Parsimonious Reconstruction of Network Evolution

Rob Patro[1,2], Emre Sefer[1,2], Justin Malin[1,3], Guillaume Marçais[1,4], Saket Navlakha[5], and Carl Kingsford[1,2,3,4]

[1] Center for Bioinformatics and Computational Biology
[2] Department of Computer Science
[3] Computational Biology, Bioinformatics and Genomics Concentration, Biological Sciences Graduate Program
[4] Program in Applied Mathematics, Statistics, and Scientific Computation University of Maryland, College Park, MD 20742, USA
[5] School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Abstract. We consider the problem of reconstructing a maximally parsimonious history of network evolution under models that support gene duplication and loss and independent interaction gain and loss. We introduce a combinatorial framework for encoding network histories, and we give a fast procedure that, given a set of duplication histories, in practice finds network histories with close to the minimum number of interaction gain or loss events. In contrast to previous studies, our method does not require knowing the relative ordering of unrelated duplication events. Results on simulated histories suggest that common ancestral networks can be accurately reconstructed using this parsimony approach.

1 Introduction

High-throughput experiments have revealed thousands of regulatory and protein-protein interactions that occur in the cells of present-day species. To understand why these interactions take place, it is necessary to view them from an evolutionary perspective. In analogy with ancestral genome reconstruction [22], we consider the problem of predicting the topology of the common ancestor of pathways, complexes, or regulatory programs present in multiple extant species.

Generating plausible ancestral networks can help answer many natural questions that arise about how present-day networks have evolved. For example, joint histories can be used to compare the conservation and the route to divergence of corresponding processes in two species. This allows us to more finely quantify how modularity has changed over time [15] and how interactions within a protein complex may have reconfigured across species starting from a single shared state [24]. Such analysis can also be integrated to develop better network alignment algorithms and better network-based phylogenies [11, 27, 8, 9, 16], and it can be used to study robustness and evolvability [1, 10, 26]. Further, inferred

changes in metabolic networks can be linked to changes in the biochemical environment in which each species has evolved, and this can reveal novel mechanisms of ecological adaptation [4, 3]. Finally, comparing network histories inferred using different model parameters can be used to estimate the likelihoods of various evolutionary events [18, 21].

There has been some recent work on reconstructing ancestral interactions. Gibson and Goldberg [13] presented a framework for estimating ancestral protein interaction networks that handles gene duplication and interaction loss using gene trees reconciled against a species phylogeny. However, their approach assumes that interaction losses occur immediately after duplication and does not support interaction gain outside of gene duplication. These assumptions are limiting because interaction loses may occur well after duplication, and independent gains are believed to occur at non-trivial rates [17]. Dutkowski and Tiuryn [8] provided a probabilistic method for inferring ancestral interactions with the goal of improved network alignment. Their approach is based on constructing a Bayesian network with a tree topology where binary random variables represent existence or non-existence of potential interactions. A similar graphical model was proposed by Pinney et al. [25], who applied it to inferring ancestral interactions between bZIP proteins. In the former method, interaction addition and deletion is assumed to occur only immediately following a duplication or speciation event. Further, both methods assume the relative ordering of duplication events is known even between events in unrelated homology groups. Pinney et al. [25] also explore a parsimony-based approach [19] and find it to work well; however, it too assumes a known ordering of unrelated duplication events. The main drawback of these approaches is that the assumed ordering comes from sequence-derived branch lengths, which do not necessarily agree with rates that would be estimated based on network evolution [31]. This motivates an approach such as we describe below that does not use branch lengths as input.

Zhang and Moret [31, 30] use a maximal likelihood method to reconstruct ancestral regulatory networks as a means to improve estimation of regulatory networks in extant species. Mithani et al. [20] study the evolution of metabolic networks, but they only model the gain and loss of interactions amongst a fixed set of metabolites, whereas we also consider node duplication and loss encoded by a tree. Navlakha and Kingsford [21] present greedy algorithms for finding high-likelihood ancestral networks under several assumed models of network growth. They applied these methods to a yeast protein interaction network and a social network to estimate relative arrival times of nodes and interactions and found that the inferred histories matched many independently studied properties of network growth. This attests to the feasibility of using networks to study evolution. The authors, however, only consider a single network at a time, and there is no guarantee that independent reconstruction of two networks will converge to a common ancestor.

Here, we introduce a combinatorial framework for representing histories of network evolution that can encode gene duplication, gene loss, interaction gain and interaction loss at arbitrary times and does not assume a known total order-

ing of duplication events. We show that nearly-minimal parsimonious histories of interaction gain and loss can be computed in practice quickly given a duplication history. In simulated settings, we show that these parsimonious histories can be used to accurately reconstruct a common ancestral regulatory network of two extant regulatory networks.

2   A framework for representing network histories

Any natural model of network evolution will include events for gene duplication, gene loss, interaction gain, and interaction loss. Many such growth models have been studied (e.g. [6, 29, 23, 14, 1, 30]). We now describe how these events can be encoded in a history graph.

Consider a set V of proteins or genes (henceforth "nodes") descended from a common ancestor by duplication events. Those duplication events can be encoded in a binary *duplication tree* T with the items of V as the leaves. An internal node u in T represents a duplication event of u into its left and right children, u_L and u_R. In this representation, after a duplication event, the node represented by u conceptually does not exist anymore and has been replaced by its two children. The leaves of a duplication tree are labeled *Present* or *Absent*. Absent leaves represent products of duplication events that were subsequently lost. A collection of such trees is a *duplication forest* F.

The gain and loss of interactions can be represented with additional non-tree edges placed on a duplication forest. A non-tree edge $\{u, v\}$ represents an *edge flip event*, where the present / absent state of the interaction between u and v is changed to Present if the interaction is currently Absent or to Absent if the interaction is currently Present. Let P_u and P_v be the paths from nodes u and v to the root. An interaction exists between u and v if there are an odd number of such flip non-tree edges between nodes in P_u and P_v. Every non-tree edge between P_u and P_v, therefore, represents alternatively interaction creation or deletion between nodes u and v in the evolution of the biological network.

A graph H consisting of the union of a duplication forest and flip non-tree edges is a *network history*. A history H *constructs* a graph G when the Present leaves of the duplication forest in H correspond to the nodes of G and the flip edges of H imply an interaction between u and v if and only if $\{u, v\}$ is an interaction in G. See Figure 1 for an example history.

Not all placements of non-tree edges lead to a valid network history. The interaction histories have to be consistent with some temporal embedding of the tree. Let t_u^c and t_u^d be respectively the time of creation and duplication of node u, Naturally, $t_u^c < t_u^d$, $t_u^d = \infty$ if u is a Present leaf, and if v is the child of u, then by definition we have

$$t_u^c < t_u^d = t_v^c < t_v^d. \tag{1}$$

If $\{u, v\}$ is a flip edge, then the time $t_{\{u,v\}}$ of appearance of this edge must satisfy

$$t_u^c \le t_{\{u,v\}} < t_u^d \quad \text{and} \quad t_v^c \le t_{\{u,v\}} < t_v^d, \tag{2}$$
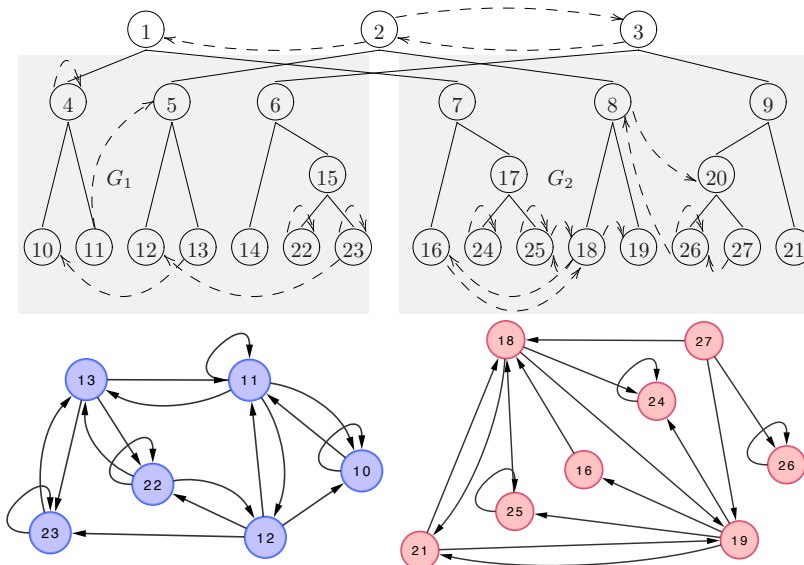
Fig. 1: A duplication forest (solid edges at top) with the non-tree edges (dashed) necessary to construct G_1 and G_2 (shown at bottom). Nodes 1, 2, and 3 represent the 3 homology groups present in the ancestral graph. Node 14 was lost. As an example of the connectivity induced by the non-tree edges, consider edge $(27, 18)$ in G_2 which is implied by the directed non-tree edge from $(3, 2)$. However, the reverse edge, $(18, 27)$, which is implied by $(2, 3)$, does not exist because its state is flipped by $(8, 20)$.

because an event between u and v can only occur when both u and v exist. A history graph H is said to be *valid* if there exist t_u^c, t_u^d for every node u such that conditions (1) and (2) are satisfied for every non-tree edge.

Whether a particular history is valid can be checked combinatorially using the following alternative characterization of validity. A *k-blocking loop* is a set of flip edges $\{\{u_i, v_i\}\}_{0 \leq i < k}$ such that u_{i+1} is an ancestor of v_i in the tree for $0 \leq i < k$ (where the index $i + 1$ is taken modulo k). See Figure 2 for examples. Blocking loops are not permitted in valid histories and, conversely, the non-existence of blocking loops implies that a history is valid, as shown in Prop. 1.

Proposition 1 *A history graph H is valid if and only if it does not have any blocking loop of any length.*

Proof. Suppose there is a k-blocking loop. Using the same notation as above, we have the inequalities

$$t_{u_0}^d > t_{\{u_0, v_0\}} \geq t_{v_0}^c \geq t_{u_1}^d > t_{\{u_1, v_1\}} \geq \ldots \geq t_{v_{k-1}}^c \geq t_{u_0}^d,$$

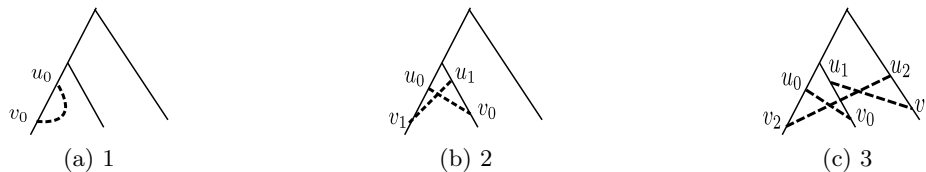which is a contradiction. Hence, to not have any blocking loops is necessary.

Fig. 2: Blocking loops of size 1, 2 and 3. The solid lines represent a subset of the tree T. The dashed lines are non-tree edges representing interaction flip events.

Conversely, suppose that H does not have any blocking loops. We assign times to the nodes and non-tree edges using a modified depth-first search (DFS) algorithm following the tree edges only. First, the root of the tree is given a creation time of 0. During DFS, just before calling DFS recursively on the left and right children of a node u, we set the duplication time $t_u^d = \max\{\max t_{\{u,v\}} + 1, t_u^c + 1\}$, where the second max is taken over all non-tree edges adjacent to u. Also, we set the creation time of the children $t_{u_L}^c = t_{u_R}^c = t_u^d$.

When DFS visits a node u with some edge $\{u,v\}$ where v has not been assigned a creation time, u is added to a set Q and DFS is not called recursively on the children of u. The main loop consists of calling DFS again on all the nodes in Q until this set is empty. By construction, the algorithm assigns times which satisfy conditions (1) and (2). Therefore, if the algorithm terminates, H is a valid history.

At each main iteration, the nodes in the set Q are all the nodes u for which t_u^c is set but t_u^d is not set. It suffices to show that at each such iteration, at least one of the nodes in the set Q will not be added again to Q by a call to DFS. In other words, for at least one node $u \in Q$, every non-tree edge $\{u,v\}$ has t_v^c set. For a contradiction, suppose not. Take $u_1 \in Q$ and $\{u_1, v_1\}$ with $t_{v_1}^c$ not set. There is necessarily an ancestor of v_1, call it u_2, which is in Q. Similarly, take $\{u_2, v_2\}$ with $t_{v_2}^c$ not set and its ancestor $u_3 \in Q$, and so on. Because Q is finite, $u_j = u_i$ for some $j > i$, and we constructed a blocking loop. Hence, the algorithm must terminate. □

3   Parsimonious reconstruction of a network history

Traditional phylogenetic inference algorithms and reconciliation between gene and species trees can be used to obtain duplication and speciation histories [5, 7, 2]. What remains is the reconstruction of interaction gain and loss events. This leads to the following problem:

Problem 1. Given a duplication forest F and an extant network G, find H, a valid history constructing G, with a minimum number of flip edges.

We will show that nearly optimal solutions to this problem for a large range of instances can be solved in polynomial time in practice. Whether Problem 1 is NP-hard or admits a polynomial-time algorithm for all instances remains open.

3.1 A fast heuristic algorithm

The challenge of Problem 1 comes from avoiding the creation of blocking loops. A polynomial-time algorithm can find a minimum set of flip edges that reconstructs a graph G and does not contain 1- and 2-blocking loops but allows longer blocking loops. We define an *interaction encoding* of $G = (V, E)$ as a function $f_G : V \times V \to \{0, 1\}$ such that: $f_G(u, v) = 1$ if $\{u, v\}$ is an interaction in G and $f_G(u, v) = 0$ otherwise. We omit the subscript on f_G if G is clear from the context.

The following intertwined dynamic programming recurrences find the minimum number of flip edges required for H to construct a given graph G if blocking loops of length ≥ 3 are allowed. First, $S(u, f)$ finds the minimum number of flip edges for the subtree rooted at u and interaction encoding f:

$$S(u, f) = S(u_L, f) + S(u_R, f) + A(u_L, u_R, f). \tag{3}$$

The expression $A(u, v, f)$ gives the minimum number of flip edges that should be placed between the subtree rooted at u and the subtree rooted at v. This can be computed using the recurrence:

$$A(u, v, f) = \min \begin{cases} A(u_L, v, f) + A(u_R, v, f) \\ A(u, v_L, f) + A(u, v_R, f) \\ 1 + A(u_L, v, \bar{f}) + A(u_R, v, \bar{f}) \\ 1 + A(u, v_L, \bar{f}) + A(u, v_R, \bar{f}). \end{cases} \tag{4}$$

In the above, if one of u or v is a leaf but the other is not, the options that look at non-existent children are disallowed.

The function $\bar{f}$ in Eqn. (4) is defined as $1 - f$ and thus represents a function such that $\bar{f}(x)$ has opposite parity from $f(x)$ for all x. The A recurrence considers two possible options: (1) We connect u and v with a non-tree edge, this costs us 1 and flips the parity of all interactions going between the subtree rooted at u and the subtree rooted at v; or (2) We do not connect u and v with a flip edge. This costs 0 and keeps the parity requirement the same. Regardless of the choice to create an edge, since we are not allowed to have a 2-blocking loop, either (a) we possibly connect u to some descendant of v (and do not connect v to a descendant of u) or (b) we possibly connect v to some descendant of u (and do not connect u to a descendant of v).

The base case for the S recurrence when u is a leaf and the base case for the A recurrence when u and v are leaves are:

$$S(u, f) = 0 \quad \text{and} \quad A(u, v, f) = f(u, v).$$

The minimum number of flip edges needed to turn a duplication forest F into a history constructing G (allowing blocking loops of ≥ 3) is then given by $\sum_r S(r, d_G) + \sum_{r,q} A(r, q, d_G)$, where d_G is the interaction encoding of G, and the sums are over roots r, q of the trees in F. Standard backtracking can be used to recover the actual minimum edge set. The dynamic program runs in $O(n^2)$ time and space because only two functions f are ever considered: d_G, and $\bar{d}_G$. This yields $\approx n \times n \times 2$ subproblems, each of which can be solved in constant time.

3.2 Removing blocking loops

If the solution contains blocking loops of length ≥ 3, one can choose an edge in some blocking loop, forbid that edge from appearing in the solution, and rerun the dynamic program. Because there are $O(n^2)$ possible non-tree edges, iterating this procedure will terminate in polynomial time. In practice, we can choose to exclude the non-tree edge that participates in the largest number of loops. We repeat this until a valid solution is obtained. In the worst case, one may obtain a solution where all non-tree edges are placed at leaves, but in practice long blocking loops do not often arise, and the obtained solutions are close to optimal (see Sec. 4.2).

3.3 Reconstruction of a common ancestor of two graphs

Given extant networks of several species, in addition to the reconstructed history, we seek a parsimonious estimate for their common ancestor network. Specifically, given extant networks G_1 and G_2, with interaction encodings d_1 and d_2, and their duplication forests F_1 and F_2, we want to find an ancestral network $X = (V_X, E_X)$ such that the cost of X evolving into G_1 and G_2 after speciation is minimized. V_X is the set of roots of the homology forests. We assume that the networks of the two species evolved independently after speciation. Therefore, we can use the recurrence above applied to F_1 and F_2 to compute $A_{F_1}(r, q, d_1)$ and $A_{F_2}(r, q, d_2)$ independently for $r, q \in V_X$, and then select interactions in X as follows. E_X of X is given by the pairs $r, q \in V_X \times V_X$ for which creating an interaction leads to a lower total cost than not creating an interaction. Formally, we place an interaction $\{r, q\}$ in E_X if

$$1 + A_{F_1}(r, q, \bar{d}_1) + A_{F_2}(r, q, \bar{d}_2) < A_{F_1}(r, q, d_1) + A_{F_2}(r, q, d_2). \qquad (5)$$

Rule (5) creates an interaction in X if doing so causes the cost of parsimonious histories inferred for G_1 and G_2 between the homology groups associated with r and q to be smaller than if no interaction was created.

3.4 Modifications for self-loops

Self-loops (homodimers) can be accommodated by modifying recurrence (3):

$$S'(u, f) = \min \begin{cases} S'(u_L, f) + S'(u_R, f) + A(u_L, u_R, f) \\ 1 + S'(u_L, \bar{f}) + S'(u_R, \bar{f}) + A(u_L, u_R, \bar{f}). \end{cases} \qquad (6)$$

The intuition here is that paying cost 1 to create a self-loop on node u creates (or removes) interactions, including self-loops, among all the descendants of u.

3.5 Modifications for directed graphs

Finally, the algorithm can be modified to handle evolutionary histories of directed graphs. For this, only the recurrence A need be modified. When computing $A'(u, v, f)$, a non-tree edge can be included from u to v, from v to u,

both, or neither. Each of these cases modifies the function f in a different way. Specifically:

$$A'(u,v,f) = \min \begin{cases} 0 + A'(u_L,v,f) + A'(u_R,v,f) \\ 1 + A'(u_L,v,\overleftarrow{f}) + A'(u_R,v,\overleftarrow{f}) \\ 1 + A'(u_L,v,\overrightarrow{f}) + A'(u_R,v,\overrightarrow{f}) \\ 2 + A'(u_L,v,\overleftrightarrow{f}) + A'(u_R,v,\overleftrightarrow{f}), \\ \quad\vdots \end{cases}$$

where the vertical ellipsis indicates the symmetric cases involving v_L and v_R, and where $\overrightarrow{f}, \overleftarrow{f}, \overleftrightarrow{f}$ are defined, depending on u and v, as follows:

$$\overrightarrow{f}(x,y) = \min \begin{cases} 1 - f(x,y) & \text{if } x \in \text{ST}(u) \text{ and } y \in \text{ST}(v) \\ f(x,y) & \text{otherwise} \end{cases} \qquad (7)$$

$$\overleftrightarrow{f}(x,y) = \min \begin{cases} 1 - f(x,y) & \text{if } x \in \text{ST}(u) \text{ and } y \in \text{ST}(v) \text{ or vice versa} \\ f(x,y) & \text{otherwise,} \end{cases} \qquad (8)$$

with $\overleftarrow{f}$ defined analogously to $\overrightarrow{f}$. Here, $\text{ST}(u)$ indicates the set of nodes in the subtree rooted at u.

The heuristic also can be extended to handle different costs for interaction addition and interaction deletion by changing the constants in the recurrences to be functions dependent on f.

4 Results

4.1 Generating plausible simulated histories

We use a *degree-dependent model* (DDM) to simulate an evolutionary path from a putative ancestral network to its extant state. The model simulates node duplication, node deletion, independent interaction gain, and independent interaction loss with given probabilities P_{ndup}, P_{nloss}, P_{egain} and P_{eloss}, respectively. The nodes or edges involved in a modification are chosen probabilistically based on their degrees (as in [28]) according to the following expressions:

$$P(u \mid \text{node duplication}) \propto 1/k_u \qquad\qquad P(u \mid \text{node loss}) \propto 1/k_u \qquad (9)$$
$$P((u,v) \mid \text{interaction gain}) \propto k_u^o \qquad P((u,v) \mid \text{interaction loss}) \propto 1/k_u^o, \qquad (10)$$

where k_u^o is the out-degree of a node u, and k_u is the total degree. At each time step, the distribution of possible modifications to the graph is calculated as $P(\text{modification}) = P_{operation}P(\text{object} \mid \text{operation})$. Nodes with out-degree of 0 are removed. Varying parameters P_{ndup}, P_{nloss}, P_{egain} and P_{eloss} can produce a wide variety of densities and sizes. We also consider a *degree-independent model*

(DIM) in which the four conditional probabilities in Eqns. (9) and (10) are all equal.

The DDM model is theoretically capable of producing evolutionary trajectories between any two networks while incorporating preferential attachment to the source node and random uniform choice of the target node. Furthermore, choosing a node for duplication or loss in inverse proportion to its degree favors an event in inverse relation to its expected disruption of the network.

We also consider a model of regulatory network evolution by Foster et al. [12], which is based on gene duplication, with incoming and outgoing interactions kept after duplication as in other models (P_{inkeep} and $P_{outkeep}$ probabilities respectively). New edges are added with probability $P_{innovation}$.

In all of the network evolution models, we started with a random connected seed graph that has 10 nodes and 25 interactions. We evolved it to X by 200 operations after which we introduce a speciation event, and then both G_1 and G_2 evolve from X by an additional 200 operations each. To generate more biologically plausible ancestral graphs, instances were kept only if the ancestral graph X had an in-degree that fit an exponential distribution with parameter between 1.0 and 1.2 or an out-degree that was scale-free with parameter between 1.8 and 2.2.

4.2 Reconstructing histories

Optimality of loop breaking. The greedy procedure to break blocking loops produces histories that are very close to optimal. We generated 1400 networks using the DDM model with the range of parameters on the x-axis of Fig. 3a. In the vast majority of cases (1325 out of 1400), either no loop breaking is required, or the solution discovered after greedily breaking all loops has the same cost as the original solution. In these cases, therefore, the method returned a provably maximally parsimonious set of interaction modification events. In the remaining 75 cases (5.4%), greedily removing blocking loops increased the number of interaction modifications by no more than 10 ($< 2\%$ of the initial number of interaction modification events). Since the initial solution provides a lower bound on the optimal, we can verify that the greedy procedure always found a solution within 2% of the optimal (and perhaps even better). Thus, it seems that in practice, while blocking loops occur, the greedy procedure does a good job of eliminating them without increasing the number of events significantly.

Effect of growth model and its parameters. Modeling the evolutionary dynamics of a regulatory network is still an active topic of research. We therefore experimented with three different network models (Sec. 4.1). Despite their differences, high precision and recall (measured as the F1 score) can be obtained for all of them for many choices of their parameters (Fig. 3a-c). Very good performance can be achieved under the general model presented above whether degree distributions are taken into account (Fig. 3a) or not (Fig. 3b) when selecting nodes and interactions to modify. In these cases, for most parameter choices, precision is close to 1.0, meaning every interaction predicted to be in the ancestor, in fact,
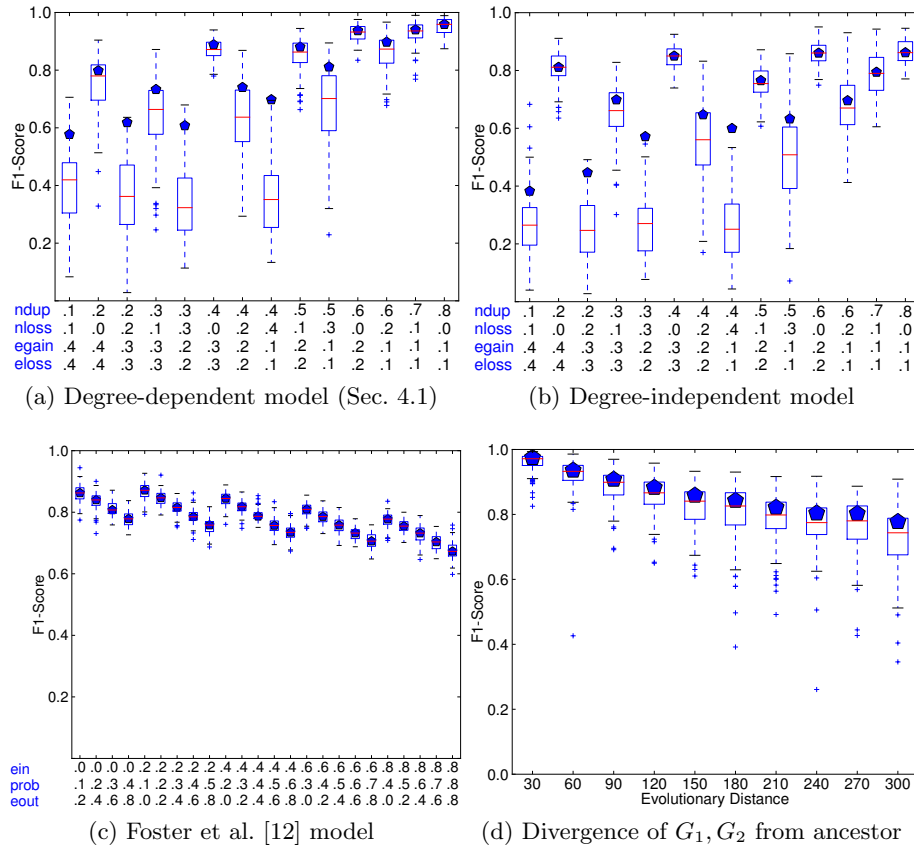
(a) Degree-dependent model (Sec. 4.1)

(b) Degree-independent model

(c) Foster et al. [12] model

(d) Divergence of G_1, G_2 from ancestor

Fig. 3: (a-c) Effect of model parameters on reconstruction accuracy under three different models. "Prob" in (c) is $P_{\text{innovation}}$. (d) Effect of evolutionary distance (number of network modification operations) on the quality of the ancestral network reconstruction. In both plots, boxes show 1st and 3rd quartile over 100 networks with median indicated by a line. Pentagons show the median if interactions incident to nodes lost in both lineages are not considered.

was. Recall is often lower. The Foster et al. [12] model, with its heavy reliance on duplication events and lack of node loss events, tends to be the simplest under which to reconstruct the ancestral graph (Fig. 3c).

The largest factor leading to poorer performance is lower recall caused by gene losses. If all descendants of a gene are lost in both extant networks, it is not possible to reconstruct interactions incident to it. If these interactions are excluded from the computation of recall, the F1 score often improves dramatically. Median F1 scores excluding these interactions are shown as pentagons in Fig. 3.

Robustness to evolutionary divergence. Naturally, the ability to recover the ancestral network degrades as time passes and the extant networks diverge. However, the degradation is slow (Fig. 3d, using the degree-dependent model with parameters fixed at $P_{ndup} = 0.35$, $P_{nloss} = 0.05$, $P_{egain} = 0.3$, and $P_{eloss} = 0.3$). When the distance is small, we are almost always able to recover the ancestral network well, as illustrated by the high F1-scores and small interquartile ranges in Figure 3d. Even when the distance between the ancestral and extant networks is large (300) compared to the average ancestral network size (55), we obtain an F1-score of 0.72 (0.77 when homology groups lost in both lineages are not considered).

5   Conclusion

We have presented a novel framework for representing network histories involving gene duplications, gene loss, and interaction gain and loss for both directed and undirected graphs. A combinatorial characterization for valid histories was given. We have shown that a fast heuristic can recover optimal histories in a large majority of instances. We further provide evidence that, even with a probabilistic, weighted, generative model of network growth, a parsimony approach can recover accurate histories.

Acknowledgements

References

1. Aldana, M., Balleza, E., Kauffman, S., Resendiz, O.: Robustness and evolvability in genetic regulatory networks. J. Theor. Biol. 245(3), 433–448 (2007)
2. Arvestad, L., Berglund, A.C., Sennblad, B.: Bayesian gene/species tree reconciliation and orthology analysis using mcmc. Bioinformatics 19(Suppl. 1), i7i15 (2003)
3. Borenstein, E., Feldman, M.W.: Topological signatures of species interactions in metabolic networks. J. Comput. Biol. 16(2), 191–200 (2009)
4. Borenstein, E., Kupiec, M., Feldman, M.W., Ruppin, E.: Large-scale reconstruction and phylogenetic analysis of metabolic environments. Proc. Natl. Acad. Sci. USA 105(38), 14482–14487 (2008)
5. Chen, K., Durand, D., Farach-Colton, M.: NOTUNG: A program for dating gene duplications and optimizing gene family trees. Journal of Computational Biology 7(3-4), 429–447 (Aug 2000)

6. Chung, F., Lu, L., Dewey, T.G., Galas, D.J.: Duplication models for biological networks. J. Comp. Biol. 10(5), 677–687 (Jan 2003)

7. Durand, D., Halldrsson, B.V., Vernot, B.: A hybrid micromacroevolutionary approach to gene tree reconstruction. J. Comp. Biol. 13(2), 320–335 (Mar 2006)

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

9. Erten, S., Li, X., Bebek, G., Li, J., Koyuturk, M.: Phylogenetic analysis of modularity in protein interaction networks. BMC Bioinformatics 10, 333 (2009)

10. Espinosa-Soto, C., Martin, O.C., Wagner, A.: Phenotypic robustness can increase phenotypic variability after nongenetic perturbations in gene regulatory circuits. J. Evol. Biol. 24(6), 1284–1297 (2011)

11. Flannick, J., Novak, A., Srinivasan, B.S., McAdams, H.H., Batzoglou, S.: Graemlin: general and robust alignment of multiple large interaction networks. Genome Res. 16(9), 1169–1181 (2006)

12. Foster, D.V., Kauffman, S.A., Socolar, J.E.S.: Network growth models and genetic regulatory networks. Phys. Rev. E 73(3), 031912 (Mar 2006)

13. Gibson, T.A., Goldberg, D.S.: Reverse engineering the evolution of protein interaction networks. Pac. Symp. Biocomput. pp. 190–202 (2009)

14. Ispolatov, I., Krapivsky, P.L., Yuryev, A.: Duplication-divergence model of protein interaction network. Phys. Rev. E 71(6 Pt 1), 061911 (Jun 2005)

15. Kreimer, A., Borenstein, E., Gophna, U., Ruppin, E.: The evolution of modularity in bacterial metabolic networks. Proc. Natl. Acad. Sci. USA 105(19), 6976–6981 (2008)

16. Kuchaiev, O., Milenkovic, T., Memisevic, V., Hayes, W., Przulj, N.: Topological network alignment uncovers biological function and phylogeny. J. R. Soc. Interface 7(50), 1341–1354 (2010)

17. Levy, E.D., Pereira-Leal, J.B.: Evolution and dynamics of protein interactions and networks. Curr. Opin. Struct. Biol. 18(3), 349–357 (2008)

18. Middendorf, M., Ziv, E., Wiggins, C.H.: Inferring network mechanisms: the *Drosophila melanogaster* protein interaction network. Proc. Natl. Acad. Sci. USA 102(9), 3192–3197 (2005)

19. Mirkin, B.G., Fenner, T.I., Galperin, M.Y., Koonin, E.V.: Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance of horizontal gene transfer in the evolution of prokaryotes. BMC Evol. Biol. 3, 2 (2003)

20. Mithani, A., Preston, G., Hein, J.: A stochastic model for the evolution of metabolic networks with neighbor dependence. Bioinformatics 25(12), 1528–1535 (2009)

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

22. Pachter, L.: An introduction to reconstructing ancestral genomes. In: Proc. Symp. in Applied Mathematics. vol. 64, pp. 1–20 (2007)

23. Pastor-Satorras, R., Smith, E., Sole, R.: Evolving protein interaction networks from gene duplication. J. Theor. Biol. 222, 199–210 (2003)

24. Pereira-Leal, J.B., Levy, E.D., Kamp, C., Teichmann, S.A.: Evolution of protein complexes by duplication of homomeric interactions. Genome Biol. 8(4), R51 (2007)

25. Pinney, J.W., Amoutzias, G.D., Rattray, M., Robertson, D.L.: Reconstruction of ancestral protein interaction networks for the bZIP transcription factors. Proc. Natl. Acad. Sci. USA 104(51), 20449–20453 (2007)

26. Raman, K., Wagner, A.: Evolvability and robustness in a complex signalling circuit. Mol. Biosyst. 7(4), 1081–1092 (2011)

27. Singh, R., Xu, J., Berger, B.: Pairwise global alignment of protein interaction networks by matching neighborhood topology. In: Proc. Intl. Conf. on Research in Computational Molecular Biology (RECOMB). pp. 16–31 (2007)
28. Stewart, A.J., Seymour, R.M., Pomiankowski, A.: Degree dependence in rates of transcription factor evolution explains the unusual structure of transcription networks. Proc. Biol. Sci. 276(1666), 2493–2501 (2009)
29. Teichmann, S.A., Babu, M.M.: Gene regulatory network growth by duplication. Nat. Genetics 36(5), 492–6 (May 2004)
30. Zhang, X., Moret, B.: Refining transcriptional regulatory networks using network evolutionary models and gene histories. Alg. Mol. Biol. 5(1), 1 (2010)
31. Zhang, X., Moret, B.M.: Boosting the performance of inference algorithms for transcriptional regulatory networks using a phylogenetic approach. In: Proc. Intl. Workshop on Algorithms in Bioinformatics (WABI). pp. 245–258 (2008)