On the Hardness of Topology Inference

H.B. Acharya[1] and M.G. Gouda[2]

[1] The University of Texas at Austin, USA
acharya@cs.utexas.edu
[2] The National Science Foundation, USA
mgouda@nsf.gov

Abstract. Many systems require information about the topology of networks on the Internet, for purposes like management, efficiency, testing of new protocols and so on. However, ISPs usually do not share the actual topology maps with outsiders; thus, in order to obtain the topology of a network on the Internet, a system must reconstruct it from publicly observable data. The standard method employs traceroute to obtain paths between nodes; next, a topology is generated such that the observed paths occur in the graph. However, traceroute has the problem that some routers refuse to reveal their addresses, and appear as anonymous nodes in traces. Previous research on the problem of topology inference with anonymous nodes has demonstrated that it is at best NP-complete. In this paper, we improve upon this result. In our previous research, we showed that in the special case where nodes may be anonymous in some traces but not in all traces (so all node identifiers are known), there exist trace sets that are generable from multiple topologies. This paper extends our theory of network tracing to the general case (with strictly anonymous nodes), and shows that the problem of computing the network that generated a trace set, given the trace set, has no general solution. The weak version of the problem, which allows an algorithm to output a "small" set of networks- any one of which is the correct one- is also not solvable. Any algorithm guaranteed to output the correct topology outputs at least an exponential number of networks. Our results are surprisingly robust: they hold even when the network is known to have exactly two anonymous nodes, and every node as well as every edge in the network is guaranteed to occur in some trace. On the basis of this result, we suggest that exact reconstruction of network topology requires more powerful tools than traceroute.

1 Introduction

Knowledge of the topology of a network is important for many design decisions. For example, the architecture of an overlay network - how it allocates addresses etc.- may be significantly optimized by knowledge of the distribution and connectivity of the nodes on the underlay network that actually carries the traffic. Several important systems, such as P4P [9] and RMTP [7], utilize information about the topology of the underlay network for optimization as well as management. Furthermore, knowledge of network topology is useful in research; for

M.K. Aguilera et al. (Eds.): ICDCN 2011, LNCS 6522, pp. 251–262, 2011.

example, in evaluating the performance of new protocols. Unfortunately, ISPs do not make maps of the true network topology publicly available. Consequently, a considerable amount of research effort has been devoted to the development of systems that reconstruct the topology of networks in the Internet from publicly available data - [10], [6], and [4].

The usual mechanism for generating the topology of a network is by the use of Traceroute [3]. Traceroute is executed on a node, called the source, by specifying the address of a destination node. This execution produces a sequence of identifiers, called a *trace*, corresponding to the route taken by packets traveling from the source to the destination. A trace set T is generated by repeatedly executing Traceroute over a network N, varying the *terminal* nodes, i.e. the source and destination.

If T contains traces that identify every instance when an edge is incident on a node, it is possible to reconstruct the network exactly. However, practical trace sets do not have this property. The most common problems are incomplete coverage, anonymity (where a node can be detected, but will not state its unique identifier, i.e. its address), and aliasing (nodes may have multiple unique identifiers). The situation is further complicated by load balancing, which may cause incorrect traces; tools such as Paris Traceroute [8] attempt to correct this problem.

In this paper, we deal with the problem of inferring the correct network topology in the presence of anonymous nodes. The problem posed by anonymous nodes in a trace is that a given anonymous node may or may not be identical to any other anonymous node. Clearly, a topology in which these nodes are distinct is not identical to one in which they are merged into a single node. Thus, there may be multiple topologies for the computed network. Note that all these candidate topologies can generate the observed trace set; no algorithm can tell, given the trace set as input, which of these topologies is correct.

To solve this problem, Yao et al. [10] have suggested computing the *minimal* topology - the topology of the network with the smallest number of anonymous nodes (subject to some constraints - trace preservation and distance preservation) from which the given trace set is generable. They conclude that the problem of computing a minimal network topology from a given set of traces is NP-complete. Accordingly, most later research in the area, such as [6] and [4], has focused on heuristics for the problem.

We attack this problem from a different direction. In our earlier papers [1] and [2], we introduced a theory of network tracing, i.e. reconstruction of network topology from trace sets. In these papers, we made the problem theoretically tractable by assuming that no node is strictly anonymous. In this theory, a node can be *irregular*, meaning it is anonymous in some traces, but there must exist at least one trace in which it is not anonymous. This simplifying assumption clearly does not hold in practice; in fact, an anonymous node is almost always consistently anonymous, not irregular. (In practical cases, anonymous nodes correspond to routers that do not respond to ping; irregular nodes are routers that drop ping due to excessive load. Clearly, the usual case is for nodes to be

consistently anonymous, rather than irregular.) However, it enabled us to develop a theory for the case when the number of nodes in the network is clearly known (equal to the number of unique identifiers).

In this paper, we develop our theory of network tracing for networks with strictly anonymous nodes. Our initial assumption was that, as irregular nodes are "partially" anonymous, the hardness results in [1] should hold for anonymous nodes. To our surprise, this turned out not to be true; in Theorem 3, we show that networks with one anonymous node are completely specified by their trace sets, while networks with one irregular node are not [1]. Consequently, we constructed a complete new proof for network tracing in the presence of strict anonymity, presented in Section 3. We show that even under the assumption that the minimal topology is correct, the network tracing problem with anonymous nodes is in fact much harder than NP-complete; it is not just intractable, but unsolvable. Even if we weaken the problem and allow an algorithm to return a "small" number of topologies (one of which is correct) the problem remains unsolvable: an algorithm guaranteed to return the correct topology returns a number of topologies which is at least exponential in the total number of nodes (anonymous and non-anonymous).

A very surprising fact is that this result even holds if the number of anonymous nodes is restricted to two. We demonstrate how to construct a trace set which is generable from an exponential number of networks with two anonymous nodes, but not generable from any network with one anonymous node or less. (It is interesting to note that our results are derived under a network model with multiple strong assumptions - stable and symmetric routing, no aliasing, and complete coverage. The reason we choose such friendly conditions for our model is to demonstrate that the problem cannot be made easier using advanced network tracing techniques, such as Paris Traceroute to detect artifact paths, and inference of missing links [5]. We would like to thank Dr Stefan Schmid for this observation.)

We would like to clarify our claim that the problem of identifying the network from which a trace set was generated, given only the trace set, is *unsolvable*. Our proof does not involve a reduction to a known uncomputable problem, such as the halting problem. Instead, we demonstrate that there are many minimal networks - an exponential number of them - that could have generated a given trace set; so, given only the trace set, it is impossible to state with certainty that one particular topology (or even one member of a small set of topologies) represents the network from which the trace set was in fact generated.

The earlier proof of NP-completeness (by a reduction to graph coloring) provided by Yao et al. holds for constructing *a* minimal topology, not *the* minimal topology from which the trace set was generated. It is NP-complete to find a single member of the exponential-sized solution set. Thus, even under the assumption that the true network is minimal in the number of anonymous nodes, trying to reconstruct it is much harder than previously thought.

In the next section, we formally define terms such as network, trace and trace set, so as to be able to develop our mathematical treatment of the problem.

2 Minimal Network Tracing

In this section, we present formal definitions of the terms used in the paper. We also explain our network model and the reasoning underlying our assumptions. Finally, we provide a formal statement of the problem studied.

2.1 Term Definitions

A network N is a connected graph where nodes have unique identifiers. However, a node may or may not be labeled with its unique identifier. If a node is labeled with its unique identifier, it is *non-anonymous*; otherwise, it is *anonymous*. Further, non-anonymous nodes are either *terminal* or *non-terminal*. (These terms are used below.)

A *trace* is a sequence of node identifiers.

A trace t is said to be *generable from* a network N iff the following four conditions are satisfied:

1. t represents a simple path in N.
2. The first and last identifiers in t are the unique identifiers of terminal nodes in N.
3. If a non-anonymous node "a" in N appears in t, then it appears as "a".
4. If an anonymous node "$*$" in N appears in t, then it appears as "$*_i$". i is a unique integer in t, to distinguish anonymous nodes from each other.

A *trace set* T is *generable from* a network N iff the following conditions are satisfied:

1. Every trace in T is generable from N.
2. For every pair of terminal nodes x, y in N, T has at least one trace between x and y.
3. Every edge in N occurs in at least one trace in T.
4. Every node in N occurs in at least one trace in T.
5. T is *consistent*: for every two distinct nodes x and y, exactly the same nodes must occur between x and y in every trace in T where both x and y occur.

We now discuss the reason why we assume the above conditions.

The first condition is obviously necessary. The third and fourth conditions are also clearly necessary, as we are interested in the problem of node anonymity, not incomplete coverage. However, the second and fifth conditions are non-trivial; we explain them as follows.

Conditions like inconsistent or asymmetric routing may or may not be true. Furthermore, it is possible, using tools like source routing and public traceroute pages, to ensure that a trace set contains traces between every possible pair of terminals. As our primary results are negative, we show their robustness by assuming the worst case: we develop our theory assuming the best possible conditions for the inference algorithm, and prove that the results are still valid.

In our earlier work, [1] and [2], we developed our theory using another strong condition: no node was anonymous. For a trace set to be generable from a network, we required that the unique identifier of every node in the network appear in at least one trace. However, on further study we learned that routers in a network appear anonymous because they are configured either to never send ICMP responses, or to use the destination addresses of the traceroute packets instead of their real addresses [10]. Thus, if a node is anonymous in a single trace, it is usually anonymous in all traces in a trace set. This fact reduces our earlier study of network tracing to a theoretical exercise, as clearly its assumptions cannot be satisfied. Accordingly, in this paper, we have discarded this condition, and updated our theory of network tracing to include networks with anonymous nodes.

The introduction of strictly anonymous nodes leads to a complication in our theory: we no longer have all unique identifiers, and cannot be sure of the total number of nodes in the network. Hence we will adopt the same approach as Yao et al. in [10] and attempt to reconstruct a topology with the smallest possible number of anonymous nodes. Accordingly, we adopt a new definition:

A *minimal network N from which trace set T is generable* is a network with the following properties:

1. T is generable from N.
2. T is not generable from any network N' which has fewer nodes than N.

Note that, if there are multiple minimal networks from which a trace set T is generable, then they all have the same number of nodes. Further, as all such networks contain every non-anonymous node seen in T, it follows that all minimal networks from which a trace set T is generable also have the same number of anonymous nodes.

2.2 The Minimal Network Tracing Problem

We can now state a formal definition of the problem studied in this paper.

The *minimal network tracing problem* can be stated as follows: "Design an algorithm that takes as input a trace set T, that is generable from a network, and produces a network N such that T is generable from N and, for any network $N' \neq N$, at least one of the following conditions holds:

1. T is not generable from N'.
2. N' has more anonymous nodes than N."

The *weak minimal network tracing problem* can be stated as follows: "Design an algorithm that takes as input a trace set T, that is generable from a network, and produces a small set $S = \{N_1, .., N_k\}$ of minimal networks such that T is generable from each network in this set and, for any network $N' \notin S$, at least one of the following conditions holds:

1. T is not generable from N'.
2. N' has more anonymous nodes than any member of S."

The minimal network tracing problem is clearly a special case of the weak minimal network tracing problem, where we consider only singleton sets to be small.

In Section 3, we show that the weak minimal network tracing problem is unsolvable in the presence of anonymous nodes, even if we consider only sets of exponential size to be "not small"; of course, this means that the minimal network tracing problem is also unsolvable.

3 The Hardness of Minimal Network Tracing

In this section, we begin by constructing a very simple trace set with only one trace,

$$T_{0,0} = \{(a, *_1, b_1)\}$$

which, of course, corresponds to the network in Figure 1.

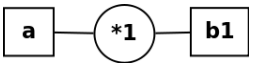


Fig. 1. Minimal topology for $T_{0,0}$

We now define two operations to grow this network, $Op1$ and $Op2$. In $Op1$, we introduce a new non-anonymous node and a new anonymous node; the non-anonymous nodes introduced by $Op1$ are b-nodes. In $Op2$, we introduce a non-anonymous node, but may or may not introduce an anonymous node; if we only consider minimal networks, then in $Op2$ we only introduce non-anonymous nodes.

To execute $Op1$, we introduce a new b-node (say b_i) which is connected to a through a new anonymous node $*_i$. We will now explain how we ensure that $*_i$ is a new anonymous node.

Note that our assumption of consistent routing ensures that there are no loops in traces. Thus, we can ensure that $*_i$ is a "new" anonymous node (and not an "old", i.e. previously-seen anonymous node) by showing that it occurs on a trace with every old anonymous node. To achieve this, we add traces from b_i to each pre-existing b-node b_j. These traces are of the form $(b_i, *_{ii}, a, *_{jj}, b_j)$. We then use consistent routing to show that $*_i = *_{ii}$ and $*_j = *_{jj}$, and (as we intended) $*_i \neq *_j$.

We denote the trace set produced by applying $Op1$ k times to $T_{0,0}$ by $T_{k,0}$.

For example, after one application of $Op1$ to $T_{0,0}$, we obtain trace set $T_{1,0}$:

$$T_1 = \{(a, *_1, b_1), (a, *_2, b_2), (b_1, *_3, a, *_4, b_2)\}$$

As we assume consistent routing, $*_1 = *_3$ and $*_2 = *_4$. Furthermore, as $*_3$ and $*_4$ occur in the same trace, $*_1 \neq *_2$.

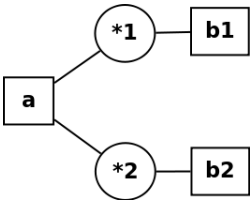


Fig. 2. Minimal topology for $T_{1,0}$

There is exactly one possible network from which this trace set is generable; we present it in Figure 2.

We now define operation "$Op2$". In $Op2$, we introduce a new non-anonymous node (c_i). We add traces such that c_i is connected to a through an anonymous node, and is directly connected to all b and c nodes. We denote the trace set produced by applying $Op2$ l times to $T_{k,0}$ by $T_{k,l}$.

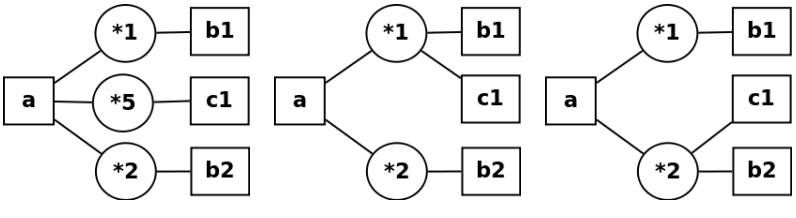
For example, one application of $Op2$ to the trace set $T_{1,0}$ produces trace set $T_{1,1}$ given below.

$$T_{1,1} = \{(a, *_1, b_1), (a, *_2, b_2), (b_1, *_3, a, *_4, b_2), \\ (a, *_5, c_1), (b_1, c_1), (b_2, c_1)\}$$

From Figure 3 we see that three topologies are possible:

(a) $*_5$ is a new node. $*_1 \neq *_5$ and $*_2 \neq *_5$.
(b) $*_1 = *_5$.
(c) $*_2 = *_5$.

But network $N_{1,1.1}$ is not minimal; it has one more anonymous node than the networks $N_{1,1.2}$ and $N_{1,1.3}$. Hence, in future we discard such topologies and only consider the cases where the anonymous nodes introduced by $Op2$ are "old" (previously-seen) anonymous nodes.



(a) Network $N_{1,1.1}$ (b) Network $N_{1,1.2}$ (c) Network $N_{1,1.3}$

Fig. 3. Topologies for $T_{1,1}$

We are now in a position to prove the following theorem:

Theorem 1. *For every pair of natural numbers (k, l), there exists a trace set $T_{k,l}$ that is generable from $(k+1)^l$ minimal networks, and the number of nodes in every such network is $2k + l + 3$.*

Proof. Consider the following construction.

Starting with $T_{0,0}$, apply $Op1$ k times successively. This constructs the trace set $T_{k,0}$, which has $k + 1$ distinct anonymous nodes.

Finally, apply $Op2$ l times in succession to get $T_{k,l}$.

Now, we show that $Op2$ indeed has the properties claimed. Note that every time $Op2$ is applied, it introduces an anonymous identifier. This identifier can correspond to a new node or to a previously-seen anonymous node. As we are considering only minimal networks, we know that this is a previously-seen anonymous node.

There are $k+1$ distinct anonymous nodes, and the newly-introduced identifier can correspond to any one of these. There is no information in the trace set to decide which one to choose. Furthermore, each of these nodes is distinct - it is connected to a different (non-anonymous) b-node. In other words, each choice produces a distinct topology from which the constructed trace set is generable.

Hence the number of minimal networks from which the trace set $T_{k,l}$ is generable, is $(k + 1)^l$.

Further, there are 3 nodes to begin with. Every execution of $Op1$ adds two new nodes (b-node and the new $*$-node), and every execution of $Op2$ adds one new node (the c-node). As the total number of nodes in a minimal network is n, we also have $n = 3 + 2k + l$.

We can see that n grows linearly with k and l. The number of candidate networks from which $T_{k,l}$ is generable, grows as $(k + 1)^l$. So, for example if we take $k = l = \frac{n-3}{3}$, the number of candidate networks is $(\frac{n}{3})^{(\frac{n}{3}-1)}$, which is obviously exponential.

In fact, this expression is so strongly exponential that it remains exponential even in the special case where we restrict the number of anonymous nodes to exactly two. Note that, if we execute $Op1$ exactly once and $Op2$ l times, then by the formula above the number of minimal networks is $2^l = 2^{n-5}$, which is $O(2^n)$ - exponential. We have proved the following theorem:

Theorem 2. *For any $n \geq 6$, there exists a trace set T such that:*

(a) n is the number of nodes in a minimal network from which T is generable.
(b) Every such minimal network has exactly two anonymous nodes.
(c) The number of such minimal networks is $O(2^n)$.

As an example, Figure 4 shows all $2^3 = 8$ possible networks from which the trace set $T_{1,3}$ is generable.

We are now in a position to state our result about the minimal network tracing problem.

Theorem 3. *Both the minimal network tracing problem and the weak minimal network tracing problem are unsolvable in general, but solvable for the case where the minimal network N, from which trace set T is generable, has exactly one anonymous node.*

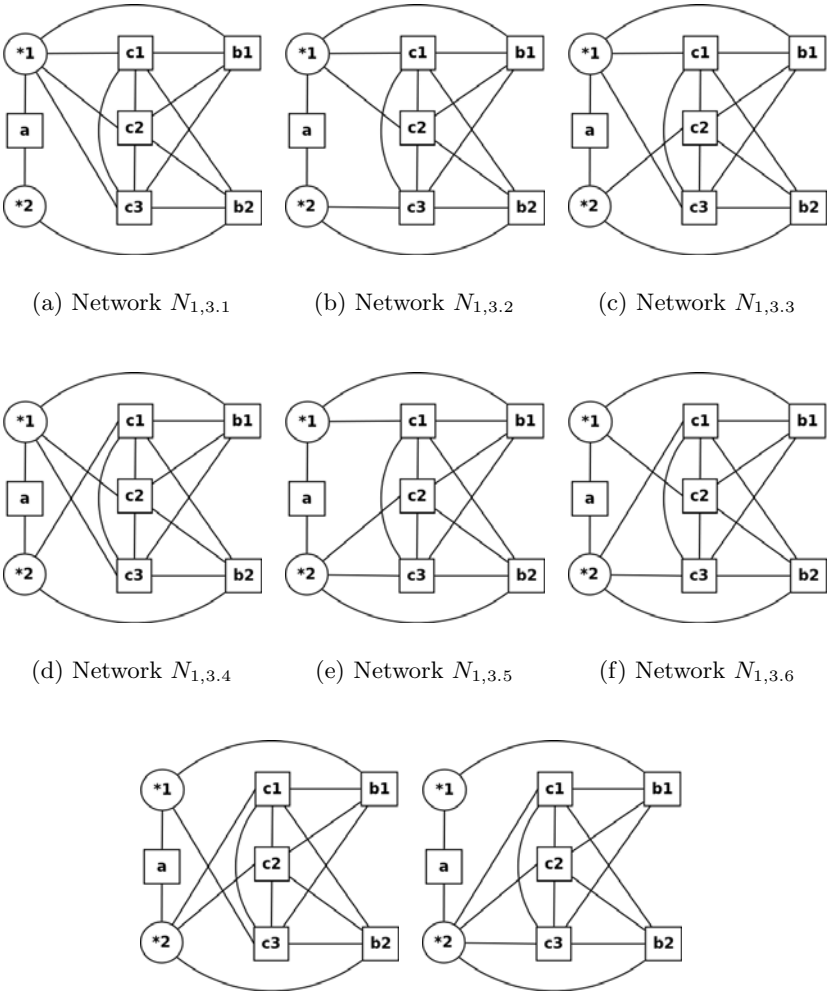
Proof. Consider any algorithm that can take a trace set and return the correct network. If the algorithm is given as input one of the trace sets shown in Theorems 1 and 2, it must return an exponentially large number of networks in the worst case. (If it does not return all networks from which the trace set is generable, it may fail to return the topology of the actual network from which the trace set was generated.) In other words, no algorithm that always returns a "small" number of networks can be guaranteed to have computed the correct network from the trace set; the weak minimal network tracing problem is unsolvable in general. As the minimal network tracing problem is a stricter version of this problem, it is also unsolvable.

The case where the minimal network has only one anonymous node is special. If there is only one anonymous node, there is no need to distinguish between anonymous nodes. We assign it some identifier (say x) that is not the unique identifier of any non-anonymous node, and replace all instances of "$*$" by this identifier. Now the problem reduces to finding a network from a trace set with no anonymous (or irregular) nodes, which is of course solvable [1]. As the minimal network tracing problem is solvable in this case, the weak minimal network tracing problem (which is easier) is solvable also.

4 Unsolvable, or NP-Complete?

In Section 3, we demonstrated the hardness of the minimal network tracing problem in the presence of anonymous nodes, and concluded that both the strict and the weak versions of the problem are unsolvable in general. It is natural to ask how we can claim a problem to be *unsolvable*, unless we reduce it to the halting problem or some other such uncomputable problem. Also, it seems on first observation that our findings conflict with the earlier results of Yao et al., who had found the problem of minimal topology inference to be NP-complete; an NP-complete problem lies in the intersection of NP-hard and NP, so it lies in NP and is definitely not unsolvable! In this section, we will answer these questions and resolve this apparent conflict.

The problem we study is whether it is possible to identify the true network from which a given trace set T was generated in practice - in other words, to find a single network N such that T is generable from N and only from N. As there is not enough information in T to uniquely identify N (because T is generable from many minimal networks), the minimal network tracing problem is not solvable. In fact, even the weak minimal network tracing problem is not solvable, as T only provides enough information for us to identify that N is one member of an exponential-sized set (which is clearly not a small set). Thus, our statement that the problem is not solvable does not depend on proving uncomputability, but on



(a) Network $N_{1,3.1}$ (b) Network $N_{1,3.2}$ (c) Network $N_{1,3.3}$

(d) Network $N_{1,3.4}$ (e) Network $N_{1,3.5}$ (f) Network $N_{1,3.6}$

(g) Network $N_{1,3.7}$ (h) Network $N_{1,3.8}$

Fig. 4. Minimal Topologies for $T_{1,3}$ (with two anonymous nodes)

the fact that no algorithm can identify the correct solution out of a large space of solutions, all of which are equally good.

We now consider how our work relates to the proof of Yao [10]. The solution to our apparent conflict is that Yao et al. claim NP completeness for the decision problem TOP-INF-DEC, which asks "Does there exist a network, from which trace set T is generable, and which has at most k anonymous nodes?" This decision problem is equivalent to the problem of demonstrating *any one* network from which T is generable, with k or less anonymous nodes.

Yao et al. implicitly assume that the space of networks, from which a trace set T is generable, is a search space; identifying the smallest network in this space will yield the true network from which T was generated in practice. This is simply not true - the number of minimal networks from which T is generable is at least exponentially large, and as these are all minimal networks we cannot search for an optimum among them (they are all equally good solutions; in fact, they satisfy a stronger equivalence condition than having the same number of nodes - our construction produces networks with the same number of nodes and the same number of edges). Finding one minimal network N from which T is generable does not guarantee that N is actually the network from which T was generated! We say nothing about the difficulty of finding a random minimal network from which a trace set is generable (without regard to whether it is actually the network that generated the trace set). Hence, there is no conflict between our results and the results in [10].

5 Conclusion

In our previous work, we derived a theory of network tracing under the assumption that nodes were not consistently anonymous. As we later learned that this assumption is impossible to satisfy, we updated our theory to include networks with strictly anonymous nodes, which we present in this paper.

As the introduction of irregularity - a limited form of anonymity - caused the problem to become hard in our previous study, we had expected that it would be even harder when we introduced strict anonymity. To our great surprise, we found a counterexample. Networks with a single anonymous node are completely specified by their trace sets (Theorem 3), while networks with a single irregular node are not (Figure 1 of [1]). We feel that this example is very interesting, as it disproves the intuition that introducing anonymous nodes should cause more trouble to a network tracing algorithm than introducing irregular (partly anonymous) nodes.

In the general case, however, we prove in this paper that both the strict version and the weak versions of the minimal network tracing problem are unsolvable: no algorithm can do better than reporting that the required network is a member of an exponentially large set of networks. This result holds even when the number of anonymous nodes is restricted to two. The question of identifying the particular classes of networks, with the property that any such network can be uniquely identified from any trace set generable from it (even if the network contains anonymous nodes), is an open problem we will attack in future research.

References

1. Acharya, H.B., Gouda, M.G.: A theory of network tracing. In: 11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (November 2009)

2. Acharya, H.B., Gouda, M.G.: The weak network tracing problem. In: International Conference on Distributed Computing and Networking (January 2010)
3. Cheswick, B., Burch, H., Branigan, S.: Mapping and visualizing the internet. In: Proceedings of the USENIX Annual Technical Conference, pp. 1–12. USENIX Association, Berkeley (2000)
4. Gunes, M., Sarac, K.: Resolving anonymous routers in internet topology measurement studies. In: INFOCOM 2008: The 27th Conference on Computer Communications, pp. 1076–1084. IEEE, Los Alamitos (April 2008)
5. Gunes, M.H., Sarac, K.: Inferring subnets in router-level topology collection studies. In: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement, pp. 203–208. ACM, New York (2007)
6. Jin, X., Yiu, W.-P.K., Chan, S.-H.G., Wang, Y.: Network topology inference based on end-to-end measurements. IEEE Journal on Selected Areas in Communications 24(12), 2182–2195 (2006)
7. Paul, S., Sabnani, K.K., Lin, J.C., Bhattacharyya, S.: Reliable multicast transport protocol, rmtp (1996)
8. Viger, F., Augustin, B., Cuvellier, X., Magnien, C., Latapy, M., Friedman, T., Teixeira, R.: Detection, understanding, and prevention of traceroute measurement artifacts. Computer Networks 52(5), 998–1018 (2008)
9. Xie, H., Yang, Y.R., Krishnamurthy, A., Liu, Y.G., Silberschatz, A.: P4p: provider portal for applications. SIGCOMM Computer Communications Review 38(4), 351–362 (2008)
10. Yao, B., Viswanathan, R., Chang, F., Waddington, D.: Topology inference in the presence of anonymous routers. In: Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, March - April 3, vol. 1, pp. 353–363. IEEE, Los Alamitos (2003)