomic evolution [28, 21].

2.1 D&D models

D&D models consider both speciation and gene duplication events. Following a speciation event, interactions (edges) can be gained or lost with specified probabilities. A duplication event duplicates all interactions of the original copy, but some interactions for both the original and the duplicated copies are immediately lost with some probability. The duplication-mutation-complementarity (DMC) model [24, 26, 42] forbids the simultaneous loss of the same interaction in the original and in the copy and allows the duplicated gene to gain a direct interaction with the original gene. The DMR (random mutation) model [38] allows the introduction of new interactions (not among those involving the original vertex) between the duplicate vertex and some random vertices in the network.

D&D models give rise to modular structures. However, the type of modular structure that results from these models has not been characterized nor compared to the structure observed in PPI networks. (In addition, D&D models are generative models, working in scenarios of unrestricted growth: no edge deletions are allowed other than those that occur as part of a vertex duplication, and a vertex gets deleted only indirectly, if and when its degree is reduced to zero.)

2.2 NEMo

In a previous paper [45] we introduced NEMo, a two-level model with modularity for PPI networks that includes both growth and reduction operators and explicitly models the influence of modularity on network evolution. The lower level is based on the DMC model, but allows random mutations for each vertex. The higher level is "module-aware," in the sense that interactions can be partitioned into "within module," "between modules," or "unrelated to modules," where modules are dynamically reconfigured during evolution using a clustering algorithm, thereby allowing implicit events such as birth, death, splitting, and merging of modules.

Such a model requires the identification of modules within a network and the extraction and quantification of some high-level attributes that can be used to measure similarity. Methodologies used in much of the work on the identification of functional modules [9, 7, 2] are not applicable here, as we deal with an anonymous graph, not with annotated proteins. We rely in part on clustering algorithms (to detect clusters, which we regard as potential modules, within the graph) and in part on matching high-level attributes of actual PPI networks and using these attributes to measure drift in the course of evolution. There are several families of clustering algorithms used in the biological domain. In this study, we use mainly ClusterOne [27], a graph clustering algorithm that allows overlapping clusters that has proved useful for detecting protein complexes in PPI networks, and a Markov clustering algorithm, MCL [8, 11, 41], which finds the clusters by iterative flow simulation.

3 Materials and Methods

3.1 PPI networks in current databases

As we showed in a previous paper [45], PPI networks for the same species can vary enormously from one database to the next. In particular, databases, such as STRING [40], that seek to amass as many interactions as possible, have very little in common with databases, such as HPRD [31], that are manually curated for a single organism. Fortunately, the more inclusive databases also offer a confidence score for their entries; previous experience indicated that restricting the entries to those with high confidence scores led to a subnetwork much more in line with those of other databases. For such databases, we use both the full network and a subnetwork consisting of only high-confidence entries.

We work with six databases, some of which include several data sources. We chose six model organisms that are represented in most of these databases: *E. coli*, *S. cerevisiae*, *C. elegans*, *D. melanogaster*, *M. musculus*, and *H. sapiens*. Our data sources are thus the following:

STRING: The full STRING database [40] aims to provide a global perspective for as many organisms as feasible, tolerating lower-quality data and computational predictions, and thus including many inferred indirect interactions (which we view as false positive entries). STRING provides an evidence score for each interaction; we chose a high threshold of 900 to filter out as many indirect and low-quality interactions as possible.

HPRD: The manually curated HPRD [31] database maintains the PPI network for just one species, *H. sapiens*,

Table 1: PPI networks in various databases.

Species	S_{900}	H	M	P_a	P_h	D_a	D_c	F_j	F_b
E.c.	+	-	+	+	+	+	+	-	-
S.c.	+	-	+	+	+	+	+	+	+
C.e.	+	-	-	+	+	+	+	+	+
D.m.	+	-	-	-	-	+	+	-	-
M.m.	+	-	-	+	+	+	+	-	-
H.s.	+	+	+	+	+	+	+	+	+

and gives the network with the fewest false positives.

MAGNA++: In the paper describing the MAGNA++ algorithm [36] for global network alignment, the authors use a testbed with PPI networks for *H. sapiens* (9'141 proteins and 41'456 interactions) [33], for *E. coli* (1'941 proteins and 3'989 interactions) [30], and for *S. cerevisiae* (2'390 proteins and 161'277 interactions) [6].

HitPredict: This database stores experimentally determined protein-protein interactions with reliability scores [20, 29]. Nearly all entries are assigned a confidence score "Low" or "High", thus defining a complete dataset, P_a, and a high-confidence subset, P_h, respectively.

DIP: The manually curated Database of Interacting Proteins (DIP) [35] stores experimentally determined interactions between proteins with confidence annotations. We use the full dataset, D_a, and the set of entries assigned confidence value "core," D_c.

FunctionalNet: `www.functionalnet.org` collects probabilistic functional gene networks for a small number of species. We take the HumanNet [16] for H. sapiens, the Wormnet [17, 18] for C. elegans, and the YeastNet [19] for S. cerevisiae. The database provides full networks of all interactions, F_j, and benchmark sets, F_b.

Table 1 shows which species is represented in which database. Throughout this paper, S_{900} stands for the dataset with confidence scores at least 900 in the STRING database, H for HPRD, M for MAGNA++, P_a and P_h for HitPredict, D_a and D_c for DIP, and F_j and F_b for FunctionalNet.

3.2 Clustering algorithms

Modules are computed from the network through clustering, using two clustering algorithms: ClusterOne and MCL. ClusterOne [27] allows overlapping clusters and requires every cluster to meet or exceed a density threshold; we used half of the network's overall density as a (fairly modest) threshold. MCL [8, 11, 41] is a Markov clustering algorithm that finds clusters by iterative flow simulation and is tuned through an inflation parameter that enhances the contrast between well connected and poorly connected subgraphs, strongly influencing the number of clusters returned by MCL [5]. We use both the preset inflation value, 2.0 (MCL_{def}) and that recommended in [5], 1.8 ($MCL_{1.8}$).

Table 2 shows how many clusters each algorithm found in the networks in the various databases and versions. To run ClusterOne, we set the minimum size of a cluster to 1, the minimum density within a cluster to half of the global density of the network, and no penalty. Singleton nodes with no module membership are counted as individual modules of size 1. While the numbers of identified clusters can vary quite a bit between the three versions, they are strongly correlated.

Table 2: General characteristics of the six PPI networks in various databases.

Species	*Source*	*#nodes*	*#edges*	*#clusters Cluster1*	*#clusters MCL*	*#clusters* $MCL_{1.8}$
E.c.	S_{900}	*3'251*	*14'555*	*470*	*600*	*524*
S.c.	S_{900}	*5'162*	*68'190*	*686*	*564*	*409*
H.s.	S_{900}	*10'974*	*118'803*	*1'131*	*1'219*	*956*
M.m.	S_{900}	*10'020*	*125'427*	*872*	*1'117*	*925*
C.e.	S_{900}	*6'232*	*62'512*	*615*	*791*	*661*
D.m.	S_{900}	*6'946*	*62'423*	*732*	*1'004*	*873*
H.s.	H	*9'673*	*39'198*	*2'104*	*2'424*	*1'965*
E.c.	M	*1'941*	*3'989*	*381*	*908*	*760*
S.c.	M	*2'390*	*16'127*	*309*	*460*	*425*
H.s.	M	*9'141*	*41'456*	*1'671*	*3'771*	*3'130*
E.c.	P_a	*3'351*	*20'239*	*170*	*915*	*607*
S.c.	P_a	*6'019*	*84'740*	*10*	*178*	*89*
H.s.	P_a	*16'637*	*155'616*	*3'418*	*858*	*479*
M.m.	P_a	*5'011*	*12'135*	*1'002*	*1'049*	*1'002*
C.e.	P_a	*5'011*	*12'135*	*919*	*1'184*	*919*
E.c.	P_h	*2'512*	*9'407*	*575*	*731*	*942*
S.c.	P_h	*5'218*	*60'248*	*982*	*178*	*125*
H.s.	P_h	*14'213*	*135'718*	*2'983*	*625*	*360*
M.m.	P_h	*5'064*	*12'117*	*897*	*983*	*827*
C.e.	P_h	*3'093*	*7'328*	*574*	*191*	*652*
E.c.	D_a	*2'940*	*12'261*	*802*	*908*	*810*
S.c.	D_a	*5'176*	*22'975*	*1'091*	*1'229*	*967*
H.s.	D_a	*4'873*	*7'750*	*1'054*	*1'072*	*1'072*
M.m.	D_a	*2'331*	*2'577*	*558*	*683*	*616*
C.e.	D_a	*2'749*	*4'171*	*543*	*726*	*541*
D.m.	D_a	*7'011*	*23'262*	*1'877*	*2'223*	*1'885*
E.c.	D_c	*1'433*	*2'126*	*500*	*570*	*528*
S.c.	D_c	*2'409*	*5'300*	*436*	*521*	*455*
H.s.	D_c	*4'671*	*7'336*	*1'023*	*1'214*	*1'048*
M.m.	D_c	*331*	*2'577*	*558*	*683*	*616*
C.e.	D_c	*2'226*	*189*	*80*	*130*	*84*
D.m.	D_c	*634*	*706*	*161*	*180*	*163*
S.c.	F_j	*5'808*	*362'421*	*10*	*593*	*97*
H.s.	F_j	*46'243*	*476'399*	*33*	*3'370*	*2'014*
C.e.	F_j	*15'139*	*993'367*	*81*	*1'545*	*968*
S.c.	F_b	*4'172*	*81'953*	*430*	*204*	*75*
H.s.	F_b	*5'369*	*270'704*	*366*	*163*	*146*
C.e.	F_b	*5'178*	*626'342*	*178*	*77*	*168*

3.3 Measures

We reuse the measures introduced in our NEMo paper [45]. We compute these measures both on the entire network and on individual modules. In addition, we introduce a version of the Gini coefficient [12] to measure the unevenness of distribution of neighborhood sizes (the number of nodes within a certain radius—we chose a radius of two edges). We plot these measures to look for power laws and other distributions and compare plots across data sources and across species in order to discern general similarities across species or databases. Similarity here refers to structural and topological features such as modularity and connectivity: we need to compare networks very different in size and composition and so cannot use tools such as

network alignment methods. The six measures we compute both for the entire network and for each module are:

Cluster Coefficient (CC): The CC is based on triplets of vertices. A triplet is open if connected with two edges, closed if connected with all three edges. The CC is just the ratio of the number of closed triplets divided by the total number of (open or closed) triplets [44].

Graph Density (GD): the ratio of the actual number of edges to the number of possible edges.

Fraction of Edges Inside (FEI): the fraction of edges contained within modules. We expect it to be high since PPI networks are made of highly connected substructures that have only few connections to vertices outside the substructure[3, 43, 15].

Diameter (⊘): the length of the longest simple path in the graph.

Average Shortest Path (SPM): the average of all pairwise shortest paths in the graph.

Gini coefficient (Gini): If household i has a yearly income of x_i, then the Gini coefficient of the population is given by

$$G = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}|x_i - x_j|}{2n\sum_{i=1}^{n} x_i}$$

For our use in studying modularity we define the "income" of a node as the degree of the node plus the sum of the degrees of its immediate neighbors.

3.4 Simulations

The final part of our paper compares networks generated under various models with the common structural features discovered in the study of the PPI databases. We run a standard D&D model as well as two versions of our NEMo model, the normal version where the modular structure is re-evaluated during the evolution of the network and a deliberately crippled one in which no such re-evaluation takes place. We vary the number of steps, the interval between re-evaluations of the modular structure, the size of the networks, and the initial networks, along with some of the parameters of the NEMo model that affect the balance between inter- and intra-module events. Specifics of these parameter settings are given in the discussion of results.

4 The Structure of PPI Networks

4.1 Global PPI network structure

The very large differences in size among the databases for the same network are striking: the STRING database has well over 4 million edges for the human PPI network, whereas the HPRD database has fewer than 40'000, or less than 1% of the number in STRING. This large discrepancy illustrates why testing models or inferences against databases must be done with great care. For instance, simply clustering the graph has pitfalls, as shown

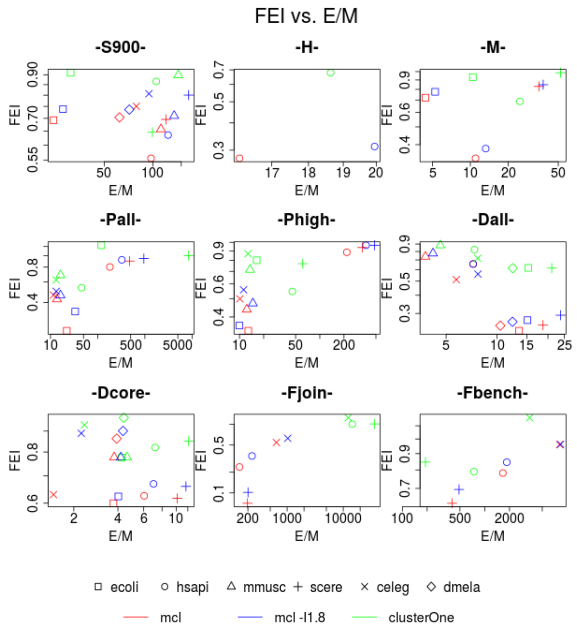

Figure 1: FEI over # E/#M plots across all data sources.

in the number of clusters found by the same algorithm for E. coli on the various databases, going from 16 clusters among 4'145 nodes in STRING to 1'151 clusters among 3'351 nodes in HitPredict—values that again differ by around two orders of magnitude. As we are interested in commonalities, we must keep in mind the effects of size on what we observe. The plots in Figure 1 provide a visualization of some of these measures in the various databases and versions.

The global clustering coefficient (CC) ranges in $[0.03, 0.45]$ overall, with just one exception (the benchmark set of FunctionalNet), with a much narrower range for most databases. Networks in S_{900} have a CC in $[0.39, 0.45]$ across all six species; in HitPredict the range is $[0.05, 0.30]$ for P_a and $[0.08, 0.43]$ for P_h; in DIP the range is $[0.02, 0.16]$ for D_a and $[0.08, 0.28]$ for D_c; and in the full set of FunctionalNet, the range is $[0.22, 0.24]$. In contrast, the range for the benchmark set of FunctionalNet is $[0.74, 0.89]$.

The fraction of edges inside (some module), or FEI, depends somewhat on the clustering algorithm, but typically stays within a small range. Using the MCL algorithm (with or without inflation) gives rise to clusterings with very similar FEI values across the species, while the values for ClusterOne tend to be somewhat larger, but also within a small range. For instance, for the six species in S_{900} MCL_{def} gives FEI values in $[0.55, 0.75]$, $\text{MCL}_{1.8}$ in $[0.63, 0.81]$, and ClusterOne in $[0.64, 0.91]$. A similar pattern holds for HPRD and the MAGNA++ networks, but the values are much lower for the networks in HitPredict, possibly because HitPredict is good at excluding indirect interactions that simply shortcut paths through transitive closure.

In contrast, the Gini coefficient, while always fairly high,

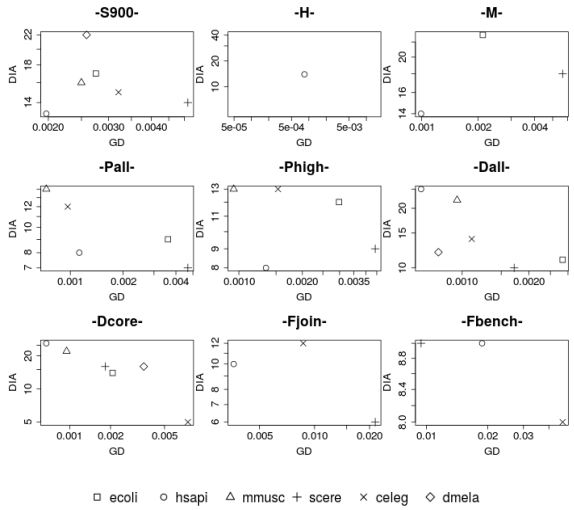

Figure 2: diameter over graph density, across all data sources.

shows a nearly uniform distribution between 0.5 and 1 across the instances: it is very high in S_{900}, around 0.8; in H and M around 0.7; in P: between 0.46 and 0.7. (Observe that the Gini coefficient changes only negligibly for the filtered networks: although a filtered network has fewer edges, the removal of edges also disconnects poorly connected nodes, which consequently disappear from the filtered network and thus no longer contribute "poor" individuals to the Gini computation.)

The diameter is assumed to anticorrelate with the graph density as Figure 2 supports, but of course it depends on the nature of the network structure provided by the source. Across databases and species, it lies in $[9,25]$. For some databases, there exists only little variance between the full and filtered set as in HitPredict and FunctionalNet: the full set P_a the diameter $\in [9,14]$ vs graph density $\in [0.0007,0.001]$, while in the filtered set P_h has diameter $\in [8.13]$ vs graph density $\in [0.0009,0.005]$; for FunctionalNet the diameter of F_b ($[8,9]$) is a subset of F_j ($[6,12]$). S_{900} seems to have relatively small variance in diameter $[13,22]$ with graph densities in $[0.002,0.005]$. Interestingly, in DIP the core data D_c show a larger variance in diameter $[5,26]$ than the full D_a with $[11,25]$ with a similar density range $[0.0006,0.003]$.

4.2 Modular PPI network structure

Given the very large number of data points here, our interest shifts from commonality in values to commonality in behavior with respect to simple variables such as cluster size. Here again, some similarities are apparent. For instance, Figure 3 plots on a log-log scale the histograms of three different basic attributes of modules computed by three different clustering algorithms from three different

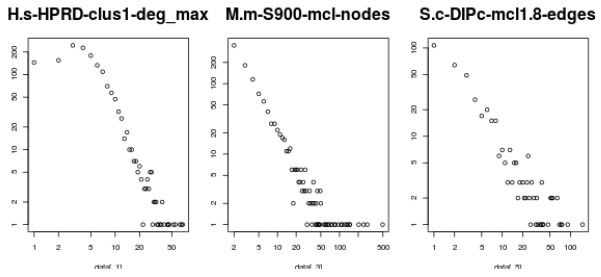

Figure 3: Histograms of the max degree, number of nodes, and number of edges all follow a power law.

databases for three different organisms, yet all clearly follow a power law. (The other possible histograms are all similar.) Once again, however, some measures do not show much commonality: the Gini coefficients for modules, while generally smaller than their corresponding value for the entire network, show no clear pattern, nor does graph density. For the distribution of the latter two features, see Figure 6 and 7 in Section 5.2.

5 Simulation Results and Comparison

Once we have identified common structural features in the PPI networks, we can use them as references in our second step. We compare them with comparable features produced by existing network models as well as by our NEMo model, in order to characterize how well these various models do in generating the type of structure and modularity observed in PPI networks. We let networks evolve under the commonly used 1-layer D&D model and our 2-layer, module-aware, NEMo model. We then subject the resulting networks to a global and modular feature analysis.

In a first test, we let the D&D model and our NEMo model start with a random network of roughly 500 nodes and run for 2'000 steps. The NEMo model reclusters the network after every 500 steps to update the decomposition into modules. Note that, while 2'000 steps run with D&D results in 2'000 evolutionary events, 2'000 steps run with NEMo can result in a different number of evolutionary events, depending on the parameters.

All networks are clustered at the end of the simulation with (1) MCL_{def}, (2) $\text{MCL}_{1.8}$, and (3) ClusterOne with minimum size of a module "1", a modular density of at least $\frac{1}{2}$ of the global density, and no penalty. We compare the values from the generated networks with the values from the database networks to assess their closeness. To investigate the impact of module-awareness in models, we run the NEMo simulations with two values of the parameter that controls the inter- vs intramodular exchange and evolution.

5.1 Global structure of simulated networks

The clustering coefficient was highlighted as one of the global measures that showed consistency across the PII networks in the databases. The NEMo networks, while producing values in the range of $[0.1, 0.15]$ that are lower than the database networks, come much closer than the D&D networks, which produce very small CC values in the range of $[0.0009, 0.01]$ and suffer from high variance.

The Gini coefficients, while varying without clear pattern, were consistently at or above 0.5 for the database networks. The NEMo networks produce smaller Gini values in a much tighter range around 0.4, while the D&D networks produce even smaller values in an even tighter range around 0.35.

D&D and NEMo both evolve networks with fairly high diameters compared to the PPI networks: the D&D networks have diameters in $[14, 21]$ (with one outlier) with graph densities in $[0.004, 0.007]$, while NEMo's have diameters in $[22, 31]$ with graph densities in $[0.001, 0.006]$, both features mostly anticorrelated to each other. With NEMo, diameters can double within one run, although reclustering the network during the process with MCL inflation parameter 1.8 reduces the diameter.

The main observation here is that both NEMo and DMC indeed show similar structure to the real-world PPI networks, although NEMo gets closer in most of the cases.

5.2 Module-aware simulation of network structure

The module-aware level of NEMo derives its power from its ability to distinguish intermodular from intramodular events. NEMo uses this power in a minimal way, by assigning slightly different probabilities to the two classes of events. The distinction between the two classes of events could be used to a much larger extent, but our results show that this minimal intervention, consistent with a slight selective pressure to preserve modularity while allowing modules themselves to adapt, suffices to create a significant difference in the types of networks produced.

We compare the plots of the maximum degree distribution of two randomly chosen D&D evolved networks, in Fig 4, with NEMo evolved ones, in Fig 5. In Fig 4 each row represents the same network sample, while each column represents the modular results reclustered with ClusterOne, MCL_{def}, and $\text{MCL}_{1.8}$, respectively. In the sequence of NEMo evolved networks as shown in Fig 5, a power law of modular edge distribution shapes up in the process, a trend that can also be observed in other modular feature distributions (node distribution, max degree distribution, etc.). Some observed distributions are less clearly shaped, such as those for the modular density and Gini coefficients, as shown in Fig. 6 (Gini) and Fig. 7 (modular density). Nevertheless, the distribution plots consistently show the same result: networks evolved using NEMo come closer to the database networks than those using D&D models.

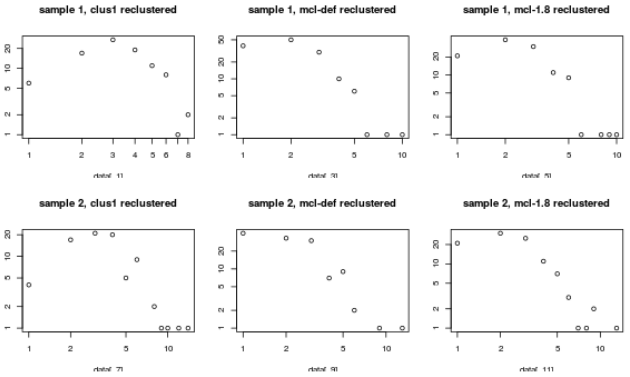

Figure 4: The modular max degree distribution of some D&D networks.

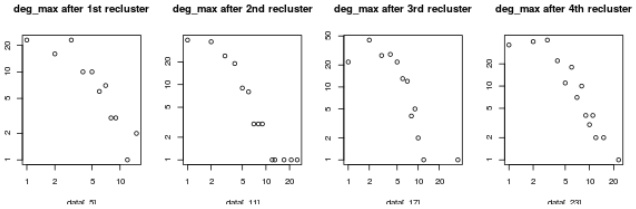

Figure 5: The modular edge distribution of NEMo networks evolves into a power law.

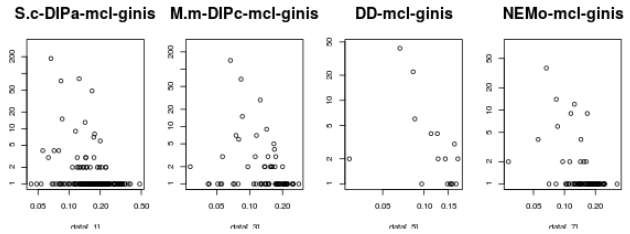

Figure 6: The modular Gini distribution for: (1) C.elegans in DIP_{all}, (2) H.sapiens in HPRD, (3) D&D sample network, (4) NEMo sample network.

Details

For simulations we run the D&D one-level model and the two-level NEMo with two parameter settings. The parameters can be grouped into three classes: the probabilities for a node duplication event with subsequent divergence (q_{con}, q_{mod}, and q_{new}), the general probabilities of an evolutionary event (p_{gain_n}, p_{loss_n}, p_{gain_e}, p_{loss_n}), and

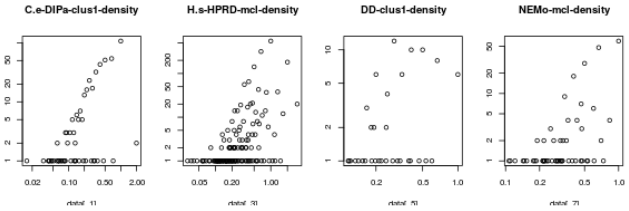

Figure 7: The modular density distribution for: (1) C.elegans in D_{all}, (2) H.sapiens in HPRD, (3) D&D sample network, (4) NEMo sample network.

the thresholds that determine whether, at a given step, there will be an intermodular or intramodular or no evolutionary event for a given module ($th_{intermod}$, $th_{intramod}$, th_{no})—these last of use only for NEMo. For NEMo, we use the same probabilities for the evolutionary events, but different values for the probabilities affecting modules to see whether and how the module-awareness affects the network's evolution and the resulting structure.

Since existing models other than NEMo are generative rather than evolutionary, no parameter values were suggested by past authors that would allow the network to evolve without an enforced growth in network size. Therefore, we adjusted values given in the literature, yielding the parameter settings in Table 3.

Table 3: Parameter settings

	setting 1 (D&D)	setting 2 (NEMo1)	setting 3 (NEMo2) (NEMo2)
q_{con}	0.1	0.1	0.1
q_{mod}	0.4	0.4	0.4
q_{new}	0.1	0.1	0.1
p_{gain_n}	0.31	0.25	0.25
p_{loss_n}	0.13	0.15	0.4
p_{gain_e}	0.26	0.3	0.3
p_{loss_e}	0.3	0.3	0.3
$th_{intermod}$	-	0.35	0.3
$th_{intramod}$	-	0.35	0.4
th_{no}	-	0.3	0.3

6 Conclusions

We studied in detail the PPI networks of six model species as found in six different public databases, looking for common structural features. Using a collection of measures at both the overall network level and the individual module level, we identified a number of such features, some easily captured in a single number (such the clustering coefficient) and others best presented through plots that demonstrate unmistakable power laws or uniform distributions. Remarkably, these features are shared across databases as well as across species, so that they can serve as reference points for the development of generative and evolutionary models for PPI networks. In that spirit, we tested a standard duplication and divergence (D&D) model, along with our own, module-aware, NEMo model, to ascertain how close these models come to reproducing the reference features extracted from PPI networks. Our results provide strong evidence that a suitable model needs to work at a more global level than individual nodes or edges, as NEMo easily outperformed the D&D models in these tests. Further work includes inverting the NEMo model for inference and parameterizing it to suit a particular organism so as to recover ancestral information.

References

[1] A. Abi-Haidar et al. Uncovering protein interaction in abstracts and text using a novel linear model and word proximity networks. *Genome Biol.*, 9 (Suppl 2)(S11), 2008.

[2] T. Aittokallio. Module finding approaches for protein interaction networks. In X.-L. Li and S.-K. Ng, eds., *Biological Data Mining in Protein Interaction Networks*, 335–353. 2009.

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

[4] A. Bhan, D.J. Galas, and T.G. Dewey. A duplication growth model of gene expression networks. *Bioinformatics*, 18(11):1486–1493, 2002.

[5] S. Brohée and J. van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, 7(1):488, 2006.

[6] S.R. Collins et al. Toward a comprehensive atlas of the physical interactive of saccharomyces cerevisiae. *Mol Cell Proteomics*, 6(3):439–450, 2007.

[7] M.T. Dittrich et al. Identifying functional modules in protein-protein interaction networks: an integrated exact approach. In *Proc. 16th Int'l Conf. on Intelligent Systems for Mol. Biol. (ISMB'08), in Bioinformatics* 24, i223–i231, 2008.

[8] S. Van Dongen. *Graph Clustering by Flow Simulation*. PhD thesis, U. of Utrecht, The Netherlands, 2000.

[9] J. Dutkowski and J. Tiuryn. Identification of functional modules from conserved ancestral proteinprotein interactions. *Bioinformatics*, 23(13):i149–i158, 2007.

[10] J. Dutkowski and J. Tiuryn. Phylogeny-guided interaction mapping in seven eukaryotes. *BMC Bioinformatics*, 10(1), 2009.

[11] A.J. Enright, S. Van Dongen, and C.A. Ozounis. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, 30(7):1575–1584, 2002.

[12] C. Gini. Measurement of inequality of incomes. *The Economic Journal*, 124–126, 1921.

[13] Y. Hao et al. Discovering patterns to extract protein-protein interactions from the literature. *Bioinformatics*, 21(15):3294–3300, 2005.

[14] L.H. Hartwell et al. From molecular to modular cell biology. *Nature*, 402(6761):C47–C52, 1999.

[15] Y. Jin et al. The evolutionary dynamics of protein-protein interaction networks inferred from the reconstruction of ancient networks. *PLoS ONE*, 8(3, e58134), 2013.

[16] I. Lee et al. Prioritizing candidate disease genes by network-based boosting of genome-wide association data. *Genome Research*, 21(7):1109–1121, 2011.

[17] I. Lee et al. A single network comprising the majority of genes accurately predicts the phenotypic effects of gene perturbation in c. elegans. *Nature Genetics*, 40:181–188, 2008.

[18] I. Lee et al. Predicting genetic modifier loci using functional gene networks. *Genome Research*, 20(8):1143–1153, 2010.

[19] I. Lee, Z. Li, and E.M. Marcotte. An improved, bias-reduced probabilistic functional gene network of baker's yeast, saccharomyces cerevisiae. *PLoS ONE*, 2(10), 2007.

[20] Y. Lopez, K. Nakai, and A. Patil. Hitpredict version 4 - comprehensive reliability scoring of physical protein-protein interactions from more than 100 species. *Database: The Journal of Biological Databases and Curation*, 2015.

[21] M. Lynch et al. The evolutionary fate and consequences of duplicate genes. *Science*, 290(5494):1151–1254, 2000.

[22] T. Makino and A. McLysaght. Evolutionary analyses of protein interaction networks. In X.-L. Li and S.-K. Ng, editors, *Biological Data Mining in Protein Interaction Networks*, 169–181. 2009.

[23] E.M. Marcotte, I. Xenarios1, and D. Eisenberg. Mining literature for proteinprotein interactio. *Bioinformatics*, 17:359–363, 2001.

[24] M. Middendorf, E. Ziv, and C.H. Wiggins. Inferring network mechanisms: the drosophila melanogaster protein interaction network. *Proc. Nat'l Acad. Sci., USA*, 102(9):3192–3197, 2005.

[25] J.H. Morris et al. Affinity purification–mass spectrometry and network analysis to understand protein-protein interactions. *Nature Protocols*, 9(11):2539–2554, 2014.

[26] S. Navlakha and C. Kingsford. Network archaeology: Uncovering ancient networks from present-day interactions. *PLoS Comput. Biol.*, 7(4, e1001119), 2011.

[27] T. Nepusz, H. Yu, and A. Paccanaro. Detecting overlapping protein complexes in protein-protein interaction networks. *Nature Methods*, 9:471–472, 2012.

[28] S. Ohno. *Evolution by Gene Duplication*. Springer Verlag, Berlin, 1970.

[29] A. Patil, K. Nakai, and H. Nakamura. Hitpredict: a database of quality-assessed protein-protein interactions in nine species. *Nucleic Acids Research*, D744–D749, 2015.

[30] J.M. Peregrin-Alvarez et al. The modular organisation of protein interactions in escherichia coli. *PLoS Computational Biology*, 2009.

[31] T.S.K. Prasad et al. Human protein reference database–2009 update. *Nucleic Acids Research*, 37:D767–772, 2009.

[32] J. Qian, N.M. Luscombe, and M. Gerstein. Protein family and fold occurrence in genomes: powerlaw behaviour and evolutionary model. *J. Mol. Biol.*, 313:673–689, 2001.

[33] P. Radivojac et al. An integrated approach to inferring gene-disease associations in humans. *Proteins*, 72(3):1030–1037, 2008.

[34] S.M.E. Sahraeian and B.-J. Yoon. A network synthesis model for generating protein interaction network families. *PLoS ONE*, 7(8, e41474), 2012.

[35] L. Salwinski et al. The database of interacting proteins: 2004 update. *Nucleic Acids Research*, D449–D551, 2004.

[36] V. Saraph and T. Milenkovi. Magna: Maximizing accuracy in global network alignment. *Bioinformatics*, 30(20):2931–2940, 2014.

[37] G. Schlosser and G.P. Wagner. *Modularity in Development and Evolution*. U. Chicago Press, 2004.

[38] R.V. Sole et al. A model of large-scale proteome evolution. *Advances in Complex Systems*, 5:43–54, 2002.

[39] R.V. Sole and S. Valverde. Spontaneous emergence of modularity in cellular networks. *J. Royal Society Interface*, 5(18):129–134, 2008.

[40] D. Szklarczyk et al. String v10: protein-protein interaction networks, integrated over the tree of life. *Nucleic Acids Res.*, 43:D447–D452, 2015.

[41] S. van Dongen and C. Abreu-Goodger. Using MCL to extract clusters from networks. In J. van Helden, A.. Toussaint, and D. Thieffry, editors, *Bacterial Molecular Networks*, volume 804 of *Methods in Mol. Biol.*, 281–295. Springer Verlag, Berlin, 2012.

[42] A. Vazquez et al. Global protein function prediction from protein-protein interaction networks. *Nature Biotech.*, 21(6):697–700, 2003.

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

[44] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, UK, 1994.

[45] M. Ye et al. NEMo: An evolutionary model with modularity for PPI networks. *Proc. 12th Int'l Symp. Bioinf. Research & Appls. ISBRA'16*. In Lecture Notes in Computer Science 9683, 224-236 (2016).

[46] X. Zhang and B.M.E. Moret. Refining transcriptional regulatory networks using network evolutionary models and gene histories. *Algorithms for Mol. Biol.*, 5(1), 2010.

[47] X. Zhang and B.M.E. Moret. Refining regulatory networks through phylogenetic transfer of information. *ACM/IEEE Trans. on Comput. Biol. & Bioinf.*, 9(4):1032–1045, 2012.