**Reconstructing the History of Interaction Networks using Phylogenetic Trees**

Luke J. Harmon and Scott L. Nuismer

Abstract

**Introduction**

Species interactions are ubiquitous in the natural world. Interactions among competitors, predators, prey, hosts, and parasites shape evolution over short (e.g. Burdon and Thrall 1999; Nuismer and Gandon 2008) and long time periods (e.g. Erlich and Raven 1964; Van Valen 1973; Schluter 2000). The pattern of species interactions in a natural community can be summarized by a species interaction network. Work on natural species interaction networks has revealed a rich array of information about how species interaction networks affect, and are affected by, ecological processes.

Species interaction networks arise via evolution. However, we have only limited information about how such networks evolve. One reason for this is that there is limited information about species interactions in the fossil record (but see xxx). However, there is another potential source of information about the history of interaction networks that we can obtain using phylogenetic approaches. The sets of species within interaction networks are related to one another through phylogeny. Because of this, closely related species are more likely to show similar patterns of species interactions (e.g. closely related predators are likely to eat the same or closely related prey species; Gomez et al. 2010). This “phylogenetic signal” in interaction networks (e.g. Cattin et al. 2004; Ives and Godfrey 2006) suggests that phylogenetic trees can be used to study the history of interaction networks. However, current approaches to combining phylogenetic trees and networks (e.g. Rezende et al. 2007) are generally pattern-based, and do not allow models of network evolution to be fit to phylogenetic comparative data. Furthermore, comparative approaches like independent contrasts, PGLS, and other methods cannot be used for networks.

In this paper we develop and apply a novel phylogenetic comparative method to understand how species interaction networks evolve. Our method requires as inputs a phylogenetic tree and an interaction network for a set of species. We develop a novel pairwise pruning algorithm that allows us to calculate the likelihood of obtaining the given species interaction network given a phylogenetic tree with branch lengths and a simple “copying” model of how interaction networks might evolve. We describe how this algorithm can be used in both an ML and a Bayesian statistical framework, and analyze the evolution of several species interaction networks.

**Methods**

*Network evolution model*

We developed a continuous-time Markov model for the evolution of interaction networks. Throughout this section, we will discuss the model as it applies to species interactions; however, our model applies equally well to other interacting entities that can be connected by a branching tree, such as orthologous gene copies. Our model starts with several assumptions. First, we assume that all species can potentially interact. This leaves out the possibility that species might be totally allopatric; we return to this assumption in the paper’s discussion. Second, we assume that all interactions can be characterized as present or absent. In other words, our model as currently formulated considers only whether or not species interact and does not include the strength or direction of that interaction. Third, we assume that these pairwise interactions change through time, as described by two parameters: q01, the instantaneous rate of interaction gains; and q10, the instantaneous rate of interaction losses. If one considers a set of n species, under our model there are n(n-1)/2 pairs, each of which has an interaction state (0 or 1). We assume that each of these pairs can change that interaction state over some time interval Δt according to our two rate parameters, q01 and q10.

Assuming that all pairs change at the same rate and independently of one another, we can use the standard approach for a continuous-time Markov model, defining a differential equation

P’(t) = P(t) Q

Where P(t) is the probability distribution for the two states (0 and 1) at time t, prime denotes differentiation, and Q is the rate matrix:

The solution to (1) is:

The matrix exponential in this case can be calculated exactly, so that:

This model is sufficient to consider the evolution of species interactions among a set of n species, but we also need to include speciation. We do that by invoking a copying mechanisms of network growth. For this we assume that at speciation, both daughter lineages inherit all of the interactions of the parent lineage. For example, consider an ancestral lineage A that splits into two daughters B and C. If the ancestral lineage A interacted with lineages X and Y but not Z, then, immediately following speciation both daughters would share this pattern of interacting with X and Y but not Z (Figure 1B). We also need to consider the new pairwise relationship between the two daughters. A number of assumptions are possible – e.g. new sister taxa always interact, never interact, interact with probability p, and so on. For simplicity, we assume that the interaction probability between new sister species follows the equilibrium probability distribution from the Markov chain:

We can describe network evolution among a set of lineages over some time interval, and also at the moment of speciation. We assume that speciation events are short and rare, so that no more than one can happen at the same time. Under this assumption, then our model provides a complete description of how networks can evolve on phylogenetic trees.

To characterize the properties of our model, we used simulations. We simulated our model across a set of phylogenetic trees generated under a birth-death model, varying tree size (n = 20, 50, and 100) and relative extinction fractions (), in each case using the appropriate net diversification rate () so that a tree of total age 10 units would have, on average, n species. For each of these 9 trees, we simulated interaction networks with both symmetric and asymmetric transition rates. We characterized the properties of the networks produced using a set of common network measurements.

*Fitting the network model to comparative data: a pairwise pruning algorithm*

We next describe how to calculate the likelihood for our network evolution model given a species interaction network and an ultrametric phylogenetic tree with branch lengths. We assume that all of the species in the tree are contemporaneous – in other words, our method as currently implemented does not allow the inclusion of fossil taxa. Branch lengths can be in any units, and results from these analyses will share those units. For example, if the tree has branch lengths in millions of years, then estimated rate parameters will have units of my-1. Ultrametric trees with arbitrary branch lengths can also be used, but parameter estimates will not be comparable across studies.

We develop a novel pairwise pruning algorithm for calculating the likelihood of paired data (such as species interaction networks) on a phylogenetic tree. Our algorithm is based on Felsenstein (1973), who introduced a pruning algorithm to calculate the likelihood for a discrete character evolving along the branches of a phylogenetic tree (Figure 2). Briefly, this algorithm prunes the phylogenetic tree by one pair of sister species at a time, calculating the conditional likelihood for each character state at that node. At the root of the tree, one can easily obtain the overall likelihood of the model over the entire tree using the conditional likelihoods.

We have modified this algorithm to create pairwise pruning (Figure 2). Pairwise pruning considers species pairs, and so, for a phylogenetic tree of n tips, requires data for the characteristics of n \* (n-1) / 2 pairs. The pairwise pruning algorithm has two main modifications from standard pruning. First, during each step of the pruning algorithm, all branches of the tree are pruned back to the level of the shallowest node in the tree (Figure 2). This differs from the original algorithm, where sister taxa are pruned from the tree. Second, when a particular species pair coalesces at their most recent common ancestor, one treats the interaction state of that pair as if it were a character at the root of the tree; that is, one finds product of the conditional likelihoods multiplied by the probability of each state at that speciation event. In other words, one part of the overall likelihood is retained by the algorithm at each node in the tree, rather than just at the root.

The pairwise pruning algorithm returns the full likelihood of the model given the data, which could then be used for statistical inference in a ML or Bayesian framework. We have implemented a Bayesian MCMC method using R to fit our model to data. Our method is flexible so that one can alter both priors and proposal distributions. As defaults, we set exponential priors with a mean of 0.1 for both q01 and q10, and use a uniform proposal distribution with width 0.2. The MCMC is constructed in a standard way using the likelihoods obtained from our pairwise pruning algorithm.

We tested our software using the method developed by xxx. To do this, we randomly drew model parameters from the default prior distribution. For each set of model parameters, we simulated data using a birth-death tree (parameters). We then analyzed our simulated data using the MCMC. For this analysis we chose starting values of xxx and ran the analysis for xxx generations. We then discarded the first xxx steps as burn-in, then sampled every 100 generations to build a posterior. We then determined the percentile of the true parameter values in this posterior distribution. We repeated this entire procedure 1000 times. For a properly implemented MCMC algorithm, the expected distribution of these percentiles is uniform; we evaluated this using a q-q plot.

To better characterize the behavior of the model in returning accurate and precise parameter estimates, we ran another set of simulations. We used the same simulation parameters described above, in each case running an MCMC to generate posterior estimates of model parameters. We then compared the posterior distributions to the true values.

*Empirical example*

**Results**

**Discussion**

**References**

Figure 1.

