True Nonlinear Dynamics from Incomplete Networks

Chunheng Jiang, Malik Magdon-Ismail, Jianxi Gao
Rensselaer Polytechnic Institute
110 8th Street, Troy, NY 12180
jiangc4@rpi.edu, magdon@cs.rpi.edu, gaoj8@rpi.edu

Abstract

We study *nonlinear* dynamics on complex networks. Each vertex i has a state x_i which evolves according to a networked dynamics to a steady-state x_i^*. We develop fundamental tools to learn the true steady-state of a small part of the network, *without knowing the full network*. A naive approach and the current state-of-the-art is to follow the dynamics of the observed partial network to local equilibrium. This dramatically fails to extract the true steady state. We use a mean-field approach to map the dynamics of the unseen part of the network to a single node, which allows us to recover accurate estimates of steady-state on as few as 5 observed vertices in domains ranging from ecology to social networks to gene regulation. Incomplete networks are the norm in practice, and we offer new ways to think about nonlinear dynamics when only sparse information is available.

1 Dynamical Systems on Incomplete Networks

The fundamental task in learning is to infer unknown quantities of interest from incomplete data. We study learning complex nonlinear dynamics on networks from incomplete data. Such problems are fundamental because complex nonlinear dynamical systems are ubiquitous, often modeled as coupled nonlinear ordinary differential equations (ODEs), for example epidemic spreading (Pastor-Satorras and Vespignani 2001), Michaelis-Menten gene regulatory dynamics (Alon 2006; Gao, Barzel, and Barabási 2016), Lotka-Volterra ecological dynamics (Lotka 1910). A graph $G = (V, E)$ with $n \times n$ (weighted) adjacency matrix A is the backbone on which the dynamical equations are coupled together. We consider a general dynamics in which each vertex i of G has a state x_i which evolves according to a self-driving force and a sum of interaction forces over neighbors

$$\dot{x}_i = f(x_i) + \sum_{j \in V} A_{ij} g(x_i, x_j). \tag{1}$$

The functions $f(\cdot)$ and $g(\cdot, \cdot)$ are general and usually nonlinear, and the positive weighted connectivity matrix A modulates the interactions between vertices. Several instances of such dynamics with appropriate choices of $f(\cdot)$ and $g(\cdot, \cdot)$ are shown in Table 1. From an initial state, one can step forward in time, simulating the dynamics in (1) until convergence to equilibrium states x_i^*.

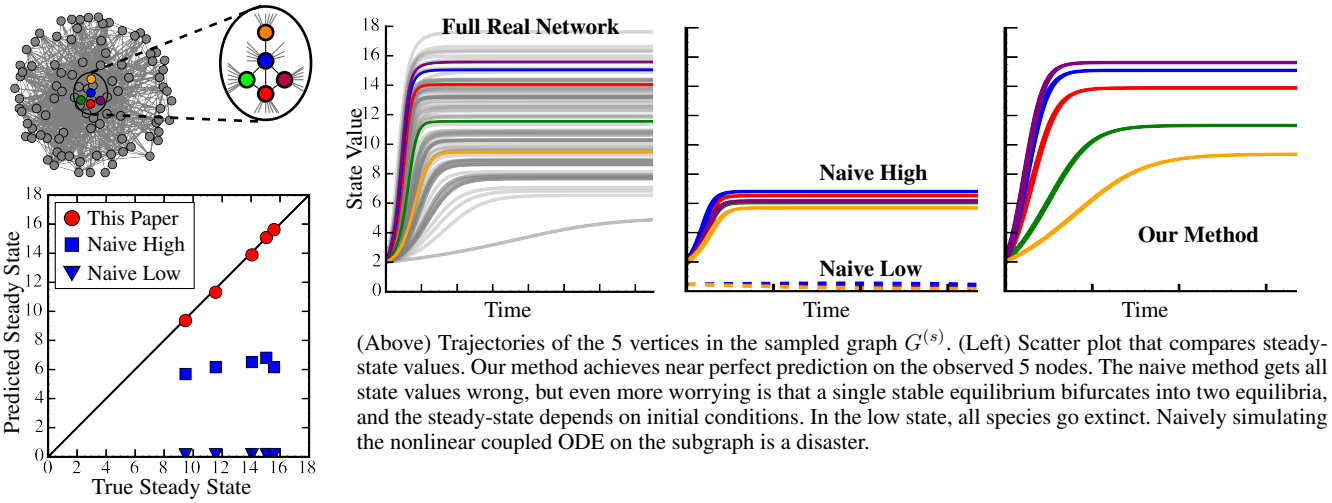
The complete information setting in (1) is unrealistic, and we must accept that in practice, only part of a network can be measured. Hence, we assume that a subgraph with m nodes $G^{(s)} = (V^{(s)}, E^{(s)})$ is known, with corresponding $m \times m$ adjacency matrix $A^{(s)}$, where $V^{(s)} \subseteq V$ and $E^{(s)} \subseteq E$ (s for sampled). This paradigm is unavoidable when the full network is unmeasurable (for example, protein-protein interactions, metabolic and terrorist networks (Stumpf and Wiuf 2005)). The paradigm is also useful when the full network is too large to handle, where one can deliberately sample a much smaller subnetwork for the analysis. Analyzing a full network from a sampled subnetwork has been studied in several contexts, e.g. to estimate average or total degree (Kurant, Markopoulou, and Thiran 2011); degree distributions and clustering coefficients (Stumpf and Wiuf 2005; Gjoka, Kurant, and Markopoulou 2013; Seshadhri, Pinar, and Kolda 2014); shortest paths (Leskovec and Faloutsos 2006); motif counts (Klusowski and Wu 2018); vertex and edge counts (Katzir, Liberty, and Somekh 2011; Kurant, Butts, and Markopoulou 2012).

Our task is to estimate true steady-states x^* for vertices in $V^{(s)}$, despite only seeing an incomplete network $G^{(s)}$. The state-of-the-art naive approach is to simulate (1) on the subgraph $G^{(s)}$. For example, one may collect a social network from Boston and run the epidemic model (Table 1) to obtain the probability of each person to be infected. The results are a dramatic and universal disaster, because the sub-social network of Boston is just a small part of a vast social network, and the people in Boston interact with people outside. That the naive method is bad is not surprising because the essence of the dynamics in (1) are the interactions, and the subgraph is missing many of those interactions. Hence, not observing a large part of the network appears to be an insurmountable hurdle to learning the true steady states on the observed (small) part of the network.

We develop a methodology to accurately predict *true* steady states using only information local to $G^{(s)}$. We demonstrate the power of our approach in Figure 1, for an

Applications	Vertex	State at vertex i	Dynamics
Ecological (1949; 2006)	Species	Abundance	$\dot{x}_i = B_i + x_i(1 - \frac{x_i}{K_i})(\frac{x_i}{C_i} - 1) + \sum_j A_{ij} \frac{x_i x_j}{D_i + E_i x_i + H_j x_j}$
Regulatory (2006; 2008)	Gene	Expression level	$\dot{x}_i = -B x_i^f + \sum_j A_{ij} R \frac{x_j^h}{x_j^h + 1}$
Epidemic (2001; 2004; 2005)	Person	Infection rate	$\dot{x}_i = -B x_i + \sum_j A_{ij} R (1 - x_i) x_j$

Table 1: Examples of real systems with nonlinear interaction dynamics.



(Above) Trajectories of the 5 vertices in the sampled graph $G^{(s)}$. (Left) Scatter plot that compares steady-state values. Our method achieves near perfect prediction on the observed 5 nodes. The naive method gets all state values wrong, but even more worrying is that a single stable equilibrium bifurcates into two equilibria, and the steady-state depends on initial conditions. In the low state, all species go extinct. Naively simulating the nonlinear coupled ODE on the subgraph is a disaster.

Figure 1: Predicting steady-state abundances of 5 species interacting in a larger 97 species ecological network. Predictions use only the interactions of those five species (incomplete information).

ecological network where vertices are species and states are species-abundance. This ecological network of 97 species follows the symbiotic dynamics in Table 1, see (Gao, Barzel, and Barabási 2016). Let us see what happens when a biologist who is interested in *five* species collects the relevant 5-vertex subgraph $G^{(s)}$ and performs a naive simulation on the subgraph to get a steady-state. The results in Figure 1 are as expected. The naive simulation is wrong. Even worse, it cannot identify the number of equilibria from the subgraph: the full network has one attractor, but the subgraph has two. It means that, with the wrong initial conditions, the biologist would conclude that the five species are going extinct, when in fact they are all doing fine in the real network.

Our Contributions. We give the first method to accurately learn steady-state dynamics when only a part of the network is observed. This is remarkable given the inherent interactive nature of the dynamics. The result in Figure 1 demonstrates that our methods extract very close approximations to the true steady state dynamics *in the full network* when just 5% of vertices are observed. This surprising result has the potential for huge impact since up to now, the state-of-the-art is the naive approach which produces completely wrong conclusions. There are three main ideas behind our method.

- A mean field approximation to account for the impact of the unobserved part of the network.
- Summarizing the mean field impact using a *resilience* parameter, β, which depends only on network topology. How the resilience impacts the final outcome depends on the coupled nonlinear dynamics through $f(\cdot)$ and $g(\cdot, \cdot)$.
- Estimating the *full network's* resilience from the observed subgraph. A network's resilience is important in other contexts. The resilience characterizes a complex system's ability to retain its basic functionality under edge and vertex faults. Hence, our estimates of resilience from incomplete information are of independent interest.

Combining these three ideas, we obtain accurate estimates of the steady-states as in Figure 1. Our estimates are near-exact matches to the true steady-states.

2 Model

The true dynamics on the full network G are governed by the coupled nonlinear dynamics in (1). We represent G by its adjacency matrix A, and assume that the total size of the network, n, is known. The observed sampled subgraph $G^{(s)} = (V^{(s)}, E^{(s)})$ has adjacency matrix $A^{(s)}$. There are many ways to sample vertices and edges from a graph. We focus on two natural sampling methods which are reasonable models of how the incomplete data is often obtained.

- **(Random Vertex Sampling)** Form the induced subgraph for randomly sampled vertices. We assume the degrees of the sampled vertices are also known. For example, we know the number of friends each person has in a social network and who is friends with whom among the sampled subnetwork.

- **(Random Walk)** A random vertex is sampled. At each step, an available edge is followed to sample a new vertex. The degrees of the vertices are implicitly available since, at each step, the available edges must be known.

Our method can be extended to other sampling schemes, such as edge sampling, degree biased sampling, Metropolis-Hastings random walks. The main property we require of the sampling is that specific topological parameters of the graph can be reliably estimated from the sample.

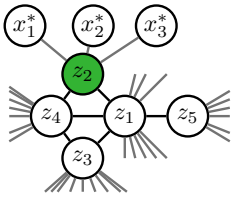
For $i \in V^{(s)}$, the steady-state value in G is denoted x_i^*. We denote by z_i^* the steady-state value obtained using the partially observed network. We loosely use z_i to refer both to the vertex and state variable at the vertex. The naive method solves the same system in (1) for $A^{(s)}$ instead of A. That is, for $i \in V^{(s)}$, z_i^* is the steady-state of the dynamical system

$$\text{Naive method:} \quad \dot{z}_i = f(z_i) + \sum_{j \in V^{(s)}} A_{ij}^{(s)} g(z_i, z_j). \quad (2)$$

This naive approach produces incorrect conclusions, yet it is common practice because that's all practitioners have. To get correct results, one *must* account for missing data.

3 Mean Field Approximation and Resilience

The main idea is shown in the (sampled) subgraph on the right. We focus on one vertex in $V^{(s)}$, z_2. Vertex z_2 interacts with its neighbors in the subgraph, z_1 and z_4, and its neighbors outside the subgraph. All the subgraph nodes $z_1, \ldots, z_5$ are in a similar situation. Now fix the value for each external neighbor of z_2 to its *true steady-state value in the full network*, shown as x_1^*, x_2^*, x_3^*. Do the same for the external neighbors of all the subgraph vertices $z_1, \ldots, z_5$. As far as the subgraph is concerned, all external neighbors have converged to their steady-state values and are providing the right interactive feedback to all subgraph nodes. The subgraph is effectively isolated from the rest of the network and will converge to steady-states of the full network. We state the next theorem without proof. The preceding discussion essentially amounts to the proof.

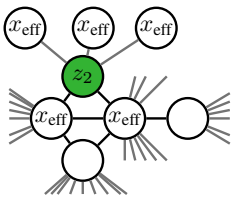


Theorem 1. *The steady-states z_i^* of the dynamical system*

$$\dot{z}_i = f(z_i) + \sum_{j \in V^{(s)}} A_{ij}^{(s)} g(z_i, z_j) + \sum_{j \notin V^{(s)}} A_{ij} g(z_i, x_j^*) \quad (3)$$

recover one of the true steady-states of the full network.

We cannot implement(3), because x_i^* are unknown. In the mean field approximation, we replace x_i^* by an average influence x_{eff}. Of course it is an approximation, but it works well for analyzing complex interacting systems such as spin-systems (Edwards and Anderson 1975). The iteration-0 estimate is x_{eff} for all states.



Fix the states for all vertices but (say) z_2 to x_{eff}, and find the steady-state for z_2, illustrated on the right. Now, z_2 is effectively isolated from the rest of the network as all its neighbors are *fixed* at x_{eff}. z_2 evolves to a steady-state, following the dynamics $\dot{z}_2 = f(z_2) + 5g(z_2, x_{\text{eff}})$. Repeat for each subgraph-vertex to arrive at $z_i^{(1)}$, the steady-states for

$$\dot{z}_i = f(z_i) + \delta_i g(z_i, x_{\text{eff}}), \quad (4)$$

where δ_i is the degree of z_i in G. These equations are uncoupled since x_{eff} is *fixed*. This is the method used in (Gao, Barzel, and Barabási 2016) to analyze the dynamics on the full network by reducing to n uncoupled ODEs. We now iterate further. Suppose the steady-state from iteration τ is $z_i^{(\tau)}$. We obtain $z_i^{(\tau+1)}$, the approximation at iteration $\tau + 1$ as the steady-state solution to the *uncoupled* equations

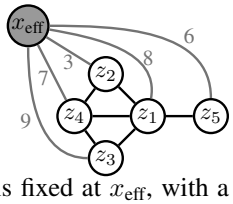
$$\dot{z}_i = f(z_i) + \textstyle\sum_{j \in V^{(s)}} A_{ij}^{(s)} g(z_i, z_j^{(\tau)}) + \sum_{j \notin V^{(s)}} A_{ij} g(z_i, x_{\text{eff}}). \quad (5)$$

Comparing (5) to the exact solution in (3), the external forces are replaced by an effective external force and the interaction term is approximated by an interaction to a previous steady-state. Iterating to convergence, $z_i^{(\tau)}$ converges to a steady-state z_i^* which solves the coupled system

$$\dot{z}_i = f(z_i) + \textstyle\sum_{j \in V^{(s)}} A_{ij}^{(s)} g(z_i, z_j) + (\delta_i - \delta_i^{(s)}) g(z_i, x_{\text{eff}}), \quad (6)$$

where $\delta_i^{(s)}$ is the degree of z_i in $G^{(s)}$.

The naive method in (2) resembles (6) with one crucial difference, an additional term to account for the external force on a vertex. The entire system in (6) corresponds to adding just one more vertex to our subgraph, whose value is fixed at x_{eff}, with a weighted edge to z_i of weight $A_{i,x_{\text{eff}}}^{(s)} = \delta_i - \delta_i^{(s)}$. We show this augmented graph for our example subgraph on the right.



To account for missing vertices, add *one* vertex to the subgraph, fixed to x_{eff} and add degree-weighted edges from x_{eff} to all vertices in the subgraph.

Next, we discuss how to compute x_{eff} to estimate the mean-field interaction with unseen vertices. The complication is that this estimate cannot depend on the missing information. This is possible because x_{eff} depends on the missing information only through global topological statistics of the network, and we can estimate those topological statistics when the subgraph is sampled appropriately.

3.1 Evaluating the Mean-Field Approximation

One non-trivial implication of our mean-field approach is that the steady-states are approximated by solving the uncoupled equations in ~~(3)~~ (4). The parameter x_{eff} in ~~(3)~~ (4) only depends on $\beta = \langle \delta^2 \rangle / \langle \delta \rangle$. So the ODE in ~~(3)~~ (4) only depends on the degree sequence of the original network, which means the steady-states can be approximated by knowing only a network's degree sequence. We verify this in Figure 2 which compares the

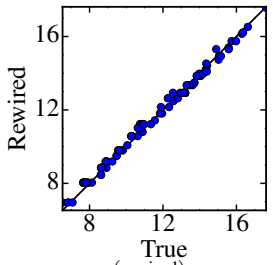


Figure 2: $G^{(\text{rewired})}$ vs. G.

true steady-states in the ecological network with the steady-states in a random network that preserves the degree sequence. The near-perfect matching of the steady-states provides ~~strong confirmation~~ **empirical evidence** of our mean-field approach.

3.2 Computing the Effective External Impact

There are two unknown quantities in (6). The degree δ_i and x_{eff}. We now discuss x_{eff}, following the general path in (Gao, Barzel, and Barabási 2016). Focus on a vertex i and the interaction term $\sum_j A_{ij} g(x_i, x_j)$ in (1), where A_{ij} is the influence j has on i. Similarly, i influences j with a weight A_{ji}. We define the in-degree $s_i^{\text{in}} = \sum_j A_{ij}$ and the out-degree $s_i^{\text{out}} = \sum_j A_{ji}$. Assuming $A_{ij} \geq 0$, the interaction term

$$\sum_j A_{ij} g(x_i, x_j) = s_i^{\text{in}} \frac{\sum_j A_{ij} g(x_i, x_j)}{\sum_k A_{ik}} \tag{7}$$

can be rewritten as the in-degree times an average interaction. Here, the in-degree s_i^{in} captures the idiosyncratic part, and the average $g(\cdot, \cdot)$ captures the network effect. Our first mean-field approximation is to replace local averaging with global averaging, which approximates the network-impact on a vertex as nearly homogeneous. Specifically, we have

$$\frac{\sum_j A_{ij} g(x_i, x_j)}{\sum_k A_{ik}} \approx \frac{\sum_{ij} A_{ij} g(x_i, x_j)}{\sum_{ik} A_{ik}} = \frac{\mathbf{1}^T A g(x_i, \boldsymbol{x})}{\mathbf{1}^T A \mathbf{1}}, \tag{8}$$

where the vector $g(x_i, \boldsymbol{x})$ has jth component $g(x_i, x_j)$. Define an averaging linear operator

$$\mathcal{L}_A(\boldsymbol{z}) = \frac{\mathbf{1}^T A}{\mathbf{1}^T A \mathbf{1}} \boldsymbol{z} = \frac{\boldsymbol{s}^{\text{out}} \cdot \boldsymbol{z}}{\boldsymbol{s}^{\text{out}} \cdot \mathbf{1}}, \tag{9}$$

which is a weighted average of the entries in $\boldsymbol{z}$. Our mean-field approximation results in the approximate dynamics

$$\dot{x}_i = f(x_i) + s_i^{\text{in}} \mathcal{L}_A[g(x_i, \boldsymbol{x})]. \tag{10}$$

In the first order linear approximation, we take $\mathcal{L}_A$ inside g. Our second mean-field approximation is that the average of external interactions is approximately the interaction with the average. That is $\mathcal{L}_A[g(x_i, \boldsymbol{x})] \approx g(x_i, \mathcal{L}_A(\boldsymbol{x}))$ and

$$\dot{x}_i = f(x_i) + s_i^{\text{in}} g(x_i, \mathcal{L}_A(\boldsymbol{x})), \tag{11}$$

where $\mathcal{L}_A(\boldsymbol{x})$ is a global state. Let $x_{\text{av}} \triangleq \mathcal{L}_A(\boldsymbol{x})$, and apply $\mathcal{L}_A$ to both sides of (11) gives

$$\dot{x}_{\text{av}} = \mathcal{L}_A[f(\boldsymbol{x})] + \mathcal{L}_A[\boldsymbol{s}^{\text{in}} g(\boldsymbol{x}, x_{\text{av}})]. \tag{12}$$

We assume that the in-degrees s_i^{in} and the interactions $g(x_i, x_{\text{av}})$ are roughly uncorrelated, so the $\mathcal{L}_A$-average of the product is the product of $\mathcal{L}_A$-averages (Gao, Barzel, and Barabási test this assumption extensively). Thus, our third mean-field approximation is $\mathcal{L}_A[\boldsymbol{s}^{\text{in}} g(\boldsymbol{x}, x_{\text{av}})] \approx \mathcal{L}_A(\boldsymbol{s}^{\text{in}}) \mathcal{L}_A[g(\boldsymbol{x}, x_{\text{av}})]$. Using the linear approximation again, we take the $\mathcal{L}_A$-average inside f and g

$$\dot{x}_{\text{av}} = f(\mathcal{L}_A(\boldsymbol{x})) + \mathcal{L}_A(\boldsymbol{s}^{\text{in}}) g(\mathcal{L}_A(\boldsymbol{x}), x_{\text{av}}). \tag{13}$$

Now we have a dynamical system for , x_{av},

$$\dot{x}_{\text{av}} = f(x_{\text{av}}) + \beta g(x_{\text{av}}, x_{\text{av}}), \tag{14}$$

where the resilience $\beta = \mathcal{L}_A(\boldsymbol{s}^{\text{in}})$. For undirected graphs, $\beta = \sum_i \delta_i^2 / \sum_i \delta_i = \langle \delta^2 \rangle / \langle \delta \rangle$. The steady-state of (14) is the external effective impact, x_{eff}. Plugging it into (11) gives an uncoupled ODE for x_i,

$$\dot{x}_i = f(x_i) + s_i^{\text{in}} g(x_i, x_{\text{eff}}). \tag{15}$$

In the mean-field approximation, $g(x_i, x_j)$ in (1) is replaced by an interaction with a mean-field external world $g(x_i, x_{\text{eff}})$ and the number of neighbors impacting x_i is captured by s_i^{in}. To approximately obtain the steady-states of the system, one first solves the ODE in (14) to get x_{eff}, and then n uncoupled ODEs at each vertex to get x_i, which only depends on s_i^{in} if given x_{eff}. The method works well because the mean-field approximations only need to hold *at the steady-state*. Hence, we can recover the steady-state for any vertex (for example the sampled vertices) from accurate estimates of degrees s_i^{in} (δ_i in the undirected case) and the resilience parameter β.

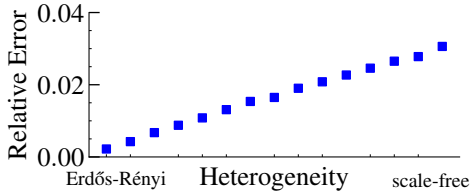
3.3 Accuracy of Our Approach

The mean-field approximation essentially replaces individual interactions with an average, and amounts to a homogeneity assumption. The more homogeneous a network, the more accurate our approximations. Indeed, the method of solving for x_{eff} as a steady state of (14) and then for x_i^* as steady states of (15) produces an exact solution for a regular network (perfectly homogeneous). The next theorem says that our method in (6) is perfect for such regular networks.

Theorem 2. *For a k-regular network, the steady-states z_i^* obtained by solving the dynamical system in* (6) *with x_{eff} obtained as a steady-state of* (14) *with $\beta = k$ recovers an exact steady-state x_i^*.*

Proof. (Sketch) x_{eff} is a steady state of (14) with $\beta = k$, hence $f(x_{\text{eff}}) + k g(x_{\text{eff}}, x_{\text{eff}}) = 0$ as $\dot{x} = 0$. We show that $x_i^* = x_{\text{eff}}$ for $i \in [1, N]$ is a fixed point of the system. Since node i has k neighbors and each of them has state $x_j^* = x_{\text{eff}}$, $\dot{x}_i = f(x_i) + \beta g(x_i, x_{\text{eff}}) = 0$ when $x_i = x_{\text{eff}}$. Lastly, (6) converges to (14), because $x_i = x_{\text{eff}}$. ■

Essentially, our approach is perfect for regular networks. The degree of inhomogeneity in the network is therefore a parameter which controls the quality of approximation. We now show some experimental results with synthetic networks where we can control for the inhomogeneity. Even for extremely inhomogeneous networks, our approach suffers little, indicating the strength of the mean-field approach.



To evaluate the impact of heterogeneity, we use 15 random 1000-vertex networks with different heterogeneities $\mathcal{H}$ measured by the relative degree dispersion, $\mathcal{H} \triangleq [\langle \delta^2 \rangle - \langle \delta \rangle^2] / \langle \delta \rangle$. Relative state estimation errors for our method

in (6) for an observed subgraph of just 10 vertices and estimates of β and x_{eff} from Section 4 are shown below, As expected, performance deteriorates as heterogeneity increases. The far right is the very heterogeneous scale-free network and leftmost is the nearly homogeneous Erdős-Rényi network. Even for very heterogeneous networks, the relative error is only about 3%.

4 Estimating the Resilience

To get x_{eff}, we solve for the steady-states of (14), so we need to estimate the resilience $\beta = \langle\delta^2\rangle/\langle\delta\rangle$, a topological statistic of the full network. Resilience is well studied in science and engineering, arising in many contexts because it captures a complex system's ability to retain its basic functionality under faults (Gao, Barzel, and Barabási 2016). Understanding a network's resilience is essential for us to evade the consequences of resilience loss, such as malfunction of gene regulation networks, cascading failures in technological systems, mass extinctions in ecological networks.

We estimate resilience β from the sampled subgraph $G^{(s)}$. In other contexts where it is important to measure resilience, the full network topology is also often not available perhaps due to privacy, or, as is usually the case in practice, the inability to measure the full network (for example, protein-protein interactions, metabolic and terrorist networks (Stumpf and Wiuf 2005)). Despite advances in graph sampling, we are not aware of accurate estimators of resilience from an incomplete view of the network. The naive resilience-estimator treats the observed subnetwork as the full network and estimates β with $\beta^{(s)}$. We derive corrections to this naive estimate for a variety of sampling schemes, and give analysis of the estimation accuracy (bias and variance) and the sample complexity. This general infrastructure enables one to manage a network's resilience from incomplete data, and may be of independent interest.

We treat resilience estimation for undirected, unweighted graphs. However, our results can be easily extended to the directed and weighted cases. This part of our work falls into the general area of graph analysis from incomplete (sampled) graphs. Typical sampling methods are vertex-based, exploration-based (see for example (Leskovec and Faloutsos 2006; Hübler et al. 2008; Ahmed, Neville, and Kompella 2011)) and edge-based. We focus on the first two sampling schemes since they are natural ways to sample a graph in practice, however, the results do extend to edge-based sampling. The birds-eye view of the workflow is

$$(G, \beta) \xrightarrow{\text{sampler } \mathbf{\Phi}} (G^{(s)}, \beta^{(s)}) \xrightarrow{\text{this work}} \hat{\beta}(G^{(s)}, \beta^{(s)}, \mathbf{\Phi}) \approx \beta,$$

where $\beta^{(s)}$ is the naive resilience estimator that treats $G^{(s)}$ as if it were the complete network. Our final estimator $\hat{\beta}$ can depend on the sampling method $\mathbf{\Phi}$.

4.1 Estimating β for Random Subgraphs

We consider several types of subgraph sampling. The simplest is to sample m vertices uniformly without replacement and measure the degree δ_i of each vertex. The sample averages $\langle\delta^2\rangle_s$ and $\langle\delta\rangle_s$ are unbiased and concentrate at the true averages $\langle\delta^2\rangle$ and $\langle\delta\rangle$. This means $\hat{\beta} = \langle\delta^2\rangle_s/\langle\delta\rangle_s$ is asymptotically unbiased and concentrates at β. We use $\langle\cdot\rangle_s$ to denote the average over the subgraph vertices in $V^{(s)}$.

An interesting variant is to sample vertices with a degree-bias, so the probability to sample vertex i is $\delta_i/\sum_i \delta_i$. In this case, the expected degree of a sampled node is $\sum_i \delta_i^2/\sum_i \delta_i = \beta$, hence $\hat{\beta} = \langle\delta\rangle_s$ is an unbiased estimate of β. More generally, one can sample vertices with an arbitrary degree-biased importance distribution $q(\delta)$. The analysis of this more general case, including a bias-variance analysis is given in the supplementary material.

Now, suppose you only measure $\delta_i^{(s)}$, the induced degrees in the subgraph, not true degrees. To implement (6) we need not only β, but also the degree δ_i. Let $\beta^{(s)}$ be the resilience of the induced subgraph, $\beta^{(s)} = \langle\delta^2\rangle_s/\langle\delta\rangle_s$. A crude analysis just uses concentration twice. First, the induced degree $\delta_i^{(s)}$ is a sum of $m-1$ Bernoulli random variables sampled uniformly without replacement from a population of $n-1$ Bernoulli values in which δ_i of them are 1. So, $\delta_i^{(s)}/(m-1)$ concentrates at $\delta_i/(n-1)$. We need concentration for each of the m sampled vertices, which, using a union bound plus a Hoeffding inequality, has a failure probability $2m\exp(-\Omega(m\varepsilon^2))$ for relative error ε. So, the estimates $\hat{\delta}_i = \frac{n-1}{m-1}\delta_i^{(s)}$ all concentrate with relative error at their respective δ_i and hence we get a resilience estimate $\hat{\beta} = \langle\hat{\delta}^2\rangle/\langle\hat{\delta}\rangle = \frac{n-1}{m-1}\beta^{(s)}$ that concentrates at β. A more refined estimate based on $\mathbb{E}[(\delta_i^{(s)})^2]$ is $\hat{\beta} = \frac{n-2}{m-2}\beta^{(s)} - \frac{n-m}{n-2}$.

Another popular way to sample a subgraph is using a random walk: start at a random vertex and move from one vertex to a neighbor, chosen uniformly at random (nodes could be revisited). Such a walk has a stationary distribution where a node's sampling probability is proportional to its degree (Kurant, Markopoulou, and Thiran 2011). From the discussion of degree-biased sampling, the estimator of resilience is $\hat{\beta} = \langle\delta\rangle_s$ when the degrees in the full network are known. When only the induced subgraph is known, then, as with the induced subgraph from vertex sampling, a correction factor is needed. We approximate the sampling as independent degree-biased sampling in a random rewiring model, and get a correction factor n/m (see supplementary material). We summarize our discussion of random vertex sampling (VS) and random walks (RW) in a table.

Sampling	Measured	$\hat{\delta}_i$	$\hat{\beta}$
VS	$V^{(s)}, E^{(s)}, \delta_i$	δ_i	$\langle\delta^2\rangle_s/\langle\delta\rangle_s$
Ind-VS	$V^{(s)}, E^{(s)}, \delta_i^{(s)}$	$\frac{n-1}{m-1}\delta_i^{(s)}$	$\frac{n-2}{m-2}\beta^{(s)} - \frac{n-m}{n-2}$
RW	$V^{(s)}, E^{(s)}, \delta_i$	δ_i	$\langle\delta\rangle_s$
Ind-RW	$V^{(s)}, E^{(s)}, \delta_i^{(s)}$	$\frac{n\langle\delta\rangle}{m\hat{\beta}}\delta_i^{(s)}$	$\frac{n}{m}\langle\delta\rangle_s$

For $i \in V^{(s)}$, δ_i is the degree in G, $\delta_i^{(s)}$ is the degree in $G^{(s)}$ and the resilience of the subgraph is $\beta^{(s)} = \langle\delta^2\rangle_s/\langle\delta\rangle_s$. The notation $\langle x\rangle_s$ means $\sum_{i\in V^{(s)}} x_i/m$.

For Ind-RW, the network average degree $\langle\delta\rangle$ is needed to estimate true degrees (in addition to n). There are methods to estimate $\langle\delta\rangle$ (Dasgupta, Kumar, and Sarlos 2014; Zhang, Kolaczyk, and Spencer 2015; Ribeiro and Towsley 2010;

Applications	Networks ($\|V\|, \|E\|$)
Ecological	**ENet1**(270,8074) (Arroyo, Armesto, and Primack 1985; Gao, Barzel, and Barabási 2016)
	ENet2(97,972) (Clements and Long 1923; Gao, Barzel, and Barabási 2016)
Gene Regulation	**MEC**(2268,5620) (Lee et al. 2002; Gao et al. 2014)
	TYA(662,1062) (Lee et al. 2002; Gao et al. 2014)
Epidemic	**Dublin**(410,2765) (Rossi and Ahmed 2015)
	Email(1133,5451) (Guimera et al. 2003; Kunegis 2013)

Table 2: List of networks in our evaluation

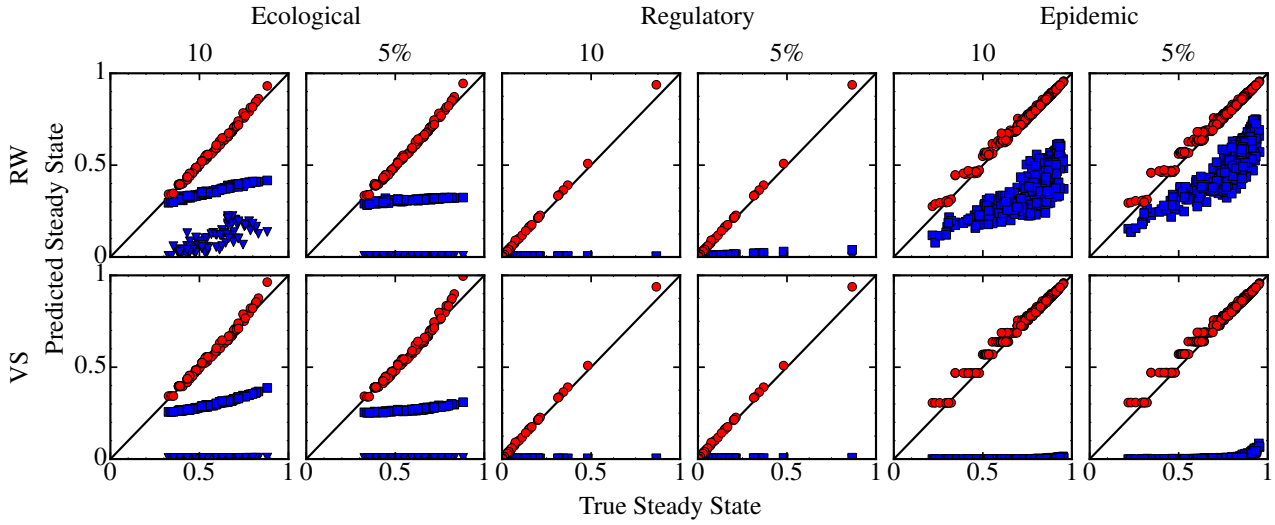


Figure 3: Predicted steady states when vertex degrees δ_i are available. The diagonal line is perfect prediction. Our method (red) is nearly perfect, while the naive method (blue) is a disaster. Often, the steady states just converge to 0 for the naive method.

Leskovec and Faloutsos 2006), but they use more powerful queries than the induced subgraph from a simple random walk can provide. The complication with the simple random walk is that it is a degree-biased sampling of vertices. This bias can be corrected with a Metropolis-Hastings random walk (Hübler et al. 2008), but that requires knowledge of vertex degrees.

One can analyze the bias and variance of estimators using the importance sampling framework that samples vertices according to the proposal distribution q (Theodoridis 2015; Cochran 1977; Wu 1982) (uniform or degree-biased). We give the details in the supplementary material.

5 Results

We tested our approach on the three popular dynamical systems in Table 1 and two corresponding networks for each dynamical system (see Table 2).

Each dynamical system contains several parameters which are set as in (Gao, Barzel, and Barabási 2016) for ecology and gene regulation, and as in (Barzel and Barabási 2013) for epidemics. We compare the performance of different subgraph-sampling (RW, VS, Ind-RW, Ind-VS), with sample size $m = 10$ (constant size) and $m = 0.05n$ (constant fraction). We average results over several subgraphs. When a vertex is sampled in multiple subgraphs, we report the average predicted steady-state. We also test the performance of our approach on different sample sizes in the Supplementary Material.

Given a subgraph, we solve (6) to estimate the true steady-states of vertices in the subgraph, for which we need δ_i (true degree) and x_{eff} (effective external state). To get x_{eff}, we need β to solve (14). For δ_i and β, we use the estimates in (4.1). In solving (14), there can be one stable attractor or more than one stable attractors (there can also be unstable attractors). When there are more than one stable attractor, the system can equilibrate at multiple steady-states.

First, we consider sampling methods that obtain vertex degrees δ_i (RW, VS), where only β needs to be estimated. The results in Figure 3 show that our approach (red) is remarkable at revealing the true steady-state, even from tiny subgraphs. We get near perfect results from just 10 vertices of a multi-thousand vertex graph. This is all the more impressive given that the current state-of-the-art, i.e., the naive approach (blue), is more or less a disaster. We emphasize that the naive method is the only method currently used by researchers in the field, and it simply fails. In the ecological application, the naive method even identifies multiple steady-states, one of which sends all species to extinction.

When only induced subgraphs are available (Ind-RW, Ind-VS), see Figure 4, our methods are still much better than naive, but performance drops. Estimating individual degrees from induced degrees is hard. Parameters like the resilience

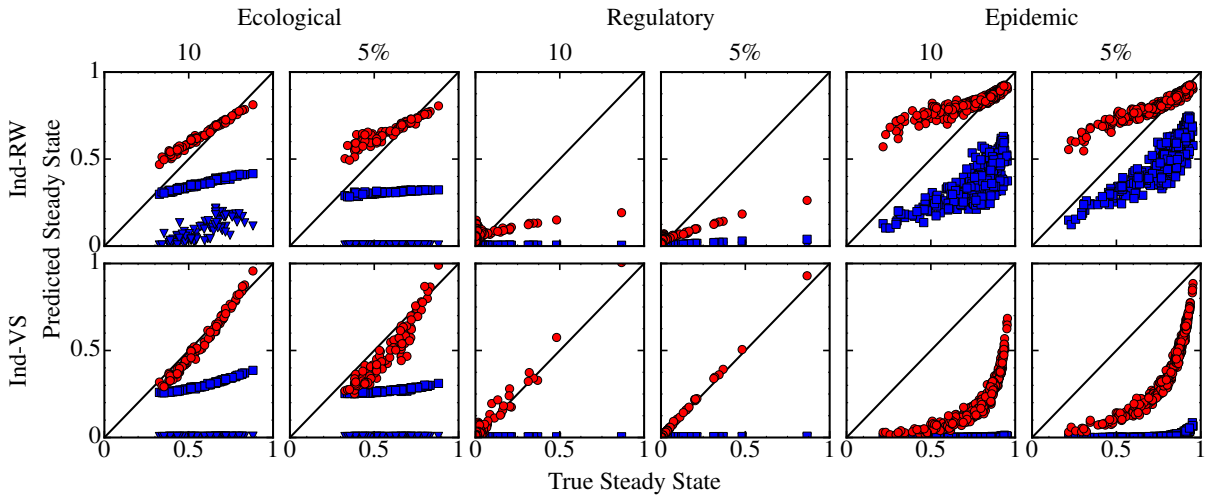


Figure 4: Predicted steady-states when only the induced subgraph is available and vertex degrees must be estimated. Our method (red) significantly outperforms naive, but performance drops compared to Figure 3, because estimating vertex degrees from observed induced degrees is tough.

β are global and can be extracted accurately from subgraphs. Local parameters like vertex degree get severely distorted. More details on how degree and resilience estimations are affected by induced subgraph sampling is in the supplementary material. For vertex sampling, our estimator is unbiased, but for random walks, our estimators are based on the approximation of independent degree-biased sampling. This approximation can break down.

Open Question. How should one estimate vertex degrees from induced subgraphs of random walks? How does the estimator depend on the network properties?

We also observe from Figure 4 that for induced subgraphs, the performance of our approach (and the naive method) depends strongly on the subgraph sampling methods, depending on the network structure and the nonlinear dynamics.

Open Question. What are the factors which influence the choice of subgraph sampling method? For example when is RW better than VS?

6 Discussion

We addressed a prevalent problem. Consider this scenario. A biologist has the favorite part of an ecosystem, their favorite 10-species, and carefully collects their relationships which are summarized in the adjacency matrix $A^{(s)}$. The biologist even knows how species interact, the dynamical system (1). The biologist carefully simulates the system to steady-state and finds that all species are going extinct. This is scary, but the result is just plain wrong. You cannot restrict a coupled nonlinear system to your favorite part of the network and expect even close to correct conclusions by just analyzing that part in isolation. One solution is to collect the full network and analyze the full system. There are two problems. First, we can't collect the full network. Second, simulating the full system to equilibrium is prohibitive in terms of convergence time. So the only feasible solution for learning the true steady states of the observed incomplete network is to somehow account for the external impact on the local system. This was our approach. In a mean field approximation, the external impact reduces to a single parameter x_{eff} which depends only on the network's resilience β, a topological parameter. We showed how to estimate resilience, depending on how the subgraph is sampled. Our results on real networks with corresponding dynamics gave spectacular success – we accurately recover steady-states from just 10 vertex subgraphs of thousand vertex networks.

There are several interesting future questions. The natural one is to find improved estimates of resilience that extend to other sampling methods (snowball sampling, edge sampling etc). A critical direction, which we address in forthcoming work, is the inverse problem. Suppose the steady-states are known. For example, the abundance of your 10 favorite monkey species is known. Can one infer the correct dynamical system $f(\cdot), g(\cdot,\cdot)$? Currently, the dynamics are fit to the observed steady-states for the partial network (Ghahramani and Roweis 1999; Schmidt and Lipson 2009; Ionides, Bretó, and King 2006; Bongard and Lipson 2007; Brunton, Proctor, and Kutz 2016). This is wrong and will produce the incorrect dynamics. It is no surprise, therefore, that the inferred dynamics keeps changing as more data is collected (Schmidt and Lipson 2009; Ionides, Bretó, and King 2006; Bongard and Lipson 2007; Brunton, Proctor, and Kutz 2016). It is absolutely necessary to account for the external impact in the inference process.

References

Ahmed, N.; Neville, J.; and Kompella, R. R. 2011. Network sampling via edge-based node selection with graph induction. *Computer Science Technical Reports*.

Allee, W. C.; Park, O.; Emerson, A. E.; Park, T.; Schmidt,

K. P.; et al. 1949. Principles of animal ecology. Technical report, Saunders Company Philadelphia, Pennsylvania, USA.

Alon, U. 2006. *An introduction to systems biology: design principles of biological circuits*. Chapman and Hall/CRC.

Arroyo, M. T. K.; Armesto, J. J.; and Primack, R. B. 1985. Community studies in pollination ecology in the high temperate andes of central chile ii. effect of temperature on visitation rates and pollination possibilities. *Plant Systematics and Evolution* 149(3-4):187–203.

Barzel, B., and Barabási, A.-L. 2013. Universality in network dynamics. *Nature Physics* 9(10):673.

Bongard, J., and Lipson, H. 2007. Automated reverse engineering of nonlinear dynamical systems. *PNAS* 104(24):9943–9948.

Brunton, S. L.; Proctor, J. L.; and Kutz, J. N. 2016. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *PNAS* 113(15):3932–3937.

Clements, F. E., and Long, F. L. 1923. *Experimental pollination: an outline of the ecology of flowers and insects*. Number 336. Carnegie Institution of Washington.

Cochran, W. G. 1977. Sampling technique.

Dasgupta, A.; Kumar, R.; and Sarlos, T. 2014. On estimating the average degree. In *Proc. Conf. WWW*, 795–806.

Dodds, P. S., and Watts, D. J. 2005. A generalized model of social and biological contagion. *Journal of Theoretical Biology* 232(4):587–604.

Edwards, S. F., and Anderson, P. W. 1975. Theory of spin glasses. *Journal of Physics F: Metal Physics* 5(5):965.

Gao, J.; Barzel, B.; and Barabási, A.-L. 2016. Universal resilience patterns in complex networks. *Nature* 530(7590):307.

Gao, J.; Liu, Y.-Y.; D'souza, R. M.; and Barabási, A.-L. 2014. Target control of complex networks. *Nature Communications* 5:5415.

Ghahramani, Z., and Roweis, S. T. 1999. Learning nonlinear dynamical systems using an em algorithm. In *Advances in NIPS*, 431–437.

Gjoka, M.; Kurant, M.; and Markopoulou, A. 2013. 2.5 k-graphs: from sampling to generation. In *INFOCOM'13*, 1968–1976. IEEE.

Guimera, R.; Danon, L.; Diaz-Guilera, A.; Giralt, F.; and Arenas, A. 2003. Self-similar community structure in a network of human interactions. *Phys. Rev. E* 68(6):065103.

Hübler, C.; Kriegel, H.-P.; Borgwardt, K.; and Ghahramani, Z. 2008. Metropolis algorithms for representative subgraph sampling. In *ICDM'08*, 283–292. IEEE.

Hufnagel, L.; Brockmann, D.; and Geisel, T. 2004. Forecast and control of epidemics in a globalized world. *PNAS* 101(42):15124–15129.

Hui, C. 2006. Carrying capacity, population equilibrium, and environment's maximal load. *Ecological Modelling* 192(1-2):317–320.

Ionides, E. L.; Bretó, C.; and King, A. A. 2006. Inference for nonlinear dynamical systems. *PNAS* 103(49):18438–18443.

Karlebach, G., and Shamir, R. 2008. Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology* 9(10):770.

Katzir, L.; Liberty, E.; and Somekh, O. 2011. Estimating sizes of social networks via biased sampling. In *Proc. Conf. WWW*, 597–606. ACM.

Klusowski, J. M., and Wu, Y. 2018. Counting motifs with graph sampling. In Bubeck, S.; Perchet, V.; and Rigollet, P., eds., *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, 1966–2011. PMLR.

Kunegis, J. 2013. KONECT: The Koblenz Network Collection. In *Proc. Conf. WWW*, 1343–1350. ACM.

Kurant, M.; Butts, C. T.; and Markopoulou, A. 2012. Graph size estimation. *arXiv preprint arXiv:1210.0460*.

Kurant, M.; Markopoulou, A.; and Thiran, P. 2011. Towards unbiased BFS sampling. *IEEE Journal on Selected Areas in Communications* 29(9):1799–1809.

Lee, T. I.; Rinaldi, N. J.; Robert, F.; Odom, D. T.; Bar-Joseph, Z.; Gerber, G. K.; Hannett, N. M.; Harbison, C. T.; Thompson, C. M.; Simon, I.; et al. 2002. Transcriptional regulatory networks in Saccharomyces cerevisiae. *Science* 298(5594):799–804.

Leskovec, J., and Faloutsos, C. 2006. Sampling from large graphs. In *Proc. Conf. ACM SIGKDD*, 631–636. ACM.

Lotka, A. J. 1910. Contribution to the theory of periodic reactions. *The Journal of Physical Chemistry* 14(3):271–274.

Pastor-Satorras, R., and Vespignani, A. 2001. Epidemic spreading in scale-free networks. *Phys. Rev. Lett.* 86(14):3200.

Ribeiro, B., and Towsley, D. 2010. Estimating and sampling graphs with multidimensional random walks. In *Proc. Conf. Internet Measurement*, 390–403.

Rossi, R., and Ahmed, N. 2015. The network data repository with interactive graph analytics and visualization. In *AAAI*.

Schmidt, M., and Lipson, H. 2009. Distilling free-form natural laws from experimental data. *Science* 324(5923):81–85.

Seshadhri, C.; Pinar, A.; and Kolda, T. G. 2014. Wedge sampling for computing clustering coefficients and triangle counts on large graphs. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 7(4):294–307.

Stumpf, M. P., and Wiuf, C. 2005. Sampling properties of random graphs: the degree distribution. *Phys. Rev. E* 72(3):036118.

Theodoridis, S. 2015. *Machine learning: a Bayesian and optimization perspective*. Academic Press.

Wu, C.-F. 1982. Estimation of variance of the ratio estimator. *Biometrika* 69(1):183–189.

Zhang, Y.; Kolaczyk, E. D.; and Spencer, B. D. 2015. Estimating network degree distributions under sampling: An inverse problem, with applications to monitoring social media networks. *Annals of Applied Statistics* 9:166–199.