On the Success of Network Topology Inference using a Markov Random Walk Model for Nested Routing Policies

University of Wisconsin Tech Report ECE-10-02

Laura Balzano Robert Nowak
University of Wisconsin, Madison
{sunbeam}, {nowak}@ece.wisc.edu

Matthew Roughan
University of Adelaide, Australia
matthew.roughan@adelaide.edu.au

ABSTRACT

Network topology inference is a topic of ongoing interest, as researchers continue to try to identify the topology of the internet, other networks, and our own networks for maintenance purposes. In this paper we explain why a simple algorithm based on network co-occurrence measurements and a Markov random walk model for routing enables perfect topology reconstruction, despite the seeming model mismatch to real network routing. We show that topology measurement and inference based on this model allows detection of routing bugs, misconfiguration, or even routers that deliberately misreport routing data.

1. INTRODUCTION

Network topology identification has been studied extensively because of a great need for precise topology information in networks. In Internet measurement there are two main approaches: direct measurement using configuration files, traceroutes, or routing monitors (see Section 2), or indirect inference methods using measurements of packet traffic, such as delays and losses of packets [10, 11, 23], or co-occurences [17, 22]. The direct and indirect approaches are complementary. Direct approaches are preferred for layer-3 topology measurement, but often cannot see layer-2 topology. That issue is being compounded by the increasing deployment of technologies such as Multi-Protocol Label Switching (MPLS) that effectively hide the layer-3 topology. Indirect inference approaches have the potential to see topological structure that is hidden from direct measurements. Perhaps most importantly, these methods involve data-path packets rather than control-plane announcements; thus indirect inference can establish topology from the point of view of actual traffic and at a time-resolution commensurate with this traffic. Therefore, these methods have the promise to be able to rapidly detect problems that are hidden from more conventional control-plane measurements.

Despite their potential utility, indirect methods have not been widely utilized in real networks. In this paper we focus on a newly proposed method for topology inference– Network Inference from Co-Occurrences (NICO) [22]. Many indirect methods are limited to tree-like networks, and NICO overcomes this limitation. Additionally, topology identification methods often adopt simplifying models that do not necessarily reflect the true behavior of the underlying network. The NICO inference algorithm is not an exception; it models packet routing as a Markov random walk on the network graph. While this modeling assumption facilitates inference, it is markedly at odds with the deterministic routing strategies used in the vast majority of networks. The surprising thing is that this model results in exact reconstructions for shortest-path networks. A key contribution of this paper is to resolve this apparent contradiction mathematically.

We go further to show that NICO[1] will work for any network which uses a *nested routing policy*, wherein every subroute of a chosen route is also a route chosen by the policy. Shortest-path routing is a nested routing policy. In shortest-path routing, we know that if a shortest path route which traverses in order nodes `ABCD` then the paths `AB`, `BC`, `CD`, `ABC`, and `BCD` are all shortest-path routes as well. Under certain restrictions, BGP also produces nested routes.

NICO is designed for a particular type of topology identification problem where the input data are "co-occurences": an unordered set of network nodes that were active during a single communication event. Formally, we define a *co-occurrence* or *route-set* as the set of nodes along a route, where we omit the sequence or ordering of these nodes. Co-occurrences would arise from collecting samples of the same packet at multiple routers without having access to the order of those routers. These kinds of measurements are also relevant to biological network inference problems [19] and the reconstruction of sentences from bags-of-words [29].

In the context of the Internet, this problem formulation is becoming more relevant. Hash-based packet sampling would allow one to see the same packets as they transit the network, but without TTLs we cannot see the ordering. TTLs may be missing in flow level aggregates, or where key elements of the topology don't decrement TTLs, as may happen in layer 2 or MPLS networks.

[1]It is important to note that in [22], "NICO" refers to an EM algorithm for jointly estimating P and route-set orderings. In this paper we are using NICO to refer simply to the philosophy of using the maximum likelihood ordering of a route-set, given the transition matrix P or an estimate of P, to infer topology.

NICO uses collections of co-occurrence measurements to construct the topology of the network. Reconstruction of the network is perfectusing these data. Even when the Markov random walk parameters are also estimated from the data, we demonstrate (Section 5.1) that NICO's topology estimate has 0% missing links and typically $< 3\%$ false alarm links.

NICO can also be used to debug routing information. For instance, suppose a router claims that an ordered set of nodes `ABCD` is a route. The main result of this paper (Theorem 1) provides an immediate check whether the route is valid. Thus NICO could be used with multiple datasources, for instance BGP route announcements and AS-traceroutes [21], in order to detect prefix hijacking or other such divergences between data- and control-plane views of routes.

2. BACKGROUND

It is tempting to think that a well organized network operator will "just know" her network topology. However, much of the work on topology measurement was pioneered by network operators. In the complex, dynamic IP environment it is critically important to measure your network.

There are two main streams of research on network topology measurements and inference. The first concerns direct measurements. There are several possible approaches:

- *configs*: Configuration files control most aspects of a router, and so contain much information about the IP layer, and can be used to extract topologies [12, 13, 20]. However, privileged access is required to obtain such files.
- *route monitors:* participate in the distributed computations of routing protocols to information about the network topology, examples being OSPF [25,26], IS-IS [1,15], and BGP [2, 3]. These data can provide up to date information, but are also limited by the degree of privilege of the observer and the number of viewpoints available.
- *traceroute:* is a standard tool for estimating routes through an IP network [8,9,16,27,28]. Well known problems (non-atomicity, aliasing, sampling bias, e.g., see [6, 18]) plague traceroute measurements.

Limitations of the above approaches are that they focus on the IP layer, and in the first two cases the methods are predicated on correct operation of the network protocols and routers, which may be what we wish to discover (e.g. two routing protocol outages that crashed large parts of major networks for hours are described in [4,5,24]). Moreover, the control plane and data plane can diverge substantially [7].

2.1 Topology Identification using NICO

Indirect inference techniques don't use specific IP properties and thus have access to topological aspects that are hidden at layer 3. Most importantly, these methods involve data-path packets rather than control plane measurements; thus indirect inference can establish topology from the point of view of actual traffic.

The NICO approach [22] assumes that route-sets arose from a Markov random walk described by a node-to-node probability transition matrix P. Thus, the probability of a particular ordering of nodes is the product of the transition probabilities for that ordering, times the probability of starting at the initial node.

Using this model, and given a co-occurrence or route-set, we calculate a likelihood of each of the $n!$ orderings, given P or an estimate of P. Then for our inferred topology, we insert edges into the graph using the one most likely ordering. If we repeat this process for many co-occurrence measurements, we will get a more and more complete picture of the graph.

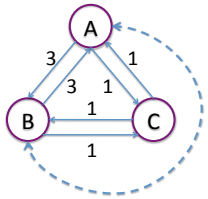
As we readily acknowledge, no real network manages packet traffic according to a Markov random walk model. Yet, we show that the model allows us to correctly infer topology in shortest-path networks or other networks that have a nested routing policy. This apparent paradox is part of the reason NICO is an interesting approach.

3. NOTATION

In this paper we consider a directed graph with no self-loops which represents a routing network: $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where the nodes of the graph $\mathcal{N}$ represent the set switches or routers in the network, and the edges of the graph $\mathcal{E}$ represent the directed links between the routers in the network. We write the adjacency matrix of the graph A, where $A_{ij} = 1$ iff $(i, j) \in \mathcal{E}$. We will talk of a network or a graph synonymously.

We define a path or route through either graph as a sequence of nodes connected by edges. Again we define a *route-set* or *co-occurrence* as the set of nodes along a route, where we omit the sequence or ordering of these nodes.

We define a link-route incidence matrix $R^{(ij)}$ for the route between nodes (i, j), which will be a $N \times N$ binary matrix, where $R^{(i,j)}_{km} = 1$ indicates that the nodes k, m occur in that sequence in the route between nodes i, j. If we take t_{ij} as the traffic from node i to node j, then the traffic weighted average of these matrices $\sum_{ij} t_{ij} R^{(ij)} = D$, a matrix of the link loads between nodes where there is a link.



Routes

Src	Dst	Route
A	B	A-C-B
A	C	A-C
B	A	B-C-A
B	C	B-C
C	A	C-A
C	B	C-B

Figure 1: Example network 1. The dashed line shows the route between A and B.

Figure 1 shows a simple example network with three nodes and directional link weights given. The shortest paths in this network are obvious, but shown in the figure for clear exposition. Given such a network and routing, and $t_{ij} = 1$ for all node pairs, then an example of a link-route incidence

matrix, and the link traffic matrix is

$$R^{(AB)} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad D = \begin{pmatrix} 0 & 0 & 2 \\ 0 & 0 & 2 \\ 2 & 2 & 0 \end{pmatrix},$$

where d_{ij} denotes the number of times the link from i to j is used (replacing indices with their alphabetic equivalent).

From D we can derive a row-stochastic transition matrix P for the whole network by normalizing each row of the matrix (dividing each row by its sum). In the example shown in Figure 1

$$P = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1/2 & 1/2 & 0 \end{pmatrix},$$

i.e. we create a Markov chain where the probability of going from node A to B (or vice versa) is zero (all packets from A or B goto C), but the probability of going from C to A and B are equal at 1/2.

We also use submatrices. For an unordered set of nodes X, P_X indicates the submatrix which contains rows and columns of nodes in X. The permutation matrix Π_X is such that $\Pi_X P_X \Pi_X^T$ is the submatrix of P containing the rows and columns associated with the nodes in X, permuted according to some conjectured ordering.

4. THE PROBLEM AND ITS SOLUTION

Now we can precisely define the problem of interest: given an unordered node-set for some route across a network, and the matrix P, can we find the correct ordering for the route-set to reconstruct the original route? We call this the "route-ordering" problem, and it turns out the answer is yes with very general conditions. The correctly ordered route can then be used to reconstruct the topology. NICO [22] provides a mechanism to estimate P, but in this section we consider P to be known.

The following theorems depend on subsets of these three conditions, which we list together for exposition purposes.

C1. The routers in the network of interest all use the same nested routing policy. A nested routing policy is one which will choose routes for which every subpath is also a chosen route.

C2a. There are no Multiple Equal-Cost (MEC) paths in the network.

C2b. If there are MEC paths in the network, the routing policy is shortest path and the path-cost metric is hop count. Ties must be broken in a deterministic way so that the routing remains nested.

C3. The first node in the true route is known. Denote it x_s.

Condition C1 is a familiar characteristic of shortest-path routing, summarized by saying "shortest paths are made up of shortest paths." However, nested routing is more general. For instance, other types of routing such as n-stratified shortest paths (using distributive route map functions) can create nested routing policies [14].

Conditions C2a and C2b are alternative conditions. In what follows, when we refer to Condition C2, it means that either C2a or C2b must hold. Condition C3 is natural in many problems; we are often enquiring about a route between a particular source and destination.

We start by giving several interesting results which will pave the way and provide intuition for Theorem 1.

LEMMA 1. *Under C1 and C2a, for any set of correctly ordered nodes, $X_c = (x_1, \cdots, x_n)$, the entries of the matrix $P_{x_i x_j}$ for $j > i + 1$ will be zero.*

Proof: We proceed by contradiction. Suppose that some entry $P_{x_k x_l}$ is non-zero for $l > k + 1$. Recall that a non-zero value in the $(x_k, x_l)^{th}$ entry of P implies that traffic flows from node x_k to node x_l in some route R. This in turn implies that the chosen routing path between x_k and x_l in route R is the direct link between them. But we assumed that X_c is correctly ordered according to the policy; so since in nested routing all sub-routes of X_c must also be routes chosen by the policy, $(x_k, x_{k+1}, \ldots, x_l)$ must also be the chosen routing path from x_k to x_l. Since there are no paths of equal cost, this is a contradiction. ■

LEMMA 2. *Under C1 and C2b the conclusion of Lemma 1 holds.*

Proof: As in Lemma 1 suppose that some entry $P_{x_k x_l}$ for $l > k + 1$ is non-zero. Again, $P_{x_k x_l} > 0$ implies that there is a direct link from k to l. C2b says that routing is done by minimum hop routing, and so the subroute $(x_k, x_{k+1}, \cdots, x_l)$ for $l > k + 1$ must cost more than the direct, single hop route (x_k, x_l), which is a contradiction. ■

These lemmas prove a simple fact: the matrix P has zeros where there isn't a link (or the link is unutilized). In nested routing networks, the position of these zeros has structure we can leverage. Where we might naively try to check all $n!$ permutations of the nodes on a route, instead, we can eliminate the vast majority because they have zero probability. The following results codify this intuition.

DEFINITION 1. *The* score *for an ordered route-set X_o is defined as $\prod_{(i,j)\in X_o} P_{ij}$.*

In NICO, scores of this form are associated with likelihood functions of particular orderings, and have been used to decide which ordering to prefer. Ideally the highest score would indicate the correct ordering, though this does not hold in general for any routing procedure.

However, a zero score does mean that an ordering is impossible, and we shall exploit this fact in what follows.

LEMMA 3. *Assume C1 and C2, then any node permutation Π of co-occurence X with positive score will have matrix $\Pi P_X \Pi^T$ in lower Hessenberg form with zeros on the diagonal.*

Proof: Without loss of generality consider the set of nodes $X = (1, 2, \ldots, n)$. Take $\widetilde{P} = \Pi P_X \Pi^T$, the permutation of P_X corresponding to a new ordering of X. Positive score for the ordering can only be attained when $\widetilde{P}_{ij} > 0$ for all (x_i, x_j) on the path. From Lemmas 1 and 2 we know that under conditions C1 and C2, $\widetilde{P}_{ij} > 0$ only if $j > i + 1$, and moreover $\widetilde{P}_{ii} = 0$ for all i because there are no self-loops in the graph. This is the definition of a lower Hessenberg matrix with zeros on the diagonal. ■

The implication is that, for some permutation Π, if the matrix $\Pi P_X \Pi^T$ is not lower Hessenberg, then the score of the ordering captured by Π is zero. We next consider the number of possible permutations resulting in a lower-Hessenberg matrix, and show that there are at most 2^{n-1} possible orderings rather than the $n!$ that we might naively expect.

LEMMA 4. *Under C1 and C2, there are at most 2^{n-1} permutations of a route-set $X = (x_1, \cdots, x_n)$ that have a non-zero score.*

Proof: From Lemma 3, any permutation with positive score must maintain that $P_{x_i x_j}$ for $j > i + 1$ will be zero. In general, an ordering X_o may have a non-zero score only if $j \leq i + 1$ and $j \neq i$, so either $j = i + 1$, i.e., j comes immediately after i, or $j < i$, so i comes after j. We can write these conditions as the following set of rules:

X_o either has x_1 directly before x_2, OR x_1 follows all nodes.

X_o either has x_2 directly before x_3, OR x_2 follows nodes $x_3, \cdots, x_n$.

$\vdots$

X_o either has node x_{n-1} directly before node x_n, OR x_{n-1} follows node x_n.

These rules are binary; there are 2^{n-1} ways to follow the rules and construct a new ordering X_o. Each choice will result in a unique ordering with possibly non-zero score, and all other permutations will have a zero score. ■

For an illustration of how the proof works, see Figure 2.

Now we have all the pieces to put together our main result, expressed in Theorem 1. We can see from the proof of Lemma 4 that adding condition C3 – knowledge of the source node – restricts the number of permutations with nonzero score to just one.

THEOREM 1. *Under assumptions C1-C3, and given a route-set $X = [x_1, \ldots, x_n]$, an ordering $\pi = [s, ...]$ corresponds to an ordering for a correct route in the network if and only if the score of the route $\{x_{\pi(1)}, x_{\pi(2)}, \ldots, x_{\pi(n)}\}$ according to the matrix P is nonzero.*

Proof: Certainly the forward direction holds: a correct ordering will correspond to a nonzero score according to P. For the other direction, since x_s must be at the beginning of the ordering, the binary choices in Lemma 4 are all decided; at no point can we put x_s after any other node. Thus there is only one choice, the ordering for the correct route. ■

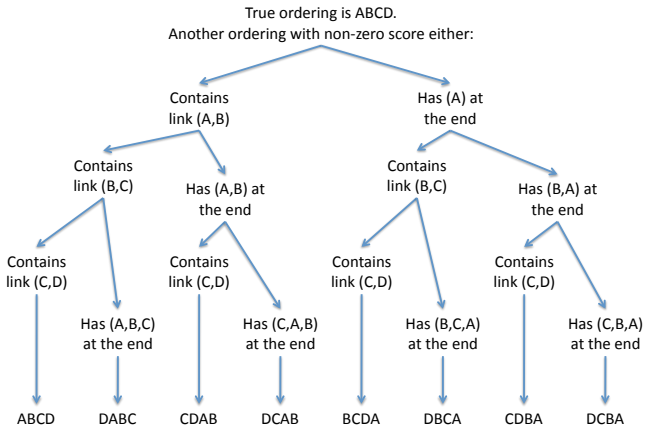


Figure 2: Example showing that there are 2^{n-1} possible reorderings of a given route-set.

Figure 2 illustrates the result – the only way to keep $x_s = A$ at the first position is to follow the leftmost path.

COROLLARY 1. *Under assumptions C1-C3, and given an unordered route-set X, if there is only one row of P_X which has a single non-zero entry, then the corresponding node is the source node x_s and the correct ordering can immediately be identified.*

Proof: The proof comes directly from the proofs of Lemma 3 and Theorem 1. ■

Discussion: The results above are straightforward, but there is an intriguing duality which results. Consider a clique with unit weights. All the routes will consist of one hop paths, and the matrix P would have no zeros, so it will not help us with route-ordering. However, because the network is fully connected, all route-sets will have a single pair of nodes, and so the correct ordering is obvious. As the paths get longer, the more zeros in P allow us to resolve the order despite the factorial growth in the number of permutations.

The proofs above are also predicated on knowledge of P. In practice P must be estimated from co-occurrence data; our simulations show that if we estimate D_{ij} simply by counting the number of times nodes i and j co-occur, and then normalize to get P, we get exact reconstruction.

5. EXAMPLES AND SIMULATIONS

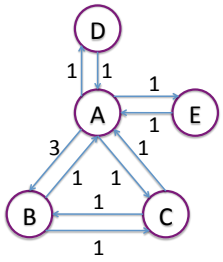
To see more clearly what is meant by the theory, we will present a few examples, starting with the one illustrated in Figure 1. Consider the route set `{A,B,C}`. There are six permutations of this route set, and Table 1 lays out these possibilities, along with a calculation of the product of the corresponding entries of P (shown in the second column). Note that the two possible routes `A-C-B` and `B-C-A` stand out, and that we cannot discriminate them unless we have information about the first node in the route.

In Example 1 all of the scores are zero except for the correct path and its reverse. This happens when routing is sym-

Ordering	Score	
	Example 1	Example 2
A–B–C	$0 \times 1 = 0$	$0 \times 1/4 = 0$
A–C–B	$1 \times 1/2 = 1/2$	$3/7 \times 4/7 = 12/49$
B–A–C	$0 \times 1 = 0$	$3/4 \times 3/7 = 9/28$
B–C–A	$1 \times 1/2 = 1/2$	$1/4 \times 3/7 = 3/28$
C–A–B	$1/2 \times 0 = 0$	$3/7 \times 0 = 0$
C–B–A	$1/2 \times 0 = 0$	$4/7 \times 3/4 = 3/7$

Table 1: The score calculations of the different orderings of the route set $\{A, B, C\}$ for Examples 1 and 2.

metric. Other paths use links that are never seen in the real network. Nested routing policies that don't have multiple equal-cost paths will select a tree of routes from a particular source, and so regardless of the degree of connectivity of the network, the link-count matrix for a particular route should be sparse (having $N-1$ non-zero entries for an $N \times N$ matrix), and in fact will be sparse in such a way that the scores of other orderings will be zero.



Routes		
Src	Dst	Route
A	B	A-C-B
A	C	A-C
A	D	A-D
A	E	A-E
B	A	B-A
B	C	B-C
B	D	B-A-D
B	E	B-A-E
C	A	C-A
C	B	C-B
C	D	C-A-D
C	E	C-A-E
D	A	D-A
D	B	D-A-C-B
D	C	D-A-C
D	E	D-A-E
E	A	E-A
E	B	E-A-C-B
E	C	E-A-C
E	D	E-A-D

Figure 3: Example network 2, with asymmetric weights (and hence asymmetric routes).

If the routing weights are not symmetric, many more possible orderings will have positive score, without information about the source node. In fact, the permutation with highest score is not necessarily the correct permutation. To see this, let us examine Example 2 shown in Figure 3.

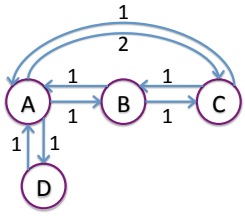
The link count matrix and transition matrix are as follows:

$$D = \begin{pmatrix} 0 & 0 & 6 & 4 & 4 \\ 3 & 0 & 1 & 0 & 0 \\ 3 & 4 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$P = \begin{pmatrix} 0 & 0 & 3/7 & 4/14 & 4/14 \\ 3/4 & 0 & 1/4 & 0 & 0 \\ 3/7 & 4/7 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The resulting scores for the co-occurrence {A,B,C} are shown in the third column of Table 1. Note that now the most likely route for route-set {A,B,C} is C–B–A, which does not occur at all in this network. Note that the critical feature of this example which pushes the score of a non-existent route above the scores of real routes is the asymmetry in the routes along with a heavily traversed link from B–A, which gives a high transition probability to B–A.

However again in this example, the only ordering with non-zero score which begins with the source node is the correct ordering. It is quite remarkable that this holds true in general.



Routes		
Src	Dst	Route
A	B	A-B
A	C	A-C or A-B-C
A	D	A-D
B	A	B-A
B	C	B-C
B	D	B-A-D
C	A	C-A
C	B	C-B
C	D	C-A-D
D	A	D-A
D	B	D-A-B
D	C	D-A-C or D-A-B-C

Figure 4: Example 3 with multiple equal-cost paths: The route from A to C could go directly or it could go through B.

The existence of multiple equal-cost (MEC) paths in the network causes a problem for us, and we see why in Figure 4. If the traffic was split evenly between the two equal-cost paths, we would have

$$D = \begin{pmatrix} 0 & 3 & 1 & 3 \\ 2 & 0 & 2 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 0 & 0 & 0 \end{pmatrix}$$

$$P = \begin{pmatrix} 0 & 3/7 & 1/7 & 3/7 \\ 1/2 & 0 & 1/2 & 0 \\ 2/3 & 1/3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

If the co-occurrence were {A,B,C}, we can see that no permutation of this submatrix gives a lower Hessenberg matrix. Not only that, but the most likely ordering would be C–A–B with score $2/3 \times 3/7 = 2/7$, as opposed to the correct A–B–C, which has score $3/7 \times 1/2 = 3/14$.

5.1 Simulation Results

These simulations show how well NICO can do in topology identification, using the true transition matrix P and also using an estimate of P based on empirical frequencies of co-occurrence. These unexpectedly accurate results were our motivation for developing the theory in Section 4.

Figure 5 shows the results on Rocketfuel topology 1239, with 33 nodes and 130 links, and Figure 6 shows the results on Rocketfuel topology 701, with 48 nodes and 368 links. All destination nodes are probed, and the results are shown

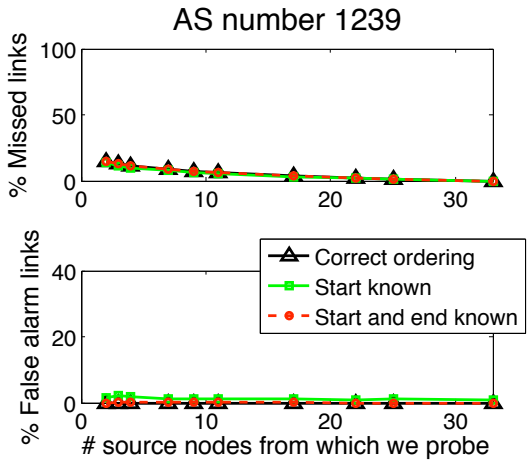


Figure 5: A Rocketfuel topology with average degree ≈ 4. On the left with fewer sources, some links that are simply not measured; the curve for the correct ordering shows the best possible result.

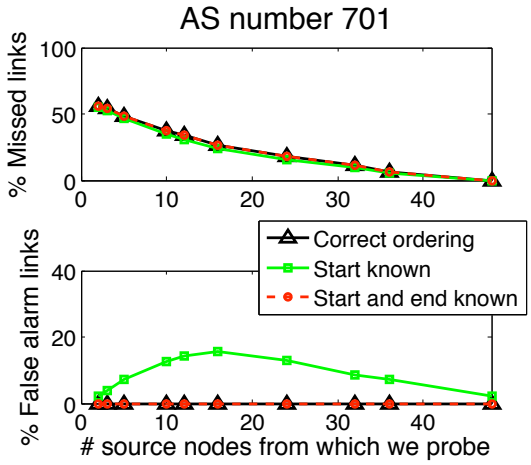


Figure 6: A Rocketfuel topology with average degree ≈ 8.

as we increase the number of source nodes, thus increasing the number of measurements.

We plotted a curve for the oracle case when the correct path orderings are known; this is the best case. Even the best method cannot see the entire network when some links are not probed; thus there are errors even in the best case. The other two curves are using an empirical estimate of P calculated directly from the co-occurrence frequencies: the entry D_{ij} was found by counting the number of times nodes i and j showed up together in a route-set, and then P is the normalized version of D as usual. We used this empirical estimate of P along with knowledge of the source only, and then we used it along with knowledge of both the source and destination. In the former case, we saw mistakes in the ordering of end nodes; this actually results in fewer missed links but many more false alarms. In the latter case, we achieved the best possible reconstruction of the network.

We show results for eight Rocketfuel topologies in Table 2, 3, and 4. When the true P is used, as described in the proofs, as expected we get the best possible performance. Also when the estimate of P is used, the results are perfect except for a few false alarms when only the source is known.

Table 2 shows the results when all nodes are used as source probing nodes, and Table 4 when half the nodes are used as source probing nodes. The results are the same even when only two sources are used to probe as in Table 3: All methods reach the best possible performance[2] except for the method which uses the empirical P and the source only; this is also illustrated in Figures 5 and 6.

Rocketfuel	True P		Src only		Src&Dst	
Topology #	FA	M	FA	M	FA	M
1239	0	0	0.75	0	0	0
7018	0	0	2.80	0	0	0
70181	0	0	2.61	0	0	0
3561	0	0	2.05	0	0	0
1	0	0	0.19	0	0	0
701	0	0	2.24	0	0	0
2914	0	0	0.93	0	0	0
3356	0	0	5.78	0	0	0

Table 2: Missing links (M) and false alarm (FA) links as a percentage of total links, when all source nodes were used for probing. Note the perfect reconstruction attained when either P is known or when the estimate of P is used along with the the source and destination node. A small number of false positives occur with the estimate of P when only the source is known.

	Best		True P		Src		Src&Dst	
Topo #	FA	M	FA	M	FA	M	FA	M
1239	0	14.8	0	14.8	1.5	13.5	0	14.8
7018	0	16.7	0	16.7	2.9	14.8	0	16.7
70181	0	15.6	0	15.6	2.5	14.1	0	15.6
3561	0	94.7	0	94.7	4.5	92.4	0	94.7
1	0	7.7	0	7.7	1.5	7.2	0	7.7
701	0	56.2	0	56.2	2.3	55.2	0	56.2
2914	0	20.9	0	20.9	4.0	18.7	0	20.9
3356	0	87.9	0	87.9	3.2	87.3	0	87.9

Table 3: Link misses and false alarms as a percentage of total links, when only two source nodes were used for probing.

5.2 The Misbehaving Router

Theorem 1 is a necessary and sufficient condition, and thus can be applied to detect malicious or misconfigured routers. If a router announces an ordered route-set, using P we can identify whether that route is a proper route.

However, inherent in the definition of "route-set" is another condition that the route-set or co-occurrence observa-

[2]When we probe from fewer starting nodes, there are missed links even in the best case simply because they are not measured.

	Best		True P		Src		Src&Dst	
Topo #	FA	M	FA	M	FA	M	FA	M
1239	0	3.66	0	3.66	1.1	3.1	0.1	3.68
7018	0	5.6	0	5.6	2.5	4.4	0	5.6
70181	0	5.1	0	5.1	2.5	4.0	0	5.1
3561	0	35.4	0	35.4	7.4	33.7	0	35.4
1	0	2.7	0	2.7	0.6	2.4	0	2.7
701	0	17.7	0	17.7	13.0	15.6	0	17.7
2914	0	5.4	0	5.4	1.6	4.6	0	5.4
3356	0	40.0	0	40.0	5.6	39.0	0	40.0

Table 4: Link misses and false alarms as a percentage of total links, when half the source nodes were used for probing.

Rocketfuel	Average	Percent Wrongly Permitted		
Topology #	degree	3-node	4-node	5-node
1239	3.94	0.75	0.19	0.03
7018	3.89	0.80	0.21	0.03
70181	4.00	1.04	0.30	0.05
3561	10.03	4.66	1.56	0.46
1	3.08	0.64	0.13	0.02
701	7.67	3.82	1.39	0.40
2914	3.74	0.27	0.05	0.01
3356	11.65	10.12	3.87	1.46

Table 5: Percent of all possible 3-, 4-, and 5-node co-occurrences that result in a route being permitted when it is not a true route in the network. A misconfigured router would be unlikely to accidentally report such a route, and the behavior of a malicious router would be severely restricted.

tion is made up of nodes that participated in the routing of a single packet. That is, the nodes in the route-set make up a real route. If the misbehaving router is clever and *concatenates* proper routes in a particular way such that the overall route is not a true route but each individual hop is legitimate, then the lack of the lower Hessenberg property of the permuted P might not identify this as a bad route. Obviously a misconfigured router will not be trying specifically to give such a route, but a malicious router might. How clever must the router be for this to happen?

For the Rocketfuel topologies we used before, we enumerated all possible 3, 4, and 5-node route-sets, including ones that do not consist of nodes in any real path. We calculated the score according to the true matrix P. If the score was non-zero, we can only assume this is a correct route; if it was in fact an incorrect route, the announcement of this route by a malicious router would go undetected.

In the rightmost section of Table 5 we can see that misbehaving routers have very few choices for false routes if they want to go undetected. The percentage of 4-node routes that would go undetected is always below 4%, even for a topology with high node degree. When we get to 5-node routes, the number is only 1.5% for a topology with high node degree. A misconfigured router would therefore be highly unlikely to report a bad route that could not be detected; a malicious router would be severely restricted in the route announcements it could make without being detected.

6. CONCLUSION

Network inference from co-occurrence measurements, using maximum likelihood and a Markov random walk model for routing, results in perfect reconstruction of shortest-path topologies. We have provided a thorough foundation for further study of the connection between a Markov random walk routing model and real routes resulting from nested routing policies. The connection is surprising but fundamental, and will lead to improved network measurement analysis.

7. REFERENCES

[1] Python routing toolkit ('pyrt'). `http://ipmon.sprintlabs.com/pyrt/`.

[2] Ripe NCC: routing information service. `http://www.ripe.net/projects/ris/`.

[3] University of Oregon Route Views Archive Project. `www.routeviews.org`.

[4] `http://www.bgpexpert.com/archive2002q4.php`, 28th August 2002.

[5] AT&T network outage. Nanog mailing list: `http://www.merit.edu/mail.archives/nanog/2002-08/msg00993.html`, 28th August 2002.

[6] Brice Augustin, Timur Friedman, Marie Curie, and Renata Teixeira. Measuring load-balanced paths in the Internet. In *ACM SIGCOMM Internet Measurement Conference*, San Diego, CA, USA, October 2007.

[7] Randy Bush, Olaf Maennel, Matthew Roughan, and Steve Uhlig. Internet optometry: assessing the broken glasses in Internet reachability. In *ACM SIGCOMM Internet Measurement Conference*, pages 242–253, New York, NY, USA, 2009. ACM.

[8] CAIDA. Skitter. `http://www.caida.org/tools/measurement/skitter/`.

[9] Hyunseok Chang, Sugih Jamin, and Walter Willinger. Inferring AS-level Internet Topology From Router-Level Path Traces. In *Proc. of SPIE*, 2001.

[10] Nicholas Duffield, Joseph Horowitz, Francesco Lo Presti, and Don Towsley. Multicast Topology Inference From Measured End-To-End Loss. *IEEE Transactions on Information Theory*, 48(1):26–45, January 2002.

[11] Nicholas Duffield, Francesco Lo Presti, Vern Paxson, and Don Towsley. Network loss tomography using striped unicast probes. *IEEE Transactions on Networking*, 14(4):697–710, 2006.

[12] Anja Feldmann, Albert Greenberg, Carsten Lund, Nick Reingold, and Jennifer Rexford. Netscope: Traffic engineering for IP networks. *IEEE Network Magazine*, pages 11–19, March/April 2000.

[13] Anja Feldmann and Jennifer Rexford. IP network configuration for intradomain traffic engineering.

IEEE Network Magazine, pages 46–57, September/October 2001.

[14] Timothy G. Griffin. The stratified shortest-paths problem. In *COMSNET*, 2009.

[15] G. Iannaccone, C-N. Chuah, R. Mortier, S. Bhattacharyya, and C. Diot. Analysis of link failures over an IP backbone. In *ACM SIGCOMM Internet Measurement Workshop*, Marseilles, France, November 2002.

[16] Van Jacobson. Traceroute. `ftp://ftp.ee.lbl.gov/traceroute.tar.gz`, 1989-04.

[17] Derek Justice and Alfred Hero. Estimation of message source and destination from link intercepts. *IEEE Transactions on Information Forensics and Security*, 1(3):374–385, 2006.

[18] Anukool Lakhina, John Byers, Mark Crovella, and Peng Xie. Sampling biases in IP topology measurements. In *Proc. IEEE Infocom*, April 2003.

[19] Y Liu and H Zhao. A computational approach for ordering signal transduction pathway components from genomics and proteomics data. *BMC Bioinformatics*, 5(158), 2005.

[20] David Maltz, Geoff Xie, Jibin Zhan, Hui Zhang, Gisli Hjalmtysson, and Albert Greenberg. Routing design in operational networks: A look from the inside. In *ACM SIGCOMM*, Portland, OR, USA, 2004.

[21] Z.M. Mao, L. Qiu, J. Wang, and Y. Zhang. On AS-level path inference. In *SIGMETRICS*, 2005.

[22] Michael Rabbat, Mario Figueiredo, and Robert Nowak. Network inference from co-occurrences. *IEEE Transactions on Information Theory*, 54(9):4053–4068, September 2008.

[23] Sylvia Ratnasamy and Steven McCanne. Inference of Multicast Routing Trees and Bottleneck Bandwidths using End-To-End Measurements. In *IEEE Infocom*, 1999.

[24] Marguerite Reardon. Who broke Worldcom's backbone? *Light Reading*, 4th October 2002. `http://www.lightreading.com/document.asp?doc_id=22156`.

[25] Aman Shaikh and Albert Greenberg. Experience in black-box OSPF measurement. In *Proc. ACM SIGCOMM Internet Measurement Workshop*, pages 113–125, 2001.

[26] Aman Shaikh and Albert Greenberg. OSPF Monitoring: Architecture, Design and Deployment Experience. In *Proc. USENIX Symposium on Networked System Design and Implementation (NSDI)*, March 2004.

[27] Craig Smith. Traceroute - whitepaper. `http://www.informatik.uni-trier.de/~smith/networks/tspec.html`.

[28] N Spring, R Mahajan, and D Wetherall. Measuring isp topologies with rocketfuel. In *Proceedings of ACM SIGCOMM*, August 2002.

[29] Xiaojin Zhu, Andrew Goldberg, Michael Rabbat, and Robert Nowak. Learning bigrams from unigrams. In *Proc. Association for Computational Linguistics Conference*, June 2008.