Network Link Tomography and Compressive Sensing

Rhys Bowden
School of Mathematical Sciences
University of Adelaide
SA, 5005, Australia
rhys.bowden@adelaide.edu.au

Matthew Roughan
School of Mathematical Sciences
University of Adelaide
SA, 5005, Australia
matthew.roughan@adelaide.edu.au

Nigel Bean
School of Mathematical Sciences
University of Adelaide
SA, 5005, Australia
nigel.bean@adelaide.edu.au

1. INTRODUCTION AND BACKGROUND

Accurate and timely performance data are of vital importance for network administration. However, modern networks are so large and transmit such enormous quantities of data that a single backbone link could fill a terabyte drive in about 3 minutes. Taking and processing all the desirable performance measurements can be wildly impractical. Aside from matters of scale there may be other difficulties, such as unreliable measurement that mean network administrators cannot make all the performance measurements they desire. Consequently, it is necessary to make the most of the measurements that *are* available. *Network Tomography* does just that, by inferring underlying performance statistics from the available measurements.

This paper considers the problem of link loss tomography: inference of link parameters from a series of end-to-end probes through a network. We specifically estimate average link loss rates. Typical problems in this setting are highly underconstrained, and so the measurements often admit infinitely many solutions. Some method is needed to select the correct solution from this possible set, and in this paper we shall use sparsity.

Network tomography is a well developed field [1, 4, 7]. However, the vast majority of performance tomography has concentrated on trees. In that setting, it is possible to develop fast, recursive algorithms [2, 4], and to employ side information such as sparsity relatively easily [3].

However, many networks are not trees. Some work has looked at combining measurements from multiple tree-like views of the network [6], however, the approach meets immediate difficulties. Intuitively we can see that it would be hard to use sparsity in the same way because there is no longer a "top" of the tree towards which we can push "bad" links.

In this paper we attack the problem on a general network. We exploit sparsity, but without reducing the problem to a binary problem. We test the idea of applying the field of Compressive Sensing to this link tomography problem. Compressive Sensing exploits the fact that many large data-sets are comprised of only a few significant elements. In practice, this means that either the data itself, or some simple transform of the data, is *sparse* in the sense that only a few of the values are non-zero. Compressive Sensing is a rapidly growing area of research, and there are many powerful results. However, the underlying assumption in most Compressive Sensing is that the experimenter controls the *measurement matrix*, but here the measurement matrix is called a *routing matrix* and it is not chosen to suit the inference problem (its choice is mandated by the design and optimisation of the network). What's more, routing matrices don't satisfy key properties such as RIP that would allow us to apply the theory of Compressive Sensing. The central question of this paper is "Can we still use the concepts and methods of Compressive Sensing despite the deficiencies of the routing matrices as measurement matrices?"

We show here that we can apply Compressive Sensing, with a reasonable degree of accuracy. More importantly, the structural features of typical routing matrices that make them unsuitable for standard Compressive Sensing theorems (highly correlated rows, variable lengths of paths) can be exploited. We develop here a new algorithm — Coherent Tomographic Deduction (CTD) — for solving the tomography problem and show that it is orders of magnitude faster than a standard Compressive Sensing technique, ℓ^1-norm minimisation, with the same level of accuracy. Apart from being much faster, our algorithm has one other very significant advantage. It knows where it is definitely right, and where it could be wrong.

1.1 Notation and assumptions

We consider the problem of inferring the loss probabilities on links across a network from path measurements. Let $l_i =$ link loss probability for $i = 1, \dots, n$, and $p_j =$ path loss probability for $j = 1, \dots, m$. When losses on different links are independent the two are related by

$$\boldsymbol{\rho} = A\boldsymbol{\tau}. \tag{1}$$

where $\tau_i = -\log(1 - l_i), \quad i = 1, \dots n$; and $\rho_j = -\log(1 - p_j); j = 1, \dots, m$. and A is the $m \times n$ routing matrix defined by $A_{j,i} = 1$ if link i is on path j and 0 otherwise. We assume that the routing matrix A is known, and we wish to find the sparsest $\boldsymbol{\tau}$ such that $\boldsymbol{\rho} = A\boldsymbol{\tau}$, i.e.,

$$\min_{\tau} ||\boldsymbol{\tau}||_0 \text{ subject to } \boldsymbol{\rho} = \Phi\boldsymbol{\tau}. \tag{2}$$

where $||\mathbf{x}||_0$ is the number of non-zero elements of $\mathbf{x}$.

2. ALGORITHM (CTD)

2.1 Part A

Iterated Bounding Step: If we examine the origins of $\boldsymbol{\tau}$ we can see that $\boldsymbol{\tau} \geq 0$. We can use this lower bound as a starting place to derive tighter bounds for each τ_i.

SIGMETRICS'11, June 7–11, 2011, San Jose, California, USA.
ACM 978-1-4503-0262-3/11/06.

Let $\mathbf{l}^t$ be the length n vector of lower bounds attained after t steps. Similarly, let $\mathbf{u}^t$ be the upper bounds attained after t steps. Start with $\mathbf{l}^0 = \mathbf{0}$. Then define the rest of the sequence by $\mathbf{u}_j^{t+1} = \min_{i:A_{i,j}=1} \left(\rho_i - \sum_{k:A_{i,k}=1, i \neq k} \mathbf{l}_k^t \right)$; $\mathbf{l}_j^{t+1} = \max_{i:A_{i,j}=1} \left(\rho_i - \sum_{k:A_{i,k}=1, i \neq k} \mathbf{u}_k^{t+1} \right)$

These two steps are using the previous bounds to find the new bounds. Halt this process when $||\mathbf{l}^t - \mathbf{l}^{t-1}|| < \varepsilon$ for some sufficiently small $\varepsilon > 0$. In the final step, each link i that has a lower bound $\mathbf{l}_i$ the same as its upper bound $\mathbf{u}_i$ we consider solved, and eliminate it from the remaining measurement equations.

Length 2 paths step: Thanks to the iterated bounding step, at this stage all paths with unsolved links have at least 2 unsolved links, and have some measured loss. We consider paths with only 2 links. Now one of two things is true: either we can find a unique solution for the entirety of this subproblem, or we can find a set of solutions with only one degree of freedom (we omit the proof here). The length 2 paths step and the iterated bounding step can be repeated alternately until no more progress is made.

2.2 Part B

While our algorithm CTD can derive solutions for part of $\boldsymbol{\tau}$ with absolute confidence, there may be some components about which it is unsure. To estimate the remainder of the solution after running CTD, we use a standard Compressive Sensing technique, ℓ^1-minimisation.

3. SIMULATION

In this section we evaluate the performance of the CTD algorithm on simulated topologies. To generate the topologies, we use ideas from Li *et al.* [5] who highlight that the Internet is a designed network, and is therefore likely to be optimised subject to the physical and social constraints placed upon it.

We compare the performance of (i) ℓ^1-minimisation by itself with (ii) CTD on heuristically optimised networks with 80 nodes: 10 nodes chosen as sources and a variable number of destinations. As far as estimation goes, the loss estimates provided by CTD and raw ℓ^1-minimisation are accurate for an identical, high proportion of the links (Figure 3). However, CTD Part A also labels some of the loss values as being accurate, and some as having uncertainty. The average proportion of links determined with certainty is shown as the solid line in Figure 3.

CTD doesn't just provide more information than ℓ^1- minimisation. As an additional advantage, CTD as a whole is much faster than ℓ^1-minimisation in finding the estimate for $\boldsymbol{\tau}$. With 40 destination nodes, there is almost an order of magnitude difference in run times; and with more destination nodes the difference becomes even greater (plot omitted due to space constraints).

4. CONCLUSION

We present here a justification of why Compressive Sensing techniques, while appearing suited, are not *guaranteed* to work when applied to the problems of Link Tomography; routing matrices poorly satisfy the typical conditions required for Compressive Sensing. We present an efficient algorithm: the first half of this algorithm determines the solution on some subset of links for which we have sufficient information; the second half then applies a Compressive Sensing algorithm to find an approximation to the sparsest solution on the remainder of the links. We test this algorithm on simulated topologies and find that it is both faster and gives more information than applying Compressive Sensing techniques directly. Future work will involve extending the method to include the treatment of noise and measurement errors.

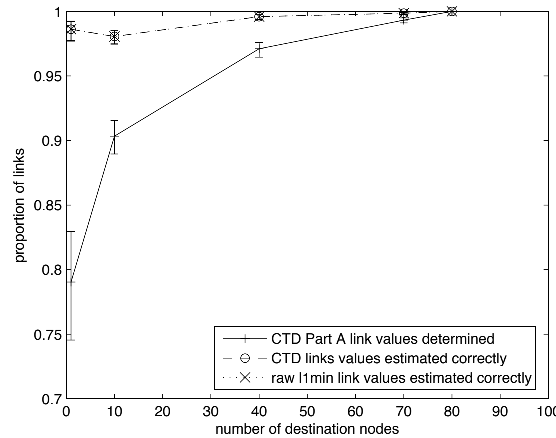

Figure 1: Average proportion of the links determined (in the case of CTD Part A), or estimated correctly (in the other cases) plotted against number of destinations. Note that the lines for CTD estimation and raw ℓ^1-min estimation are identical.

5. REFERENCES

[1] M. Coates, R. Hero, A. Nowak, and B. Yu. Internet tomography. *IEEE Signal Processing Magazine*, 19(3):47–65, May 2002.

[2] M. Coates and D. Nowak. Network tomography for internal delay estimation. In *IEEE ICASSP '01*, May 2001.

[3] N. Duffield. Network tomography of binary network performance characteristics. *IEEE Transactions on Information Theory*, 52(12):5373–5388, December 2006.

[4] N. Duffield, F. LoPresti, V. Paxson, and D. Towsley. Network loss tomography using striped unicast probes. *IEEE/ACM Trans. Networking*, 14(46):697–710, Aug. 2000.

[5] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the Internet's router-level topology. In *SIGCOMM '04*, pages 3–14, New York, NY, USA, 2004. ACM.

[6] M. Rabbat, R. Nowak, and M. Coates. Multiple source, multiple destination network tomography. In *IEEE INFOCOM '04*, 2004.

[7] Y. Shavitt, X. Sun, A. Wool, and B. Yener. Computing the unmeasured: An algebraic approach to internet mapping. In *IEEE INFOCOM '01*, 2001.