Metadata of the chapter that will be visualized in SpringerLink

Book Title	Bioinformatics Research and Applications	
Series Title		
Chapter Title	NEMo: An Evolutionary Model with Modularity for PPI Networks	
Copyright Year	2016	
Copyright HolderName	Springer International Publishing Switzerland	
Author	Family Name	**Ye**
	Particle	
	Given Name	**Min**
	Prefix	
	Suffix	
	Division	School of Computer and Communication Sciences
	Organization	EPFL
	Address	Lausanne, Switzerland
	Email	min.ye@epfl.ch
Author	Family Name	**Racz**
	Particle	
	Given Name	**Gabriela C.**
	Prefix	
	Suffix	
	Division	Department of Mathematics
	Organization	University of Zagreb
	Address	Zagreb, Croatia
	Email	
Author	Family Name	**Jiang**
	Particle	
	Given Name	**Qijia**
	Prefix	
	Suffix	
	Division	Department of Electrical and Computer Engineering
	Organization	Stanford University
	Address	Palo Alto, USA
	Email	
Author	Family Name	**Zhang**
	Particle	
	Given Name	**Xiuwei**
	Prefix	
	Suffix	
	Division	
	Organization	European Bioinformatics Institute (EMBL-EBI)
	Address	Cambridge, UK

	Email	
Corresponding Author	Family Name	**Moret**
	Particle	
	Given Name	**Bernard M. E.**
	Prefix	
	Suffix	
	Division	School of Computer and Communication Sciences
	Organization	EPFL
	Address	Lausanne, Switzerland
	Email	bernard.moret@epfl.ch

Abstract

Modelling the evolution of biological networks is a major challenge. Biological networks are usually represented as graphs; evolutionary events include addition and removal of vertices and edges, but also duplication of vertices and their associated edges. Since duplication is viewed as a primary driver of genomic evolution, recent work has focused on duplication-based models. Missing from these models is any embodiment of modularity, a widely accepted attribute of biological networks. Some models spontaneously generate modular structures, but none is known to maintain and evolve them.
We describe NEMo (Network Evolution with Modularity), a new model that embodies modularity. NEMo allows modules to emerge and vanish, to fission and merge, all driven by the underlying edge-level events using a duplication-based process. We introduce measures to compare biological networks in terms of their modular structure and use them to compare NEMo and existing duplication-based models and to compare both generated and published networks.

Keywords (separated by '-')

Generative model - Evolutionary model - PPI network - Evolutionary event - Modularity - Network topology

NEMo: An Evolutionary Model with Modularity for PPI Networks

Min Ye[1], Gabriela C. Racz[2], Qijia Jiang[3], Xiuwei Zhang[4], and Bernard M.E. Moret[1(✉)]

[1] School of Computer and Communication Sciences, EPFL, Lausanne, Switzerland
{min.ye,bernard.moret}@epfl.ch
[2] Department of Mathematics, University of Zagreb, Zagreb, Croatia
[3] Department of Electrical and Computer Engineering, Stanford University, Palo Alto, USA
[4] European Bioinformatics Institute (EMBL-EBI), Cambridge, UK

Abstract. Modelling the evolution of biological networks is a major challenge. Biological networks are usually represented as graphs; evolutionary events include addition and removal of vertices and edges, but also duplication of vertices and their associated edges. Since duplication is viewed as a primary driver of genomic evolution, recent work has focused on duplication-based models. Missing from these models is any embodiment of modularity, a widely accepted attribute of biological networks. Some models spontaneously generate modular structures, but none is known to maintain and evolve them.

We describe NEMo (Network Evolution with Modularity), a new model that embodies modularity. NEMo allows modules to emerge and vanish, to fission and merge, all driven by the underlying edge-level events using a duplication-based process. We introduce measures to compare biological networks in terms of their modular structure and use them to compare NEMo and existing duplication-based models and to compare both generated and published networks.

AQ1

Keywords: Generative model · Evolutionary model · PPI network · Evolutionary event · Modularity · Network topology

1 Introduction

The rapid growth of experimentally measured data in biology requires effective computational models to uncover biological mechanisms in the data. Networks are commonly used to represent key processes in biology; examples include transcriptional regulatory networks, protein-protein interaction (PPI) networks, metabolic networks, etc. The model is typically a graph, directed or undirected, where edges or arcs represent interactions and vertices represent actors (genes, proteins, etc.). Establishing experimentally the existence of a particular interaction is expensive and time-consuming, so most published networks have been inferred through computational methods ranging from datamining the literature

A. Bourgeois et al. (Eds.): ISBRA 2016, LNBI 9683, pp. 1–13, 2016.
DOI: 10.1007/978-3-319-38782-6_19

(see, e.g., [1,10,15]) to inferring the evolutionary history of the networks from present observations [8,26,35,36]. (Makino and McLaughlin [14] present a thorough discussion of evolutionary approaches to PPI networks.) Often these networks are built through a process of accretion, by adding new actors and new interactions as they are observed, published, or inferred, with the result that errors in many current biological networks tend to be false positives (errors of commission) rather than false negatives (errors of omission). A variety of databases store inferred networks and range from large graphs, such as the human PPI network in the STRING database (ca. 3'000'000 interactions) [9,31], down to quite small ones, such as the manually curated Human Protein Reference Database (ca. 40'000 interactions) [23]. Even a cursory reading of the literature shows that agreement among findings is rather limited, due in part to the variety of samples used and the dynamic nature of the networks, but also in good part because of the difficulty of inference.

This intrinsic difficulty has led researchers to go beyond the inference of a single network from data about one organism and to use comparative methods. Pairwise comparative methods, while more powerful, offer only limited protection against noise and high variability. This weakness in turn has led to the use of evolutionary methods that use several organisms and carry out simultaneous inference on all of them [8,14,36]—a type of inference that falls within the category of transfer learning [20]. A unique feature in these approaches is their use of evolutionary models (not commonly associated with transfer learning). These approaches posit a model of evolution for the networks, typically based on inserting and deleting edges and duplicating or losing vertices, and then seek to infer present-day networks as well as ancestral networks that, under the chosen evolutionary model, would best explain the data collected. The evolutionary model is thus the crucial component of the inference procedure.

An early finding about biological networks such as regulatory networks and PPI networks was the clear presence of modularity [11]: these networks are not homogeneous, with comparable connectivity patterns at every vertex, but present a higher-level structure consisting of well connected subgraphs with less substantial connectivity to other such subgraphs. Modularity is now widely viewed as one of the main characteristics of living systems [28]. While some of the models devised for networks lead automatically to the emergence of modules within the network [30], these models are purely generative—increasing the size of the network at each step—and thus do not match biological reality. There is thus a need for an evolutionary model for PPI networks that, while still based on the gain and loss of vertices and edges, takes into account modularity.

In this paper, we introduce NEMo, a network evolutionary model with modularity for PPI networks that includes both growth and reduction operators, and that explicitly models the influence of modularity on network evolution. While modules remain the product of purely local events (at the level of single vertices or edges), they are subject to slightly different selection constraints once they have emerged, so that our model allows modules to emerge, to disappear, to merge, and to split. We present the results of simulations and compare the networks thus produced to the consensus networks currently stored in a variety

of databases for model organisms. Our comparisons are based on both network alignment ideas and new measures aimed at quantifying modularity, so we also discuss the usefulness of these measures and evaluate published PPI networks with respect to these measures. Our measures of modularity can be used to analyze the general characteristics of PPI networks and clearly distinguish the various models organisms. Our findings support the accepted bias of published networks towards false positives and the often reported distribution of modules into a few large subgraphs and a collection of much smaller subgraphs; NEMo produces networks with the latter characteristic and maintains it even when it has reached a target range of sizes and simply makes small changes to the structure of the network.

2 Current Generative Models for PPI Networks

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). In terms of complexity and verisimilitude, however, models proposed to date vary widely. Most of the recent models are based on duplication followed by divergence, denoted D&D [4,24], in which a vertex is duplicated (think of a gene duplication) and inherits some randomly chosen subset of the connections of the original vertex (the copy of the gene initially produces much the same protein as the original and so enters into much the same interactions). 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 [13,19], so models based on D&D have become widely used for PPI networks.

The full D&D model considers both specialization and gene duplication events. Following a specialization 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. A recent variation on the D&D model is the duplication-mutation-complementarity (DC) model [16,17,32], in which the same interaction cannot be lost simultaneously in the original and in the copy and in which the duplicated gene itself may gain a direct interaction with the original gene. The DMR (random mutation) model [29] is another variation, in which new interactions (not among those involving the original vertex) can be introduced between the duplicate vertex and some random vertex in the network.

3 NEMo

While, as noted earlier, the D&D model (and, by extension, its various derivatives) will automatically give rise to modular structures, it does so 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. In that sense, the D&D, while a generative model, is not an evolutionary model: it can only grow networks, not evolve them while keeping their size within some fixed range. The same is true of its several variants.

Our aim is to produce a generative model that is also an evolutionary model and that we can later use for reconstructing the evolutionary history of PPI networks. Under such a model, a network may grow, shrink, or, most commonly, vary in size within some bounded range. Since the dominant growth operator is duplication and since this operator typically adds multiple edges to the network, random (i.e., unrelated to other events) loss of edges must be common. We conjecture that, under such a model, modularity might not be preserved—because, under such a model, the selection of which interactions to lose is independent of the modular structure. Since modules appear both necessary to life and quite robust against mutations, a model of evolution of PPI networks that is biased (as nature appears to be) in favor of the survival of modules would need to "know" about the module structure.

We therefore decided to design a two-level model. The lower level is just a variant of the DC model, except that it allows random mutations for each vertex—a vertex can be lost at any step rather than just when its degree is reduced to zero—and that, due to the same random mutations, arbitrary edges can be added to or removed from the network. The higher level, however, is "module-aware" so that interactions can be classified as within a module, between modules, or unrelated to modules. This classification allows us to treat these three types differently in the evolutionary model, with interactions within modules less likely to be lost. Our model represents a PPI network as a graph, with the set of vertices representing proteins and the set of undirected edges representing undirected interactions between the proteins. In addition, the graph is at all times subdivided into subgraphs, which correspond to modules.

The events directly affecting vertices and edges are similar to those of the D&D model and its relatives and can be classified into four categories: protein gain, protein loss, interaction gain, and interaction loss. Protein gain is exclusively through duplication and thus also includes interaction gains for the newly added protein. Protein loss removes a randomly chosen vertex; it can be a consequence of, e.g., pseudogene formation. (As in the DMC model, it is also possible to lose a vertex through progressive loss of interactions until the vertex has degree zero.) Interaction loss removes a randomly chosen edge; it can come about through domain mutations, structural mutations, subfunctionalization, and the like. Interaction gains come in two varieties: those caused by vertex duplication and those arising purely at random, by connecting a previously unconnected pair of vertices, which could arise, like loss, through domain or structural mutations, or through progressive neofunctionalization (Fig. 1).

AQ2

We use the module level to influence the event chain as follows. First, we allow events to arise within the same time frame in different modules; whereas existing models treat the network as one unit and allow a single event at a time, our model treats the network as a collection of subgraphs and allows up to one event in each subgraph. Multiple events within the same time frame can

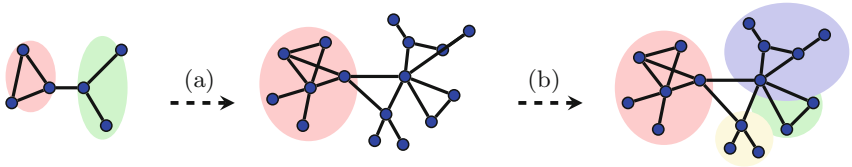


Fig. 1. NEMo: schema of the evolutionary process: (a) after multiple timesteps; (b) after reclustering

more closely model events such as segmental duplication (in which many genes are duplicated together). Second, we distinguish intramodular events (all four events can be intramodular) from intermodular events (only edge gains and losses can be intermodular), allowing us to use different parameters for the two types. While we automatically place a duplicate vertex within the same module as the original vertex, we also periodically recompute the subgraph decomposition, thereby inducing changes in the module structure, including both fission and fusion of modules. Thus there is no specific evolutionary event associated with changes in the module structure: rather, it is a recognition that underlying events have, over some number of steps, sufficiently altered the network as to have altered, destroyed, or created some modules.

Such a model as this 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 [2,6,7] 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 ClusterOne [18], a graph clustering algorithm that allows overlapping clusters. It has been useful for detecting protein complexes in PPI networks tolerating nodes to have multiple-module membership.

4 Assessing Modularity

In order to evaluate the output of NEMo, we must first quantify significant attributes of PPI networks. The resulting features can then be used to measure the similarity of our generated networks to real networks, as well as the differences between networks generated by our model and networks generated under existing models. 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. We thus propose a set of features applicable to hall networks, features chosen to measure global properties of networks and to quantify aspects of modularity.

Most of the features proposed here are commonly used in the analysis of networks [2,3]; several are modified so as to provide a level of independence

from size—bacterial PPI networks are necessarily smaller than mammalian PPI networks, while simulations can be run at all sizes. For each network, we compute the number of nodes, the number of edges, and the degree distribution; we also run the ClusterOne cluster algorithm (always with the same parameters) and store the number of clusters as well as the size and composition of each cluster. We then compute the following five global measures.

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 [34].

Graph Density (GD): The density of a graph is the ratio of the actual number of edges to the number of possible edges.

Diameter (⊘): The diameter of a graph is the length of the longest simple path in the graph.

Fraction of Edges Inside (FEI): FEI is the fraction of edges contained within modules. We expect it to be high since PPI networks contain highly connected substructures (modules) that have only few connections to vertices outside the substructure [3,12,33].

Tail Size (TS): A simple representation of the tail of the degree distribution, TS is fraction of the number of nodes with degree higher than one-third of that maximum node degree.

5 Results on Natural PPI Networks

For the data, we chose to work with model organisms, as they have large numbers of high-confidence interactions. We chose to download the following species since they have the largest number of well documented interactions: *E. Coli*, *S. Cerevisiae*, and *H. Sapiens*. Different sources were considered to emphasize the discrepancies of the networks stored and provided in existing datasets of real world PPI networks.

One source is the STRING database [9] that aims to provide a global perspective for as many organisms as feasible, tolerating lower-quality data and computational predictions. With this purpose the database holds a large part of false positive interactions. Although the STRING database stores evidence scores for each protein-protein interaction to allow elimination of as many false positive entries as possible by the user, it is still very much biased. For other sources, we consulted the manually curated *H. sapiens* PPI network HPRD [22] database and the experimental setup of the MAGNA++ algorithm [27] that aims at maximizing accuracy in global network alignment: an *H. sapiens* PPI network of 9141 proteins and 41456 interactions (Radivojac et al., 2008 [25]), an *E. coli* PPI network [21] of high-confidence of 1941 proteins with 3989 interactions, and a yeast *S. cerevisiae* PPI network with 2390 proteins and 161277 PPIs (Collins et al., 2007 [5]).

Table 1. General characteristics of the three PPI networks in various databases

Species	# nodes	# edges	# clusters
H. sapiens STRING	19247	4274001	2077
E. coli STRING	4145	568789	16
S. cerevisiae STRING	6418	939998	159
H. sapiens HPRD	9673	39198	2886
E. coli MAGNA++exp	1941	3989	393
S. cerevisiae MAGNA++exp	2390	16127	360
H. sapiens MAGNA++exp	9141	41456	2306

Table 2. Values of our features for the three PPI networks in various versions

Species	CC	GD	⊘	FEI	TS
H. sapiens STRING	0.23058	0.02308	18	0.94506	0.99777
E. coli STRING	0.21368	0.06623	9	1.00942*	0.80555
S. cerevisiae STRING	0.27757	0.04565	20	1.08949*	0.99564
H. sapiens HPRD	0.19602	0.00084	30	0.53896	0.99369
E. coli MAGNA++exp	0.3394	0.00212	33	0.92454	0.98454
S. cerevisiae MAGNA++exp	0.43854	0.00565	34	0.97055	0.95105
H. sapiens MAGNA++exp	0.16377	0.00099	30	0.56549	0.99103

* (FEI > 1) comes from the multiple membership of nodes. Edges shared by two nodes that belong to more than one same module are counted more than once.

We downloaded PPI networks from the STRING database [31] and used a high threshold (900) on the supplied confidence scores to retain only high-confidence interactions. Table 1 provides a brief description of these three PPI networks in the various databases and versions.

We then computed our network features for each of these networks, as shown in Table 2.

6 Results on Simulations

6.1 Simulation Goals and Setup

The goal of our simulations was to verify the ability of NEMo to produce networks with characteristics similar to those of the natural PPI networks and also to compare the networks it produces with those produced without the module-aware level and with those produced by D&D models. In particular, we wanted to test the ability of NEMo to sustain modules in networks not undergoing growth, but subject only to change—where gain of proteins and interactions is balanced by loss of same. Therefore we ran two distinct series of simulations, one for generation and one for evolution.

The first series uses both the DMC model [32], perhaps the most commonly used model in the D&D family today, and NEMo to grow networks to fixed sizes. We then compute our features on these networks and compare both types of generated networks with the PPI networks of the model organisms. Since DC is not module-aware, but claimed to generate modular networks, whereas NEMo is explicitly module-aware, we want to see how well the characteristics of each type of generated network compare to the PPI networks of the model organisms.

In the second series of simulations, we use NEMo in steady-state mode (balanced gains and losses) over many steps to evolve networks produced during the first simulation series. Our main intent here is to observe the evolution (mostly in terms of size, edge density, and modules) of the networks. We use parameters for NEMo that give it a slight bias towards growth, mostly to avoid the natural variance of the process from "starving" too many of the networks.

6.2 Results for Network Generation

We set parameters of our model for simulating growth of the network and compared the resulting networks with those built with the standard DMC model for similar sizes, as well as with the PPI networks from the three model organisms.

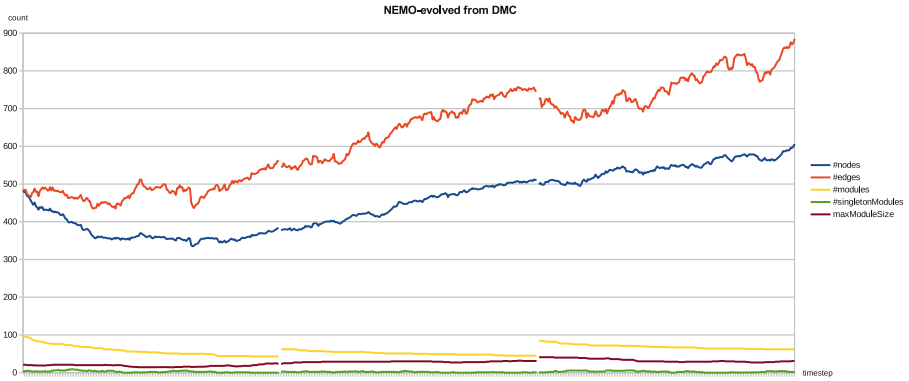
We then computed our network features for each of these networks, as shown in Table 3, where they can be compared to the same features shown for PPI networks (from Table 2). Both DC and NEMo generated networks with features comparable to those observed in the PPI networks collected from MAGNA and HPRD, although the significantly lower clustering coefficient of the DC-generated network (0.04 as compared to 0.14 for the NEMo-generated network) indicates a less resolved modular structure. Note that all PPI networks from databases have larger clustering coefficients than the generated networks, a difference attributable in good part to the generation mode.

Table 3. Values of our features for the generated networks and the three PPI networks in various versions

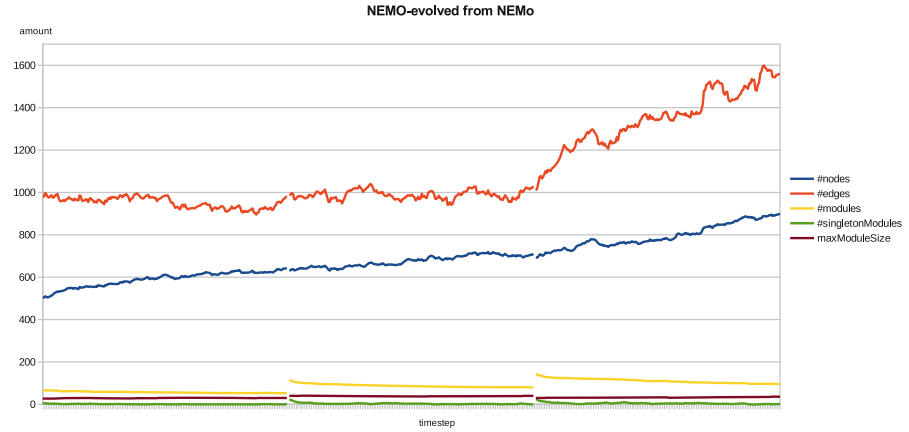
Species	CC	GD	⊘	FEI	TS
H. sapiens HPRD	0.196021	0.000837947	30	0.538956	0.993694
E. coli MAGNA++exp	0.3394	0.00211869	33	0.924542	0.984544
S. cerevisiae MAGNA++exp	0.438538	0.00564897	34	0.970546	0.951046
H. sapiens MAGNA++exp	0.163768	0.000992379	30	0.565491	0.991029
DC-generated net500	0.0478	0.0040	12	0.9520	0.9880
NEMO-generated net500	0.1417	0.0078	31	0.9559	0.9519

6.3 Results for Network Evolution

In the second step of our experiments we test the ability of NEMo to simulate the evolution of a PPI network (with roughly balanced ngain and loss rates)



(a) evolution from a DMC-generated initial network



(b) evolution from a NEMo-generated initial network

Fig. 2. Evolution of network characteristics under the NEMo model over 600 steps, with reclustering into modules at 200 and 400 steps. Top line shows the total number of edges, second line the number of vertices, third line the number of modules, fourth line the size of the largest module, and bottom line the number of singleton modules.

while preserving modularity and also test how NEMo's behavior is affected by its initial condition by using for DMC- and NEMo-generated networks at time zero. Figure 2 shows the changes in network size (numbers of edges and vertices) and structure (numbers of modules) as an initial n-etwork is evolved through 600 steps, with reclustering into modules taking place after 200 and 400 steps.

The main observation here is that NEMo, when started with a DMC-generated network (part (a) of the figure), begins by reconfiguring the network, reducing its number of vertices by about one third over the first hundred steps and replacing edges. It then moves into much the same mode as depicted in

part (b) of the figure, which shows a steady evolutionary behavior mixed with a small bias towards growth. The implication is that, while the DC-generated network may have a modular structure, that structure is less well structured (as observed above) as well as not well supported under the evolutionary model. We can observe that the graph density of the DMC-generated network is low and gets swiftly increased by NEMo, while the initial number of modules is high and gets switfly decreased by NEMo as a consequence of the removal of many nodes. After the first 200 steps and the first reclustering of modules, the evolution follows the same path as that followed immediately when working from a NEMo-generated intial graph, as seen in part (b) of the figure. Part (b) shows variance in the rate of increase in the number of edges, partly a consequence of the node duplication process—duplicating a few high-degree nodes in rapid succession quickly increases the overall degree of the network, while also increasing the number of high-degree nodes.

The mild generative bias we deliberately introduced into the evolutionary simulations can be harmlessly removed for evolving NEMo-generated networks and, through larger numbers of steps, evolving a modular structure closer to that of the PPI networks from the databases.

It is worth noting that the module-aware level of NEMo is very limited in its effects: its power derives from its distinguishing intermodular from intramodular events, but NEMo uses this power in quite a minimal way, by assigning slightly different probabilities to the two classes of events—in evolutionary terms, it simulates a slightly stronger negative selection for intermodular events than for intramodular events. The distinction between the two classes of events could be used to a much larger extent, but our results show that even this minimal intervention, consistent with a selective pressure to preserve modularity while allowing modules themselves to adapt, suffices to create a significant difference in the type of networks generated.

7 Discussion and Future Work

We presented NEMo, a module-aware evolutionary model for PPI networks. The emphasis of NEMo, as compared to existing models for PPI networks, is on evolution rather than generation: whereas existing models (and the first layer of NEMo, which is a variant of existing models) are know to generate modularity when growing networks, we were interested in a model that would evolve existing networks, using the same basic set of evolutionary events.

The salient feature of NEMo is a module-aware layer that sits above the event layer and distinguishes between intermodular and intramodular events. The awareness is achieved through periodic recomputation (triggered by sampling and analysis for drift) of the modular structure. The uses to which this awareness are put are minimal: NEMo simply gives a slightly higher probability to intramodular events than to intermodular events, thereby slightly favoring conservation of modules and evolution of internal module structure. The details of the model are broadly adjustable: the algorithm used to detect modules, the

number and nature of parameters used to control intra- vs. intermodular events, the features chosen to characterize the network, and the distance measure used to measure drift in order to decide when to re-evaluate the composition of modules, are all flexible.

Our simulation results show that this second layer enables NEMo to run through large numbers (as compared to the size of the network) of evolutionary events, balanced so as not to affect the expected size of the network, while preserving the characteristics of its original (growth-derived) modular structure. To the best of our knowledge, this is the first such result and it paves the way for phylogenetic analyses as well as population studies of PPI networks.

As discussed by Makino and McLaughlin [14], however, the number of factors that could affect the evolution of PPI networks is very large. NEMo captures only a small subset of these factors, since it works just on the graph structure and, at the level of individual events, makes the same independence assumptions as current models. Interdependent events or hidden underlying events present serious challenges. Incorporating externally supplied data (in addition to the network itself) makes sense in a data-rich era, but will require, for each type of data, further development of the model.

Acknowledgments. MY wishes to thank Mingfu Shao for many helpful discussions.

References

1. Abi-Haidar, A., 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. Aittokallio, T.: Module finding approaches for protein interaction networks. In: Li, X.L., Ng, S.K. (eds.) Biological Data Mining in Protein Interaction Networks, pp. 335–353. IGI Publishing, Hershey (2009)
3. Barabási, A.L., Oltvai, Z.: Network biology: understanding the cell's functional organization. Nat. Rev. Genet. **5**, 101–113 (2004)
4. Bhan, A., Galas, D., Dewey, T.: A duplication growth model of gene expression networks. Bioinformatics **18**(11), 1486–1493 (2002)
5. Collins, S., Kemmeren, P., Zhao, X., et al.: Toward a comprehensive atlas of the physical interactive of saccharomyces cerevisiae. Mol. Cell. Proteomics **6**(3), 439–450 (2007)
6. Dittrich, M., et al.: Identifying functional modules in protein-protein interaction networks: an integrated exact approach. In: Proceedings of 16th International Conference on Intelligent Systems for Molecular Biology (ISMB 2008), in Bioinformatics, vol. 24, pp. i223–i231 (2008)
7. Dutkowski, J., Tiuryn, J.: Identification of functional modules from conserved ancestral protein interactions. Bioinformatics **23**(13), i149–i158 (2007)
8. Dutkowski, J., Tiuryn, J.: Phylogeny-guided interaction mapping in seven eukaryotes. BMC Bioinform. **10**(1), 393 (2009)
9. Franceschini, A., et al.: String v9.1: protein-protein interaction networks, with increased coverage and integration. Nucleic Acids Res. **41**, D808–D815 (2013)

10. Hao, Y., Zhu, X., Huang, M., Li, M.: Discovering patterns to extract protein-protein interactions from the literature. Bioinformatics **21**(15), 3294–3300 (2005)
11. Hartwell, L., Hopfield, J., Leibler, S., Murray, A.: From molecular to modular cell biology. Nature **402**(6761), C47–C52 (1999)
12. Jin, Y., Turaev, D., Weinmaier, T., Rattei, T., Makse, H.: The evolutionary dynamics of protein-protein interaction networks inferred from the reconstruction of ancient networks. PLoS ONE **8**(3), e58134 (2013)
13. Lynch, M., et al.: The evolutionary fate and consequences of duplicate genes. Science **290**(5494), 1151–1254 (2000)
14. Makino, T., McLaughlin, A.: Evolutionary analyses of protein interaction networks. In: Li, X.L., Ng, S.K. (eds.) Biological Data Mining in Protein Interaction Networks, pp. 169–181. IGI Publishing, Hershey (2009)
15. Marcotte, E., Xenarios, I., Eisenberg, D.: Mining literature for protein protein interactions. Bioinformatics **17**, 359–363 (2001)
16. Middendorf, M., Ziv, E., Wiggins, C.: Inferring network mechanisms: the drosophila melanogaster protein interaction network. Proc. Nat. Acad. Sci. USA **102**(9), 3192–3197 (2005)
17. Navlakha, S., Kingsford, C.: Network archaeology: uncovering ancient networks from present-day interactions. PLoS Comput. Biol. **7**(4), e1001119 (2011)
18. Nepusz, T., Yu, H., Paccanaro, A.: Detecting overlapping protein complexes in protein-protein interaction networks. Nat. Methods **9**, 471–472 (2012)
19. Ohno, S.: Evolution by Gene Duplication. Springer, Berlin (1970)
20. Pan, S., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**, 1345–1359 (2010)
21. Peregrin-Alvarez, J., et al.: The modular organisation of protein interactions in escherichia coli. PLoS Comput. Biol. **5**(10), e1000523 (2009)
22. Prasad, T.S.K., et al.: Human protein reference database-2009 update. Nucleic Acids Res. **37**, D767–D772 (2009)
23. Prasad, T., et al.: The human protein reference database-2009 update. Nucleic Acids Res. **37**, D767–D772 (2009)
24. Qian, J., Luscombe, N., Gerstein, M.: Protein family and fold occurrence in genomes: powerlaw behaviour and evolutionary model. J. Mol. Biol. **313**, 673–689 (2001)
25. Radivojac, P., Peng, K., Clark, W., et al.: An integrated approach to inferring gene-disease associations in humans. Proteins **72**(3), 1030–1037 (2008)
26. Sahraeian, S., Yoon, B.J.: A network synthesis model for generating protein interaction network families. PLoS ONE **7**(8), e41474 (2012)
27. Saraph, V., Milenkovi, T.: Magna: maximizing accuracy in global network alignment. Bioinformatics **30**(20), 2931–2940 (2014). http://bioinformatics.oxfordjournals.org/content/30/20/2931.abstract
28. Schlosser, G., Wagner, G.: Modularity in Development and Evolution. University of Chicago Press, Chicago (2004)
29. Sole, R., Pastor-Satorras, R., Smith, E., Kepler, T.: A model of large-scale proteome evolution. Adv. Complex Syst. **5**, 43–54 (2002)
30. Sole, R., Valverde, S.: Spontaneous emergence of modularity in cellular networks. J. R. Soc. Interface **5**(18), 129–133 (2008)
31. Szklarczyk, D., et al.: String v10: protein-protein interaction networks, integrated over the tree of life. Nucleic Acids Res. **43**, D447–D452 (2015)
32. Vazquez, A., Flammini, A., Maritan, A., Vespignani, A.: Global protein function prediction from protein-protein interaction networks. Nat. Biotech. **21**(6), 697–700 (2003)

33. Wagner, A.: The yeast protein interaction network evolves rapidly and contains few redundant duplicate genes. Mol. Biol. Evol. **18**, 1283–1292 (2001)
34. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Cambridge University Press, UK (1994)
35. Zhang, X., Moret, B.: Refining transcriptional regulatory networks using network evolutionary models and gene histories. Algorithms Mol. Biol. **5**(1), 1 (2010)
36. Zhang, X., Moret, B.: Refining regulatory networks through phylogenetic transfer of information. ACM/IEEE Trans. Comput. Biol. Bioinf. **9**(4), 1032–1045 (2012)

Author Queries

Chapter 19

Query Refs.	Details Required	Author's response
AQ1	Please confirm if the corresponding author is correctly identified. Amend if necessary.	
AQ2	Please check and confirm if the inserted citation of Fig. 1 is correct. If not, please suggest an alternate citation.	

MARKED PROOF

Please correct and return this set

Please use the proof correction marks shown below for all alterations and corrections. If you wish to return your proof by fax you should ensure that all amendments are written clearly in dark ink and are made well within the page margins.

Instruction to printer	*Textual mark*	*Marginal mark*
Leave unchanged	··· under matter to remain	
Insert in text the matter indicated in the margin		New matter followed by or
Delete	through single character, rule or underline or through all characters to be deleted	or
Substitute character or substitute part of one or more word(s)	through letter or through characters	new character or new characters
Change to italics	under matter to be changed	
Change to capitals	under matter to be changed	
Change to small capitals	under matter to be changed	
Change to bold type	under matter to be changed	
Change to bold italic	under matter to be changed	
Change to lower case	Encircle matter to be changed	
Change italic to upright type	(As above)	
Change bold to non-bold type	(As above)	
Insert 'superior' character	through character or where required	or under character e.g. or
Insert 'inferior' character	(As above)	over character e.g.
Insert full stop	(As above)	
Insert comma	(As above)	
Insert single quotation marks	(As above)	or and/or or
Insert double quotation marks	(As above)	or and/or or
Insert hyphen	(As above)	
Start new paragraph		
No new paragraph		
Transpose		
Close up	linking characters	
Insert or substitute space between characters or words	through character or where required	
Reduce space between characters or words	between characters or words affected	