A Two-step Statistical Approach for Inferring Network Traffic Demands*

A. Medina[1], K. Salamatian[2], N. Taft[3], I. Matta[4], C. Diot[5],
[1] ICIR, Berkeley, CA, USA
[2] University Paris 6, Paris, France
[3] Intel, Berkeley, CA, USA
[4] Boston University, Computer Science Dept., Boston, MA, USA
[5] Intel, Cambridge, UK

March 1, 2004
Technical Report BUCS-2004-011
Revises BUCS-TR-2003-003

ABSTRACT

Accurate knowledge of traffic demands in a communication network enables or enhances a variety of traffic engineering and network management tasks of paramount importance for operational networks. Directly measuring a complete set of these demands is prohibitively expensive because of the huge amounts of data that must be collected and the performance impact that such measurements would impose on the regular behavior of the network. As a consequence, we must rely on statistical techniques to produce estimates of actual traffic demands from partial information. The performance of such techniques is however limited due to their reliance on limited information and the high amount of computations they incur, which limits their convergence behavior. In this paper we study a two-step approach for inferring network traffic demands. First we elaborate and evaluate a modeling approach for generating good starting points to be fed to iterative statistical inference techniques. We call these starting points *informed priors* since they are obtained using actual network information such as packet traces and SNMP link counts. Second we provide a very fast variant of the EM algorithm which extends its computation range, increasing its accuracy and decreasing its dependence on the quality of the starting point. Finally, we evaluate and compare alternative mechanisms for generating starting points and the convergence characteristics of our EM algorithm against a recently proposed Weighted Least Squares approach.

1. INTRODUCTION

An IP traffic matrix (TM) is a succinct representation of a given metric for the traffic exchanged between end points in an IP network. Metrics such as average volume, delay, and loss rates, and end points such as Points-of-Presence (PoP), routers or even address prefixes may be considered to define different types of traffic matrices [8]. For example, a traffic volume PoP-to-PoP traffic matrix X represents the amount of traffic exchanged between two PoPs, where X_{ij} captures the volume of traffic traveling from ingress PoP i to egress PoP j.

Access to accurate TMs is of paramount importance because they enable or enhance many traffic engineering and network management tasks needed to operate the network at optimal levels of performance and efficiency. Capacity planning, routing protocol (e.g OSPF or ISIS) configuration, definition of load balancing policies and fail-over strategies are among the tasks that would significantly benefit from having access to accurate TMs.

In this paper we address the problem of obtaining PoP-to-PoP traffic volume traffic matrices in the context of a Tier-1 ISP. A PoP is an aggregate of core and access routers usually co-located in a building. A Tier-1 ISP has a set of interconnected PoPs distributed across a geographical area forming the PoP-level network topology of the ISP. Directly measuring TMs is very difficult because of the highly complex structure of PoPs, the number of links that need to be measured and the daunting amounts of data that need to be collected. Sampling approaches [3, 1] have been proposed recently and appear to be promising but are not readily available yet to be useful to operational ISPs. The challenge is then to obtain TMs using only limited measurements and information from the network.

Previous work on obtaining traffic matrices has relied on statistical inference techniques that use partial information to estimate the TM. The term *Network Tomography* [12] was coined for this problem when the partial data come from repeated measurements of the traffic flowing along directed links in the network. Such data are usually obtained using the Simple Network Management Protocol (SNMP), which allows measuring the total amount of incoming and outgoing bytes on a link. Inference approaches use these *link* statistics to infer the characteristics of *end-to-end flows,* which are defined within a single domain and are usually referred to as origin-destination (OD) pairs. In a PoP-to-PoP topology, the origin and destination nodes are PoPs. In addition to inference methods, it is also possible to formulate the traffic matrix estimation problem as a constrained optimization problem and use techniques such as Linear Programming [5].

In [9] a comparative study of existing TM inference techniques was conducted. It was observed that the evaluated statistical techniques [12, 11, 2] outperformed an optimization technique based on linear programming. However, statistical techniques are still significantly restricted in the accuracy of their estimations because they rely on scarce actual network information and require very intensive computations. These techniques usually start from an initial point in the solution space and iteratively attempt to converge to an optimal solution. The statistical techniques evaluated in [9] showed high sensitivity to the quality of the provided starting point

---

*The work of Matta was supported in part by NSF grants ANI-0095988, EIA-0202067 and ITR ANI-0205294. Most of this work was done while the authors were at Sprint Labs.

since degraded estimations were obtained as the quality of the starting points decreased. Such sensitivities are due to the lack of actual network information used by the techniques, the amounts of computations required, and the fact that many statistical techniques are prone to converge to local minima.

In this paper we investigate a two-step statistical approach aimed at overcoming the limitations described in [9] and enabling an efficient and accurate estimation of network traffic demands for operational networks. The first step in this approach is to use a mechanism to generate good starting points. In the second step, the generated starting point is fed into a statistical inference technique for estimating network traffic demands. The division of the TM estimation process into two steps offers great flexibility for combining and evaluating different strategies that could be applied to solve the inference problem. Different mechanisms can be used in the first step, to evaluate the performance of a given inference technique in the second step. Alternatively, a given mechanism for generating starting points can be used in the first step, to evaluate the performance of alternative inference techniques in the second step. In this paper we combine three alternatives for the first step with two alternatives for the second step and evaluate the performance of the possible combinations.

In the first step we address the sensitivity of statistical techniques to the quality of starting points. Specifically, we investigate alternative modeling approaches to obtain *reasonable* starting points for inference techniques. We call these starting points *informed priors* because they are obtained from models that incorporate substantial network information. Specifically we focus on commonly applied *distributional* models (e.g. Gaussian, Bimodal), a simple gravity model as introduced by Zang et al. [13], and a choice model as introduced by Medina et al. [9]. Although choice models were proposed in this context in [9], they were not evaluated then, and we leverage on that work by defining a *choice-model framework* that provides a family of models for generating starting points; we developed a choice-model calibration strategy, and evaluate the performance of instances of choice models for generating starting points.

In the second step we address the limitation imposed by the intensive computations incurred by statistical techniques by introducing a fast variant of the Expectation Maximization (EM) algorithm for the network traffic inference problem. The improvements made to the EM algorithm are aimed at reducing its computation requirements and expanding the iterative horizon in search of global optima.

We found that some of the approaches for modeling starting points behave similarly and produce results within the same error range. We observed that providing *informed starting points* to statistical techniques yields improved estimation results. We compared our EM algorithm to a recently proposed alternative approach that uses quadratic programming, or more specifically, a weighted least squares (WLSE) algorithm [13], and observed the largest performance gains when providing a choice-model starting point with the modified EM algorithm.

The study presented in this paper is the first, to the best of our knowledge, to compare with the same test scenarios different mechanisms for generating starting points combined alternatively with the EM algorithm or the WLSE inference techniques.

The rest of the paper is organized as follows. In Section 2 we define the TM inference the problem formally. In Section 3 we review the main statistical techniques that have been proposed for inferring network traffic demands, and discuss the steps of our inference approach, namely, the modeling and generation of informed priors and the fast variant of the EM algorithm. Section 4 describes the collection of packet traces and SNMP data we use in this study. In Section 5 we discuss the methodology we followed for the performance evaluation of the studied techniques. In Section 6, we evaluate alternative starting points models. In Section 7, we present and discuss the results of the performance evaluation. Finally, Section 8 concludes the paper.

2. PROBLEM STATEMENT

The problem of inferring network traffic demands can be formulated as follows. Let m be the number of origin-destination (OD) pairs. In a network with n nodes, $m = n \times (n-1)$. Rather than representing the amount of data transmitted from node i to node j as X_{ij}, it is more convenient to represent the OD pairs in vector form. Thus, let X_j be the amount of data transmitted by OD pair j [1]. Let $Y = (y_1, ...y_L)$ be the vector of link counts where y_i gives the link count for link i, and L denotes the total number of links in the network. The vectors X and Y are related through an L by m routing matrix R. R is a $\{0,1\}$ matrix where $r_{ij} = 1$ if link i belongs to the path associated with OD pair j, and $r_{ij} = 0$ otherwise. The OD flows are thus related to the link counts according to the following linear relation:

$$Y = RX \tag{1}$$

In IP networks, the routing matrix R can be obtained by gathering topological information, as well as OSPF or IS-IS link weights. Using this information we can compute the shortest-paths between all OD pairs.

For simplicity, we assume the existence of a fixed single-path routing, that is, there is a single shortest path selected by all traffic flowing between any pair of end nodes in the network.[2] Link counts in Y are obtained from SNMP data. The problem is thus to compute X, that is, to find a set of OD flows that would reproduce the observed link counts as closely as possible. Notice that this formulation assumes that the components of Y come from a single measurement interval. A series of consecutive measurements of SNMP link counts, Y_i^k, can be considered, each one denoting the average load on link i in measurement period k. With such repeated measurements, the demands are as well modified to X_j^k, denoting the traffic demand for OD pair j in measurement interval k. The OD traffic demands and link counts are still related through R, as $Y^k = RX^k$.

The problem described by Equation (1) is highly under-determined because in almost any network, the number of OD pairs is much higher than the number of links in the network, that is, $L \ll m$. This means that there are an infinite number of feasible solutions for X.

There is additional information that may be incorporated into the problem statement. Specifically, the total amount of bytes leaving a node i corresponds to the sum of the SNMP link counts for all outgoing links from node i. Similarly, the total amount of bytes incoming into a node j corresponds to the sum of the SNMP counts over all links coming into node j. The amount of traffic traveling from i to j can be computed from the total amount of traffic exiting node i (denoted by O_i) multiplied by the fraction of this traffic headed toward node j. Let α_{ij} denote the fraction of the total traffic from node i traveling toward node j. With this notation, we can write X_{ij} as

---

[1] In this subsection we use X defined this way as a vector for mathematical convenience. In the rest of the paper we let X be indexed by $_{ij}$ to identify the origin and destination indices.

[2] It is straightforward to relax this assumption to deal with other routing schemes, e.g. multi-path (ECMP) routing.

$$X_{ij} = O_i \alpha_{ij} \tag{2}$$

The set of proportions, $\alpha_{ij}, \forall j$ corresponds to what is often called the *fanout intensities* of node i. An alternative angle to look at the traffic estimation problem is to focus on the estimation of the *fanout intensities* of nodes in the network [9]. In other words, the problem now becomes that of estimating the proportionality factors, α_{ij}.

It is important to notice that if the fanout intensities can be accurately obtained, then the traffic matrix itself would consequently be accurately estimated from Equations (2), and there would not be any need for further inference or estimation procedures. For the same reasons described in the introduction, directly measuring the fanout intensities is prohibitively expensive. Therefore, the likely scenario would be one in which the fanout intensities are estimated with certain errors. Nevertheless, these sub-optimal estimated fanout intensities would be very useful to provide good starting points for the estimation procedures of statistical techniques.

3. TWO-STEP INFERENCE APPROACH

Statistical approaches for estimating network traffic demands have the general structure depicted in Figure 1. There are three main inputs. First, each statistical approach makes an assumption about the elements (entries) of the TM. Such an assumption is not actually an input but the foundation of the estimation procedure used later is fundamentally influenced by such assumption. Second, statistical methods usually require some starting point (prior) information, aimed at conveying some *clues* about the traffic matrix being estimated. Such a starting point may correspond to an outdated version of the TM or be the output of some other mechanism aimed at obtaining a prior (as we shall see in Section 6). Finally, additional information is provided such as the link counts (the vector Y) and routing information used to construct the routing matrix for the studied network topology. The estimation part includes computing the parameters of the assumed probability distribution—parameters that maximize the likelihood of observing the measured link counts on the given routing matrix. Once these parameters are obtained, the output traffic matrix is populated with the average for each entry. A final step called *proportional fitting* adjusts the estimated average values to satisfy as close as possible the constraints imposed by the link counts.

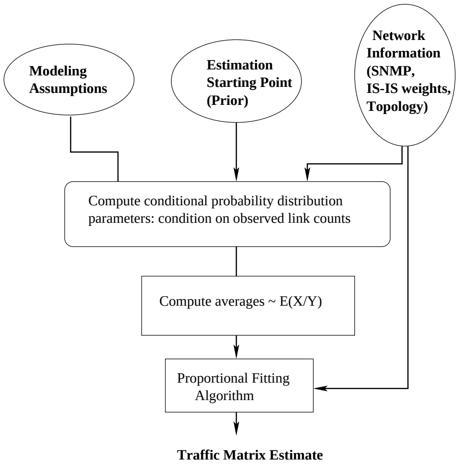


Figure 1: General diagram of statistical techniques

Few statistical inference approaches have been proposed to date [12, 11, 2, 13]. The basic idea behind the first three approaches is to first define a probabilistic model describing the bandwidth of OD pair flows. First, estimation techniques, such as maximum likelihood estimators, are used to estimate all model parameters. Then, the traffic matrix is populated with a conditional expectation capturing the mean bandwidth of the flow between two end nodes, conditioned on the observed SNMP link counts. For example, Vardi [12], and Tebaldi and West [11] define a probabilistic model that assumes origin-destination flows follow a Poisson distribution. Cao et al. [2] assume instead that origin-destination flows follow a Gaussian distribution. To estimate the model parameters, Tebaldi and West [11] use a Bayesian approach, combining Gibbs sampling with Monte Carlo simulations, while Cao et al. [2] use an Expectation Maximization (EM) algorithm to compute maximum likelihood estimates.

As we said in Section 1, the focus of this paper is on two-step approaches for the TM inference problem. Recall that in such an approach, first a mechanism to generate good starting points is used. Second, the generated starting point is provided to a statistical inference technique. In this Section we describe the components of each step both for our approach and alternatives approaches proposed elsewhere [12, 11, 2, 13].

3.1 Our Approach

3.1.1 **First step**: *Mlogit and Linear Choice Models*

Medina el al. [9] proposed an approach to modeling the fanouts of nodes using a *choice model framework* derived from Economic Consumer Theory. In this approach, the engineering characteristics of nodes in the network determine the likelihood that a byte will be transferred from node i to node j. Some degree of uncertainty in the process is allowed by incorporating a random component into the choice models. More specifically, in [9], the utility U^i_j that a given ingress PoP i gains from choosing to send a packet to PoP j, is the sum of a deterministic component, V^i_j, and a random component, ϵ^i_j. Since a random component is included for modeling the uncertainty, the utility function becomes a random variable. Therefore, the probability that PoP i selects PoP j from a set of egress PoPs, representing the fanout intensities α_{ij}, equals the probability that the random variable U^i_j has the largest value among the utilities of all alternatives.

In general, given K attributes for each PoP and letting $f(A^i_k)$ ($g(A^j_k)$) denote a function of the k^{th} attribute of ingress PoP i (egress PoP j), V^i_j is given by:

$$V^i_j = \sum_{k=1}^{K} \beta_k f(A^i_k) + \sum_{k=1}^{K} \beta_{K+k} g(A^j_k) + \gamma_j \tag{3}$$

where β_k defines the relative importance of attribute k with respect to the others, and γ_j is a scaling term.

Many different choice models can be defined based upon how many and which combination of attributes are included in the deterministic component. Assuming Gumbel distribution for the random uncertainty, which is a good approximation to a Gaussian distribution, the so-called *multinomial logit* or *mlogit* model is derived in which the probability of PoP i choosing a given egress PoP j is given by [9]:

$$\alpha_{ij} = \frac{e^{V^i_j}}{\sum_{k \in C} e^{V^i_k}} \tag{4}$$

where C is the set of egress PoPs. Therefore, the traffic between a pair of PoPs can be modeled by:

$$X_{ij} = O_i \alpha_{ij} \tag{5}$$

where O_i represents the total outgoing bytes sent into the network by PoP i. Intuitively, the mlogit function captures behavior in which a few traffic exchanges are large and dominate the overall characteristics of the traffic matrix, and in which there can be great differences between small and large traffic exchanges.

In this paper, we also consider a variant of choice models we call *Linear Choice models*, in which the form of the mlogit function is simplified by eliminating the exponential function at both the numerator and denominator of Equations (4) as follows:

$$\alpha_{ij} = \frac{V_j^i}{\sum_{k \in C} V_k^i} \tag{6}$$

For the linear-choice models we set the weights of the V_j^i function to 1, yielding α_{ij} values that are linearly correlated with the attributes of the PoPs.

3.1.2 **Step 2:** *Expectation Maximization Algorithm*

We use the framework established by Cao et al. [2]. Let $Y_t = (Y_t^1, \dots, Y_t^L)$ be a vector of observed traffic counts at time t on L links, and let $\lambda = (\lambda_1, \dots \lambda_m)$ be the vector of mean rates, where m is the number of OD pairs. It is common in these kinds of problems to assume some kind of relationship between the mean and the variance. Without such an assumption the variances, and possibly covariances, would also need to be estimated. This may drive the number of variables to estimate very high. We therefore assume that the variance and the mean of traffic rates can be related by $\sigma_i^2 = \phi \lambda_i^c$. The value of c can be fixed to a known value or estimated over empirical data.

The parameters to be estimated in this framework are $\theta = (\lambda, \phi)$. We wish to estimate θ by a maximum likelihood criteria. The log-likelihood of the observed traffic values $(Y_1, ..., Y_T)$ can be calculated as:

$$\begin{aligned} l(\theta|Y_1, ..., Y_T) &= -\frac{T}{2}\log|R\Sigma R^{'}| \qquad (7) \\ &\quad -\frac{1}{2}\sum_{t=1}^{T}(Y_t - R\lambda)^{'}(R\Sigma R^{'})^{-1}(Y_t - R\lambda) \end{aligned}$$

where Σ is the covariance matrix.

The maximum likelihood estimate $\hat{\theta}$ is defined as:

$$\hat{\theta} = \arg\max_{\theta} l(\theta|Y_1, ..., Y_T)$$

As Σ is related to λ there is no analytic solution to the above optimization problem. Even if it remains possible to do a brute force resolution, however as the inversion of $(R\Sigma R^{'})$ is inside the optimization, it might be hazardous and difficult. We therefore choose to use an EM approach to do the optimization. The EM method replaces the previous optimization problem by an iterative procedure where at each step a conditional expectation function Q is optimized.

In the problem under study the complete data log-likelihood can be obtained from:

$$l(\theta|X_1, ..., X_T) = -\frac{T}{2}\log|\Sigma| - \frac{1}{2}\sum_{t=1}^{T}(X_t - \lambda)^{'}\Sigma^{-1}(X_t - \lambda)$$

The EM conditional expectation function is defined as follows:

$$\begin{aligned} Q(\theta, \theta^k) &= E(l(\theta|X)|Y, \theta^k) \\ &= -\frac{T}{2}(\log|\Sigma| + \mathrm{Tr}(\Sigma^{-1}W^{(k)})) \\ &\quad -\frac{1}{2}\sum_{t=1}^{T}(u_t^{(k)} - \lambda)'\Sigma^{-1}(u_t^{(k)} - \lambda) \end{aligned} \tag{8}$$

where

$$\begin{aligned} u_t^{(k)} &= \lambda^{(k)} + \Sigma^{(k)}R'(R\Sigma^{(k)}R')^{-1}(Y_t - R\lambda^{(k)}) \\ W^{(k)} &= \Sigma^{(k)} - \Sigma^{(k)}R'(R\Sigma^{(k)}R')^{-1}R\Sigma^{(k)} \end{aligned}$$

where the terms $u_t^{(k)}$ and $W^{(k)}$ are the conditional mean and variance of X given both Y and the current estimate θ^k. Tr(.) denotes the trace of a matrix, i.e. the sum of the diagonal elements.

Each iteration of the EM method consists of two steps: one expectation step (usually called the E-step) and one maximization step (called the M-step). The E-step consists of calculating the conditional expectation function $Q(\theta, \theta^k)$ as per Equation (8), by using the k^{th} estimate of θ, namely θ^k. In the M-step, the new value $\theta^{(k+1)}$ is obtained by maximizing the conditional expectation function:

$$\theta^{(k+1)} = \arg\max_{\theta} Q(\theta, \theta^k)$$

It can be shown that θ^k converges to a minima of the likelihood function.

Implementation of EM Algorithm: The optimization problem involved in the M-step can be solved by finding the value that drives the gradient of the function Q to zero, that is, $\frac{\delta Q}{\delta \theta}|_{\theta = \theta^{(k+1)}} = 0$. In [2] it was shown that this is equivalent to solving the following nonlinear equation:

$$\begin{aligned} 0 &= c\phi\lambda_i^c + (2-c)\lambda_i^2 - 2(1-c)\lambda_i b_i^{(k)} - c a_i^{(k)}, \; i = 1, \cdots, m \\ 0 &= \Sigma_{i=1}^{m}\lambda^{-c+1}(\lambda_i - b_i^{(k)}) \end{aligned} \tag{9}$$

where

$$\begin{aligned} b_i^{(k)} &= \frac{1}{T}\sum_{t=1}^{T} m_{t,i}^{(k)} \\ a_i^{(k)} &= w_{ii}^{(k)} + \frac{1}{T}\sum_{t=1}^{T}(m_{t,i}^{(k)})^2 \end{aligned}$$

The authors in [2] replace the classical EM method by a modified EM method where at each step $\theta^{(k+1)}$ is updated using a Newton-Raphson or a second order method. The convergence of this modified EM method is reported to be slow, and singularity problems appear frequently when inverting the $(R\Sigma^{(k)}R')^{-1}$ term at each iteration. This is mandatory for calculating $u_t^{(k)}$ as well as $W^{(k)}$. Because the number of iterations of the EM algorithm could be very large in this approach, the problematic matrix inversion step would be carried out many times, significantly increasing the complexity of the solution approach.

In this paper we propose two modifications to the above algorithm aimed at obtaining a fast version of the EM approach for this problem. As TM estimation gets applied to networks with larger number of nodes, such as router-to-router TMs, scalable and fast EM algorithms become essential. The two improvements are the following:

- We convert the routing matrix R to one which is a linear transform of the original matrix and looks as close to an identity matrix as possible. We do so by transforming the matrix to a reduced echelon form. Having an R matrix in this form enables the optimization procedure to run much more quickly.

- As suggested in [2], we transform the optimization problem involved in the M-step of the EM method to solving a non-linear equation. However we solve this equation using sophisticated numerical techniques suited to large scale problems and we follow closely the EM algorithm, *i.e.* we set exactly $\theta^{(k+1)} = \arg\max_\theta Q(\theta, \theta^k)$.

The combination of these two ingredients speeds up significantly the optimization steps of the EM algorithm. Next, we describe in more detail the two steps involved in our implementation.

Echelon Forms: The goal here is to transform the extended routing matrix R into a format more suitable for the optimization step. For this purpose we rewrite the R matrix in a reduced echelon form. Computing the reduced echelon form is merely taking a linear transform of the R matrix and thus does not change the solution sought.

There are two reasons for doing this. First, the result of this step may yield some rows in which all elements are zero except for one element that is a one. The corresponding column in which this 'one' is located identifies an OD pair that in fact is explicitly known and does not need to be estimated. This OD pair can be removed from the estimation process and we thus reduce the dimension of the problem and the number of parameters that need to be estimated. Second, feeding an EM algorithm with matrix that has large component of it resembling an identity matrix gives a numerical advantage, as it will lead to a more sparse matrix and less error propagation.

EM steps: The last improvement provides a good deal of the speedup obtained in our method. Instead of obtaining $\theta^{(k+1)}$ as suggested in [2], by a Newton-Raphson or second order method, we assign $\theta^{(k+1)}$ such that $\theta^{(k+1)} = \arg\max_\theta Q(\theta, \theta^k)$. This optimization problem is carried out by solving a set of nonlinear equations using a procedure based on least squares estimation that uses a trust region method and an interior-reflective Newton method. This was implemented using the optimization toolbox of Matlab [7]. With this approach we follow precisely the EM method whereas the method proposed by [2] is a modified approach.

Generally we found that our EM method converges in about 10 steps, because during the optimization of $Q(\theta, \theta^k)$, the values of $u_t^{(k)}$ and $W^{(k)}$ do not change, thus we only need to carry out the costly matrix inversion operation once in every step, whereas the modified approach proposed in [2] needs to do the matrix inversion hundreds (and sometimes thousands) of times. As example, for a problem with 169 OD pairs, the estimation processs runs in under 1 minute on a modern laptop computer.

3.2 Other Approaches

3.2.1 **Step I:** *Gravity Models*

Gravity models are trip distribution models that have been widely used in transportation applications for estimating traffic demands between urban areas [10]. gravity model says that the trip interchange between zones in an urban area is directly proportional to the relative *attraction* of each of the zones and inversely proportional to some function of the *separation* between zones. In the context of the traffic estimation problem, we want to relate the amount of data exchanged between two nodes to the attraction, the ability of attracting data sent by other nodes, and some *friction* factor that influences how much data actually flows between the two nodes.

A general formulation of a gravity model may be given by the following equation:

$$X_{ij} = \frac{f(R_i, A_j)}{g_{ij}} \tag{10}$$

where $f(.)$ is a non-decreasing function, X_{ij} is the traffic volume from i to j, R_i is a parameter representing *repulsive* factors which are associated with "leaving" i, A_j is a parameter representing *attractive* factors related to "going" to j, and g_{ij} represents the *friction* factor between i and j.

Since X_{ij} is a fraction of the total amount of traffic coming out of PoP i, a simple gravity model formulation is given by rewriting the general Equation (10) as $X_{ij} = O_i \alpha_{ij}$.

Note that this formulation is identical to the choice model formulation, leaving the fanout intensity factor, α_{ij} as a variable to be defined. In this model O_i is the repulsion factor, and it reflects the amount of traffic PoP i dumps into the network.

In [13], two simple and elegant gravity models for generating starting points for traffic matrix estimation are proposed. Their first model is called a "simple gravity model" while their second model is called a "generalized gravity model." In this paper we consider the simple gravity model for our comparative purposes. In this model, the friction factors in Equation (10) are assumed to be constant. Despite of such assumption being the simplest form for the friction factors, the formulated model does a good job at producing reasonable starting points to be input to a statistical approach.

At the PoP-to-PoP level, the main idea is that the traffic exchanges between PoPs in the network should be proportional to the volumes of traffic entering and exiting the end nodes in any OD pair. In a nutshell, the gravity model at the PoP level is given by:

$$X_{ij} = O_i \frac{T_j^{out}}{\sum_k T_k^{out}} \tag{11}$$

where O_i is defined as above, and T_j^{out} is the total amount of bytes leaving the network through PoP j. Note that this gravity-based formulation is similar to the linear-choice formulation.

It is important to note that the choice-model framework described in Section *3.1.1* actually defines a family of models for starting points. For example, if we remove the random component from the utility function in the derivation of the *mlogit* model, we would obtain a similar model without exponential terms. We call such a model a *linear choice model* or just *linear model*. The formulation of the simple gravity model of this section corresponds to an instance of a linear choice-model with a single PoP attribute.

3.2.2 **Step I:** *Other Models*

A common approach to the generation of starting points for the estimation procedure has been to assume some underlying *standard* distribution for the elements of the traffic matrix and then synthetically populate starting points by generating random values according to the chosen distribution. For example, the technique proposed in the pioneering work of Vardi [12] assumes a Poisson distribution for the underlying traffic matrix. Therefore we may generate starting points for such a technique by populating synthetic traffic matrices according to a Poisson distribution. The EM approach proposed in [2] is developed based on the assumption that elements of the underlying traffic matrix are distributed according to a Gaussian distribution.

In [2], a simple mechanism for generating starting points is also proposed. That mechanism generates constant starting points where the constant value of each entry in the TM is a weighted sum of average link utilization levels where the weights are set according to the number of OD pairs traversing each link on the OD-pair path.

We experimented as well with such constant starting points.

We included in our experimental framework more *extreme* distributions such as multi-modal and skewed distributions. Investigating these distributions is important since they should expose the behavior of the studied statistical techniques in the presence of "unreasonable" starting points. Note that by *reasonable starting point* we mean starting points that are not radically different from the actual distributional shape of the underlying traffic matrix we are seeking to estimate.

3.2.3 **Step II:** *Weighted Least Squares*

We refer to the approach in [13] as the Weighted Least Squares Estimation (WLSE) method. This method was proposed as part of an estimation method coined by the authors as *tomogravity*. Tomogravity consists of obtaining a starting point using a gravity model (see Section *3.2.1*), and then reducing the error in the starting point by using *quadratic programming*. The error-reduction step seeks to find a solution that minimizes the distance to the starting point while at the same time satisfying the restrictions imposed by the system $RX = Y$.

4. MEASUREMENTS USED

The work presented in this paper was done in the context of a Tier-1, continental-US backbone network. We use packet traces from several monitored PoPs, as well as SNMP data collected for all backbone links. We use information computed from the packet traces, together with SNMP data, for calibrating and validating the studied starting point models, and for testing the performance of our EM algorithm.

4.1 Packet Traces

We used two sets of full packet traces, which were collected on September 5, 2001 for a time interval of 12 hours, and on November 21, 2002 for an interval of 10 hours. These two sets contain packet traces for 3 POPs and 2 POPs, respectively. The collection of these packet traces was performed by monitoring sets of links at each monitored PoP (about 10 links per PoP) in the studied backbone network. Specifically, we monitored aggregated access links (customers), which connect access routers to core routers, peering links and inter-PoP backbone links. The collected packet traces provide us with *measured* estimates of actual rows of the corresponding POP-to-POP traffic matrix.

In order to compute actual rows of a TM from packet traces we apply a mapping procedure that takes as input the destination address of an incoming packet and outputs the egress PoP through which the packet will leave the network. The implementation of such a mapping mainly uses BGP routing information and, for some cases in which BGP information is not enough to establish the mapping, traceroutes are used. Using our mapping procedure we are able to map more than 99% of the monitored packets. We can then compute the fraction of all packets that were sent from a monitored (ingress) PoP to every egress POP, i.e., the fanout intensities α_{ij}.

4.2 SNMP Data

The Simple Network Management Protocol (SNMP) provides per-link information regarding the number of bytes flowing through each link in the network over some interval of time (e.g., 5 minutes). This information is systematically collected from all links in the backbone network and we use it at different aggregation levels for computing POP attributes as well as for evaluating performance improvements gained by the combinations of starting point and estimation techniques we have studied. Specifically, from SNMP data we draw information about aggregated customer, peering, inter-POP, and intra-POP link utilization levels in the network. For each of these link types we determine the average used capacity over a certain interval of time. SNMP provides per-link byte-count information at a minimum granularity of 5 minutes.

Note that the SNMP data used to compute the link-utilization statistics were collected during the same period as the packet traces, that is, 12 hours on September 5, 2001, and 10 hours on November 21, 2002.

4.3 Time Scales

The characteristics, availability and applications of measured or estimated network traffic demands depend to a large extent on the time granularity used to collect the data.

On one hand, the collected packet traces in the studied backbone network are gathered at the time granularity of packet arrivals. For this work, we pre-process the packet traces to compute a basic aggregation level capturing the number of packets and bytes per second arriving to the measured links. Such minimal level of aggregation can be further increased as needed. On the other hand, the SNMP link utilization data is collected at a time granularity of 5 minutes. As with packet traces, higher levels of aggregation, always in multiples of 5 minutes, are obtained as needed. For example, if we want to estimate a TM over a one-hour time period, the SNMP link counts would be aggregated by summarizing 12 5–minute measurements with an average value.

In our study, we are interested in aggregation levels of at least one hour since we are targeting traffic engineering and network management tasks for which changes in POP-to-POP traffic exchanges over finer timecales are not of interest.

5. EVALUATION METHODOLOGY

One of the challenges that must be tackled when investigating inference mechanisms to estimate network traffic demands is the issue of how to validate the results. Ideally, we would have complete accurately measured network traffic demands to compare the results of the inference process against them. However, if we had an effective and efficient mechanism to obtain such accurate measurements we would not need to rely on statistical inference. Alternatively, we would like to obtain substantial information about network traffic demands using mechanisms such as Netflow or BGP Policy Accounting. Doing so, however, is difficult since these mechanisms may impose a significant burden on routers and consequently may degrade the performance of the network. In this section we describe the approach we adopted for validating our EM algorithm and for assessing its convergence behavior.

5.1 Empirical Model for Synthetic TMs

In general, previous studies and comparative evaluations have relied on limited actual network information and on synthetically generated traffic matrices based on seemingly strong assumptions regarding the underlying distributions of the actual traffic exchanges between origin-destination (OD) pairs [12, 2, 11, 9]. For example, a common approach has been to assume that OD demands are distributed according to a Gaussian or Poisson distribution. Alternatively, more skewed distributions (e.g. Bimodal) have been proposed for testing purposes as well. Although making such assumptions may be useful in terms of agreeing with the intrinsic assumptions made by the statistical technique used, they may not be representative of the actual characteristics of OD traffic exchanges [9].

The validation approach we use in this paper makes use of what we call an *empirical model* for synthetic traffic matrix generation.

This very simple empirical model consists of two steps which use the measurement data described in Section 4. Specifically, we use packet traces collected at a Tier-1 backbone network to determine an empirical distribution of the PoP-to-PoP fanouts, and use SNMP utilization information to establish a hierarchy of *importance* among egress PoPs. The procedure is as follows:

(1) Determine empirical distribution of fanouts: As described in Section 4, we have access, on different dates, to information regarding actual PoP-to-PoP traffic exchanges for up to three PoPs. Despite the very large amount of data collected for each of the measured POPs, we are capturing only a fraction of the total traffic flowing through each PoPs. However, we believe that by carefully choosing the POPs and links from which packet traces are collected, the traffic demand information gathered in the process would capture an important component of the behavior of traffic exchanges. Using an empirically derived distribution of PoP fanouts for, say, three POPs, we generate random fanouts. Figure 2 shows an example empirical complementary cumulative distribution function of fanouts and the associated fit with a simple single-exponential function.

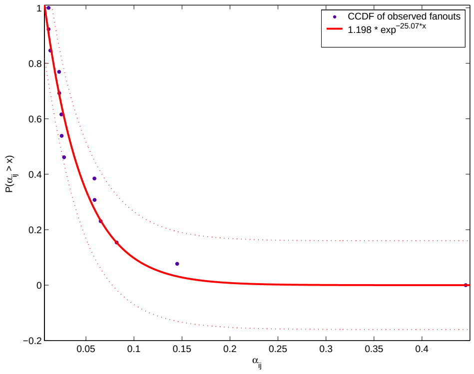


Figure 2: Fit of empirical CCDF for fanouts distribution

(2) Define egress PoP ranking: Building on the premise that PoPs are engineered in correlation with the amounts of data they would need to handle, we establish a POP ranking based on utilization information about the PoPs as given by SNMP data.

Specifically, for each egress PoP, we rank it according to its individual attributes, such as utilization levels for incoming and outgoing customer, peering and inter-POP links. Then an overall ranking is determined by summing the individual rank values for each egress PoP. If $Rank_k^j$ is the rank of egress PoP j with respect to attribute A_k, then we compute the overall rank of PoP j as $\sum_k Rank_k^j$.

(3) Match random fanouts to ranked egress PoPs: the last step consists of sorting the random fanouts obtained in step one for each ingress PoP, and assigning them to the egress POPs in order according to their rank established in step two.

Although this empirical model is very simple, it is aimed at providing synthetic target traffic matrices that are in some sense more realistic and can provide more meaningful evaluation test cases.

5.2 Synthetic–data Experiments

Synthetic data is very useful to evaluate the performance of traffic matrix estimation techniques since it enables us to assess their behavior with respect to whole matrices rather than partially measured TMs. By performing synthetic–data experiments we can better assess the errors yielded by the evaluated techniques, determine the distribution of errors among the estimated OD traffic demands, etc.

In this step of the evaluation process, we use the empirical model described in Section 5.1 to generate a target synthetic traffic matrix. As depicted in Figure 3, we route this target matrix onto the topology of the studied network to obtain a set of synthetic link counts equivalent to the set of link counts that would be provided by SNMP data. Then, we generate a starting point for the estimation procedure according to any of the models described in Section 6. We pass the link counts and the starting point to the chosen estimation technique to obtain an estimated TM. Finally, we compare the output of the estimation to the target TM to assess the error incurred by the estimation procedure.

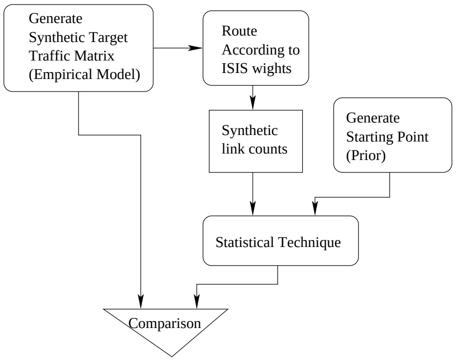


Figure 3: Performance evaluation for synthetic cases

Both the synthetic target TM and the chosen starting point are generated consistently using packet traces and SNMP data corresponding to the same period of time. Once the synthetic fanouts ($\hat{\alpha}_{ij}$) have been defined (cf. Section 5.1), the synthetic target TM is populated using actual SNMP data to determine the total amount of bytes leaving PoP i via inter-PoP links as follows:

$$X_{ij} = O_i \times \hat{\alpha}_{ij} \tag{12}$$

5.3 Real–data Experiments

The next step is to evaluate estimated network traffic demands with respect to their *goodness-of-fit* or closeness to measured traffic demands. The approach is similar to the one described in Section 5.2 with two differences. First, we do not have a full target traffic matrix which would be used as before to generate a set of consistent link counts. Second, after the estimation procedure finishes, the comparison is not done against a full synthetic TM. Instead, we feed the given estimation technique with a set of actual SNMP link counts and a starting point generated in the same way as before. We then take the output estimated TM and compare the rows that correspond to the actual measured rows to assess the goodness-of-fit of the estimation.

As an example and following the diagram in Figure 4, suppose we have measured the third row of the actual traffic matrix for a given date, say November 21, 2002. We extract from the SNMP data repository, link utilization information for the same time and aggregation intervals. Then we generate a starting point according to, say, a choice model (cf. Section 6), and feed these into the EM algorithm. We then take the third row of the estimated TM and compare it against the measured row we have from the beginning.

For all experiments, the starting points are calibrated and populated with data (SNMP and packet traces) corresponding to the input data fed into the estimation technique used.

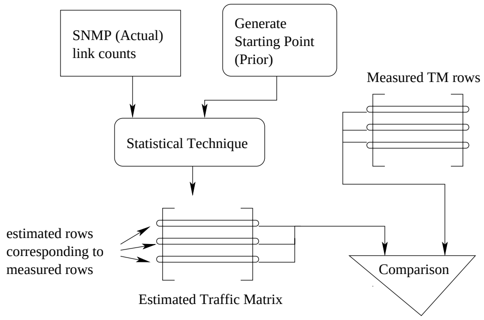


Figure 4: Performance evaluation for real-data test cases

To quantitatively compare the experimental results, we plot entries of the estimated traffic matrix versus the target traffic matrix. The closer such a plot follows a linear trend the better is the mean quality of the estimated traffic matrix. Furthermore, we need to evaluate the dispersion of the estimation points around the mean. We compute this dispersion using the well-known Pearson's coefficient R [6]. The closer R is to one the better the estimation is.

6. OBTAINING REASONABLE STARTING POINTS

Although in general we may generate starting points arbitrarily or according to any *standard* distribution (e.g. Gaussian, Poisson, etc.), the convergence behavior of statistical techniques may be significantly influenced by the characteristics of the provided starting point [9]. In this section we describe different approaches for the modeling and population of reasonable starting points to be provided as input to statistical inference techniques for the traffic matrix estimation problem.

6.1 Calibration Mechanisms

A model for generating starting points may need to be calibrated in order to assign concrete values to its parameters. After such calibration has been performed we would have a specific instance of the model, which will be used to populate traffic matrices to be used as starting points.

6.1.1 *Calibrating choice models*

Choice models need to be calibrated so as to specify the coefficients β_k in Equations (3). To that end, packet traces and SNMP data are used in the calibration process. Packet traces, aggregated at the PoP level, enable us to compute individual TM rows for the ingress PoPs at which the packet traces were collected. These measured TM rows are used as the equivalent of sample surveys of the decisions made at the ingress PoP as to where to send the bytes it generates, and they are provided to the calibration procedure. From SNMP data we extract PoP-to-PoP information regarding the capacity and utilization information for incoming and outgoing customer and peering links, as well as for inter-PoP links in the studied tier-1 backbone network. To discuss the use of this information we use the following notation. Let D_j denote the total amount of traffic received by egress PoP j from the backbone, which is computed by summing the SNMP link counts of all inter-PoP links entering PoP j. Let O_i denote the total traffic leaving PoP i, which is computed by summing the SNMP link counts of all inter-PoP links exiting PoP i. Let C_i^{in} (C_i^{out}) denote the used capacity for incoming (outgoing) customer links at PoP i. Finally, let P_i^{in} (P_i^{out}) denote the used bandwidth of incoming (outgoing) peering links for PoP i.

	O_i	D_j	C_i^{in}	C_j^{out}	P_i^{in}	P_i^{out}
O_i	1.0000	0.5992	0.9217	0.6032	0.5587	0.2167
D_j	-	1.0000	0.4316	0.7961	0.0767	0.3341
C_i^{in}	-	-	1.0000	0.5261	0.8366	0.3182
C_j^{out}	-	-	-	1.0000	0.2730	0.5386
P_i^{in}	-	-	-	-	1.0000	0.3744
P_i^{out}	-	-	-	-	-	1.0000

Table 1: Correlation coefficient of PoP attributes

Intuitively, the six most useful attributes should be O_i, D_j, C_j^{out}, C_i^{in}, P_j^{out} and P_i^{in}, for ingress PoP i and egress PoP j. We want to include attributes in our choice-models that are as uncorrelated as possible, since otherwise we may have co-linearity problems. To assess the correlation among different PoP attributes, we calculated the correlation coefficient between all pairs of attributes (see Table 1) [3]. Only the pairs (O_i, C_i^{in}), (D_j, C_j^{out}) and (C_i^{in}, P_i^{in}), have correlation coefficients higher than 0.65. This implies that a model should not include both the members of these pairs. Note that the relatively high correlation level for these pairs is expected. In the first case, (O_i, C_i^{in}), it is intuitive that the volume of data on the incoming customer links at an ingress PoP is correlated to the amount of traffic the PoP dumps onto the inter-PoP backbone links (assuming that most of the customer traffic wants to cross the backbone and not exit immediately at the same PoP). Similarly for the pair (D_j, C_j^{out}), there must be a strong correlation between the amount of traffic entering an egress PoP j from the backbone and exiting the PoP on its customer links. The correlation between (C_i^{in}, P_i^{in}) is a bit more surprising. Perhaps this indicates that if an ingress PoP is small (large) it will have similarly small (large) numbers of customer and peering links, respectively.

Table 2 describes, in terms of the included attributes, the three choice models we have included in the results of this paper. These models behave best with respect to yielding lowest errors and producing reasonable starting points for the TM estimation procedure. Model I uses only two PoP attributes given by the total amount of bytes entering and exiting a PoP. Model II uses instead the volume of traffic leaving the network at PoP j via customer and peering links. Finally, Model III replaces the use of O_i by the total volume of data coming into the network at PoP i via customer and peering links.

Model	Attributes
I	(O_i, D_j)
II	$(O_i, C_j^{out}, P_j^{out})$
III	$(C_i^{in}, P_i^{in}, C_j^{out}, P_j^{out})$

Table 2: Attributes included in each model

The actual calibration of the choice model requires the calculation of the coefficients β_k in Equations (3) so as to match the α_{ij} for the measured PoPs i. This is done by curve-fitting to the mlogit function using a maximum likelihood estimation implemented in the Econometrics toolbox of Matlab [7]. Once the model is calibrated, we compute the remaining fanout values α_{ij} using (4), and the full prior TM is then populated using (5).

6.1.2 *Linear-choice and Gravity model Calibration*

The linear-choice and gravity models do not need to undergo a calibration procedure since they do not have coefficient values.[4]

---

[3] Since this matrix is symmetric, we only include half the values for ease of readability.

[4] In these models, the coefficients of the PoP attributes are all set to

These models need to be populated by extracting from the SNMP archives the information they require. We can then generate the starting point using Equations (5) and (11), respectively.

6.2 Comparative Analysis

Figures 5-8, show a comparison between four different starting points, generated randomly, according to a skewed distribution, according to the gravity model, and according to the mlogit-choice model. Random starting points are obtained by generating random numbers around the mean of the target TM which gives them some *knowledge* of the target. Skewed starting points are generated such that, for a given ingress PoP, most of the egress PoPs would have a low fanout value while a few will have significantly larger fanout values.

Each figure shows a plot comparing the corresponding starting point against the target TM we seek to estimate and in the caption we indicate the value of the *Pearson* correlation coefficient (R^2), which measures the linear relationship between two data sets [6] and the value of the relative error for 70% of the starting point elements. The target TMs used throughout most of our experimental scenarios were generated using the empirical model described in Section 5.1. As can be observed, the value of R^2 increases and the value of the error decreases as we go from a random starting point towards a choice-model starting point. Since the value of R^2 is very low and the errors very high for the random and skewed starting points, we refer to them as *unreasonable* starting points. The gravity and mlogit-choice models produce priors that are scattered more around the values of the target TM and yield improved R^2 and error values. We call these *reasonable* starting points.

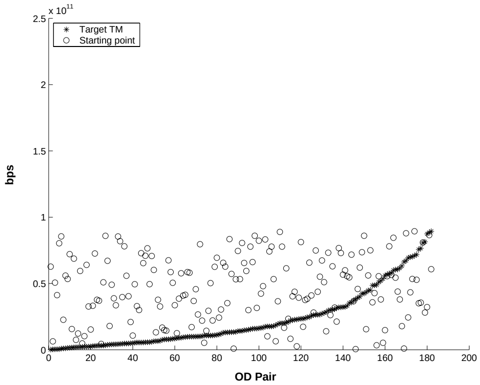


Figure 5: Rand-Empirical;$R^2 = 0.05$**;**$P[err < 114\%] = 0.7$

For each type of generated starting point, we compute the fanout values from the resulting starting TM. We then compare the observed fanouts, obtained form the actual measurements described in Section 4.1 against the corresponding starting-point fanouts. Figure 9 depicts an sample result of this comparison. We observe that the studied *reasonable* starting-point models produce similar results when compared to observed fanouts for two measured PoPs. Therefore, a conclusion to make is that both of these mechanisms may be used to produce starting points for statistical techniques. Furthermore, the more powerful the statistical technique is, the more resilient it will be, i.e. it will be more capable of recovering from lower quality starting points and produce reasonably accurate estimations. We further explore this issue in Section 7.

7. PERFORMANCE EVALUATION

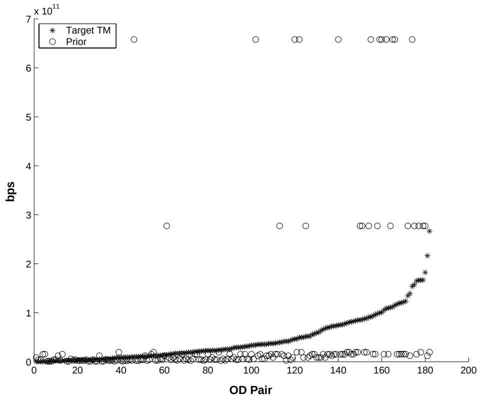


Figure 6: Skewed-Empirical; $R^2 = 0.1$**;** $P[err < 100\%] = 0.7$

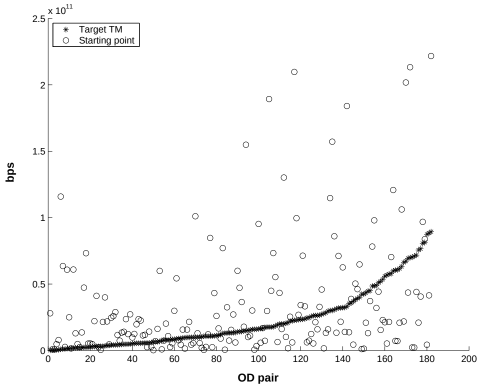


Figure 7: Gravity-Empirical; $R^2 = 0.3$**;** $P[err < 62\%] = 0.7$

In this section we discuss experimental results for the application of our two-step process for TM estimation. The motivation for the creation of models for generating starting points such as the gravity or choice models, was that the state-of-the art before then was to provide random starting points, or starting points generated based on the assumptions underlying the statistical technique to be used in the second step of the estimation process [9]. Therefore, we compare the convergence behavior of our modified EM algorithm and the WLSE technique described in [13] by varying the type of starting points provided.

7.1 Synthetic Target Experiments

This set of experiments was designed according to the methodology described in Section 5.2. Specifically, two different target TMs were generated using archived SNMP for September 5^{th}, 2001, and November 21^{st}, 2002. Recall from Section 6, that the calibration procedure for the choice models makes use of available measured rows of the actual traffic matrix. Therefore, in the synthetic case, we calibrate the choice models varying the number of rows (one to six rows) from the synthetic target TM used for their calibration. However, varying the number of rows used in the calibration did not affect significantly the estimation results and therefore we will refer to the choice models without specifying the number of rows used in their calibration. Figures 10 and 11 depict plots showing, for both the WLSE and EM methods, the 70th-percentile of errors for the estimation results obtained using various models of starting points. We consider 70% of the largest OD pairs to com-

---

1.

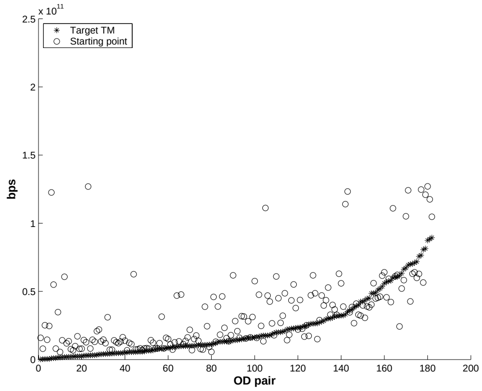


Figure 8: Choice-Empirical; $R^2 = 0.65$; $P[err < 40\%] = 0.7$

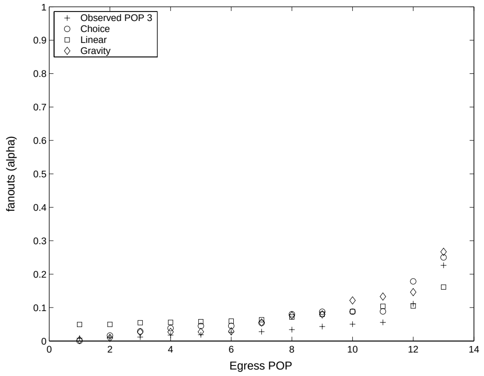


Figure 9: Observed vs. predicted fanouts for PoP3 TM row

pute the errors. Estimating small flows is difficult because they are usually close to zero and the difference from large to small flows is 3 orders of magnitude or higher. Furthermore, operators are mainly interested in the behavior of big TM elements because they consitute the majority of the traffic in the network. Thus, results of TM estimation efforts are usually evaluated considering only the errors incurred for the large TM components [9, 4, 14].

We can observe in the figures and in Table 3 that the EM method using different starting points yields lower error for a higher percentage of the estimated OD pairs. For example, using the EM and a choice-model prior, about 70% of the OD pairs are estimated with an relative error of less than 50%. The WLSE method achieves the same error value for only 30% of te OF pairs.

When choice models are used to generate the starting points, the performance of the EM exhibits little variability independently of the specific choice model used. For the WLSE method, the results vary slightly more but still the results are relative *stable* across choice models.

StepI/StepII	EM	WLSE
Choice	70%	30%
Gravity	60%	30%

Table 3: Fraction of OD pairs with less than 50% errors for different Step I mechanisms and Step II inference techniques

The previous plots show, for both the WLSE and EM methods, the errors for only 70% of the estimated TM elements. To qualitatively compare the estimations for all the TM elements, we plot in Figures 12(a)-12(f) scatter plots for all the entries of the estimated TM against the corresponding elements of the target TM being estimated. In these plots, the closer the data follows a linear trend the better is the mean quality of the estimated TM. Furthermore, we evaluate the *dispersion* of the estimated elements around the mean by using the *Pearson's* correlation coefficient, R [6].

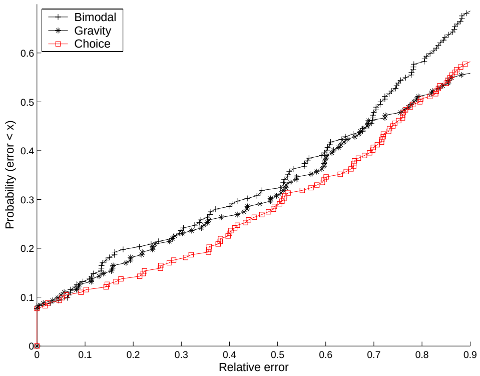


Figure 10: Convergence of WLSE method

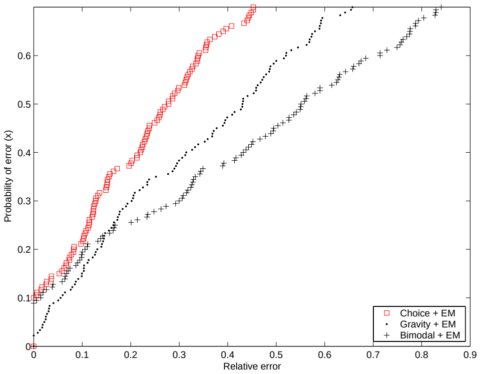
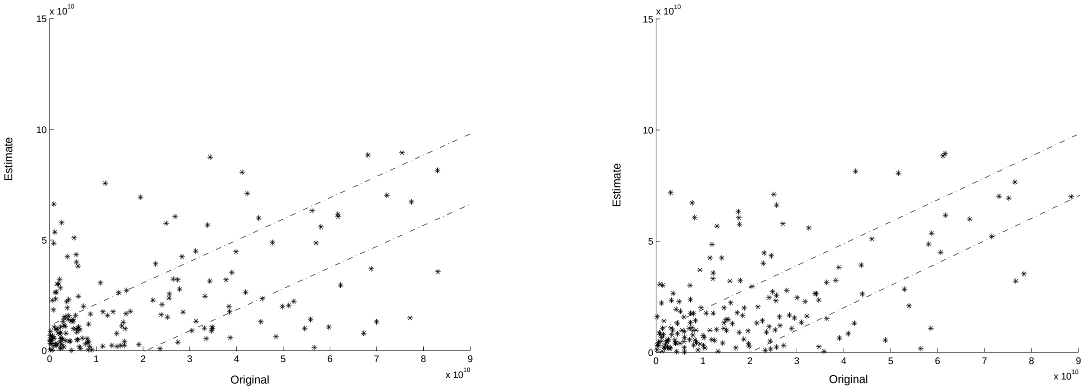


Figure 11: Convergence of EM method

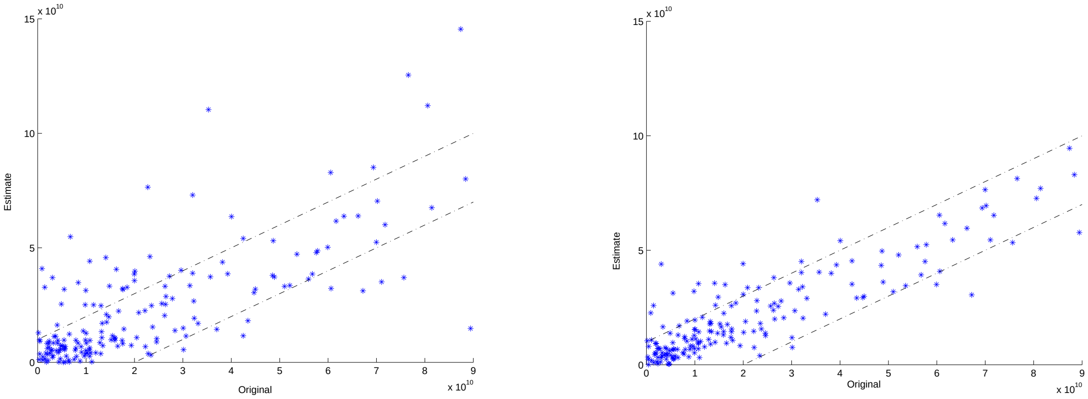
The scatter plots depicted show as well the flexibility of the two-step approach for inferring TMs. A given pair of plots sharing the same model for step one allow us to compare the performance difference obtained by changing the inference technique for step 2. For example, Figures 12(a) and 12(b) show the results of both the WLSE and EM methods when provided a random starting point. Also, Figures 12(b), 12(d) and 12(f), show the performance behavior of the EM method when provided with different types of starting points.

We can observe a trend of improvement for each method when we vary the starting point from a random, to a gravity-model and then to a choice-model starting point. In each case there is an increase in the value of R as well as a reduction in the value of the relative error for 70% of the estimated TM elements. In addition, we oberve that the use of the EM algorithm yields an additional 2-7% improvement in the errors as compared to the WLSE.

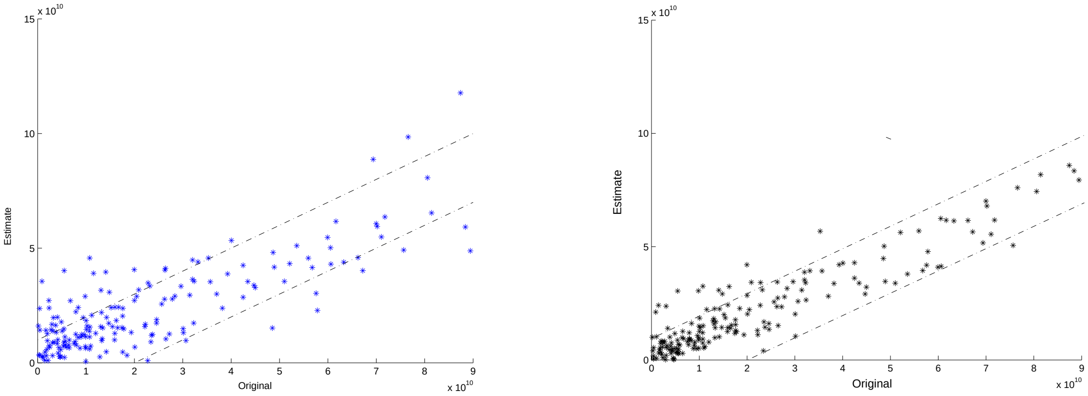
Choice models yield a improvement with respect to the gravity models but they perform similarly. Gravity models have the advantage of being simpler since they do not require a calibration procedure. Choice models, on the other hand, are more complex but also more general and reliable than the gravity models since they may incorporate significant amounts of network information into the estimation process and they attempt to capture the rationality behind



(a) Random + WLSE; $R^2 = 0.4$; $P[err < 45\%] = 0.7$ (b) Random + EM; $R^2 = 0.6$; $P[\text{err} < 35\%] = 0.7$



(c) Gravity + WLSE; $R^2 = 0.7$; $P[err < 32\%] = 0.7$ (d) Gravity + EM; $R^2 = 0.75$; $P[err < 30\%] = 0.7$



(e) Choice + WLSE; $R^2 = 0.80$; $P[err < 27\%] = 0.7$ (f) Choice + EM; $R^2 = 0.87$; $P[err < 20\%] = 0.7$

Figure 12: Original vs. Estimate Scatter plots for different combinations of starting point and inference technique

the choice behavior of the sending endpoints and it is likely that such behavior remains more stable across time. Gravity models are based on relationships among traffic demands and for Internet traffic those relationships are more likely to fluctuate. Finally, we note that gravity models assume that the proportionality factors are constant across PoPs while the choice models do not. The evaluation conducted by Medina et al. [9] shows that not to be the case for real PoP-to-PoP network traffic.

7.2 Real-data Evaluation

In this section we describe the results of experiments performed with actual but limited network traffic demands measured directly from the studied network. These experiments were designed as explained in Section 5.3. We apply the statistical techniques providing actual SNMP counts and a starting point generated from a model calibrated and populated with actual data collected during the same time period as the collection of the packet traces. Specifically, we use packet traces as described in Section 4.1 to compute measured TM rows for two PoPs (November 21^{st} data) or three PoPs (September 5^{th} data) — starting point models are calibrated using 1 or 2 PoP rows (corresponding to PoPs in PoP1, PoP2, or PoP3), leaving the remaining measured row for assessing the error after the TM estimation.

Figures 13 and 14 show the results of these experiments. We observe less differentiation between models and higher overall errors. The results show EM yielding better results than WLSE specially at the right end of the curve, that is, at higher volumes of traffic demand, but in general the difference between both methods is less pronounced than in the case of full synthetic TMs (cf. Section 7.1).

One reason is the fact of having to calibrate the choice models with a single row of the target TM. We expect as we become able to obtain more complete measurements, the advantage of our EM algorithm would become more pronounced.

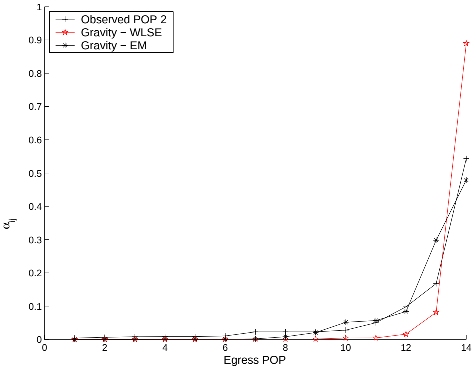


Figure 13: WLSE estimation of measured PoP2 TM row

8. CONCLUSIONS AND FUTURE WORK

In this paper we addressed the issues raised in [9] regarding the sensitivity to the quality of starting points and the lack of available network information by statistical techniques for inferring network traffic demands, by proposing and investigating a mechanism to generate starting points which incorporates significant information about the network. We showed that such mechanism does generate improved starting points for synthetic cases and we showed also showed that combining it with a good statistical technique also yields performance gains in terms of the estimation results.

We observed a trend of improvement for each method when we vary the starting point from a random, to a gravity-model and then to a choice-model starting point. With error gain improvements between 5% and 20%. We found that both the gravity and choice models for generating *informed* starting points produce initial estimates that are within a reasonable error range from the target TM being estimated. Although a choice-based prior TM produces better estimates than a gravity-based one, the latter is simpler in that no calibration with real data is required. It will be up to carriers to decide on their individual tradeoffs between accuracy and simplicity. Unlike arbitrary models (e.g. random, skewed, etc.), these starting-point models are *informed* by partial SNMP data and network measurements. Such informed starting-point TM models are crucial for the success of statistical estimation techniques such as EM and WLSE.

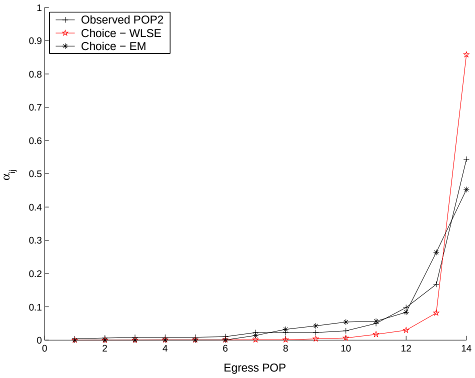


Figure 14: EM estimation of measured PoP2 TM row

We introduced a modified EM algorithm, which is much faster than conventional implementations. The modifications expand its iteration range in search for global optima, and makes the algorithm less sensitive to the quality of the starting point. Given a starting point mechanism, our EM algorithm consistently produces estimates which outperform that of the WLSE approach by $2-10\%$ error gains.

We compared choice starting-point models to an alternative simple gravity model for PoP-to-POP TMs. In the future we intend to compare against the extended gravity model proposed in [13]. This extended gravity model allows one to isolate separate traffic matrices such as peer-to-customer, customer-to-customer, and customer-to-peer TMs. We intend as well to study router-to-router level TMs. This is now possible given that both our fast EM algorithm and the WLSE algorithm would scale to work on much larger TMs.

The focus of this paper was on the use of a two-step approach for the TM estimation problem. Part of our future work is to expand the evaluation performed with alternative mechanisms and inference techniques for steps 1 and 2 of the studied approach. For example we plan on evaluating more complex gravity models and other mechanisms for generating informed starting points for step 1, as well as other statistical techniques for step 2.

9. REFERENCES

[1] G. Varghese C. Estan, S. Savage. Automatically Inferring Patterns of Resource Consumption in Network Traffic. In *ACM/SIGCOMM'03*, Germany, August 2003.

[2] J. Cao, D. Davis, S. Vander Weil, and B. Yu. Time-Varying Network Tomography. *J. of the American Statistical Association.*, 2000.

[3] N. G. Duffield and M. Grossglauser. Trajectory Sampling for Direct Traffic Observation. *Transactions on Networking*, pages 226–237, June 2001.

[4] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True. Deriving Traffic Demands for Operational IP Networks: Methodology and Experience. *ACM SIGCOMM*, August 2000.

[5] O. Goldschmidt. ISP Backbone Traffic Inference Methods to Support Traffic Engineering . In *Internet Statistics and Metrics Analysis (ISMA) Workshop*, San Diego, CA, December 2000.

[6] Peter Kennedy. *A Guide to Econometrics*. MIT Press, Cambridge, MA. USA, 1998.

[7] J. P. LeSage. Econometrics Toolbox. Technical report, University of Toledo, 2001.

[8] A. Medina, C. Fraleigh, N. Taft, S. Bhattacharyya, and C. Diot. A Taxonomy of IP Traffic Matrices. In *SPIE Workshop on Scalability and Traffic Control in IP Networks II*, Boston, MA, July 2002.

[9] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic Matrix Estimation: Existing Techniques Compared and New Directions. In *ACM SIGCOMM*, Pitsburgh, PA, August 2002.

[10] W.J. Reilly. *The Law of Retail Gravitation*. Fillsburg Publishers, New York, 1953.

[11] C. Tebaldi and M. West. Bayesian Inference of Network Traffic Using Link Count Data. *J. of the American Statistical Association.*, pages 557–573, June 1998.

[12] Y. Vardi. Network Tomography: Estimating Source-Destination Traffic Intensities from Link Data. *J. of the American Statistical Association.*, pages 365–377, 1996.

[13] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg. Fast Accurate Computation of Large-Scale IP Traffic Matrices from Link Loads. *To appear in ACM SIGMETRICS*, 2003.

[14] Y. Zhang, M. Roughan, C. Lund, and David Donoho. An Information-Theoretic Approach to Traffic Matrix Estimation. *ACM SIGCOMM*, August 2003.