Quick Traffic Matrix Estimation Based on Link Count Covariances

Ilmari Juva[1], Sandrine Vaton[2], Jorma Virtamo[1]

(1) Helsinki University of Technology, Espoo, Finland.

(2) ENST Bretagne, Brest, France.

Abstract

In this paper we consider the problem of traffic matrix estimation. As the problem is under-constrained, some additional information has to be brought in to obtain solution. If we have several link count measurements available, a natural candidate is to use the link count sample covariance matrix. We propose two computationally light-weight methods for traffic matrix estimation based on the covariance matrix, the projection method and constrained minimization method. The accuracy of these methods is compared with that of other methods using second order moment estimates by simulation under synthetic traffic scenarios.

Keywords: Traffic Matrix Estimation

1 Introduction

In traffic matrix estimation, the basic relationship between link counts $\boldsymbol{y}$ and traffic matrix $\boldsymbol{x}$ can be written as

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}, \tag{1}$$

where $\boldsymbol{A}$ is the routing matrix and $\boldsymbol{x}$ is the traffic matrix written in vector form, i.e. each component represents a traffic demand of an OD pair. Since in any realistic network there are many more OD pairs than links, the problem of solving $\boldsymbol{x}$ from $\boldsymbol{A}$ and $\boldsymbol{y}$ is strongly underdetermined. This means that accurate explicit solutions cannot be found, as there is an infinite number of solutions for $\boldsymbol{x}$ that satisfy equation (1). To overcome this ill-posedness, some type of additional information has to be brought in to solve the problem. Reviews of the proposed methods can be found e.g. in [6] and [7].

In [1], Vardi proposes a method using the second moment estimates to serve as the additional information to make the system identifiable. With the Poisson assumption, meaning that variance is equal to mean, the system becomes

$$\begin{pmatrix} \overline{\boldsymbol{y}} \\ \epsilon \boldsymbol{S}^{(y)} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} \\ \epsilon \boldsymbol{B} \end{pmatrix} \boldsymbol{\lambda} \tag{2}$$

where, as is explained in more detail in section 2, $\boldsymbol{S}^{(y)}$ is the sample link covariance matrix and $\boldsymbol{B}$ is the matrix of element-wise products of rows of $\boldsymbol{A}$. Coefficient $\epsilon \in (0, 1]$ defines how much weight is given to the second moment estimate in the final solution, and $\boldsymbol{\lambda}$ is the estimator for the mean of $\boldsymbol{x}$. This is a linear inverse positive, or LININPOS, problem and can be solved by numerical likelihood methods, such as the EM-algorithm. The solution obtained this way is minimizer of the Kullback-Leibler distance between the observed moments and theoretical values. If we instead minimize (2) in least square sense, the solution is easily obtained in closed form.

Vardi's method, however, does not give very accurate estimates, as was discovered by Gunnar et al. [4]. This is due to the fact that the Poisson assumption is not accurate in current IP networks. Cao et al. [2] generalize the maximum likelihood approach by assuming a Gaussian traffic distribution and assuming that the variance is related to the mean through a power-law. While this MLE approach is efficient and theoretically justifiable, the size of the problem in traffic matrix estimation requires the use of iterative numerical methods, such as the Expectation Maximization algorithm, which is computationally quite heavy.

The MLE relies on the fact that the system of first and second order link count statistics together make the system identifiable with regard to the first order OD-pair statistic, if there exists a functional relationship between the mean and the variance of OD-pair traffic. The commonly used relation is the power-law relation

But in fact the second order statistic for OD-pairs is identifiable based solely on the second order statistic of the link counts, as long as we assume independence among OD-pairs and a sensible routing scheme. This result is proven by Soule et al. [3]. Since we can analytically solve the variance of the OD-pairs by least squares method, and the power-law relation between variance and mean is assumed, we can then solve the traffic matrix from our variance estimate.

The benefit is that this does not call for numerical methods, and is thus extremely quick to calculate. The problem with this approach is that it does not take into account the link count equation 1, which is a stronger condition as opposed to the mean-variance relation which is only an assumption. Therefore, in this paper we propose two simple methods that incorporates this information into the solution obtained through estimation of the variance yet maintaining the computational simplicity of the model.

2 Solving OD-pair covariance matrix from link counts

The MLE relies on the fact that the system of first and second order link count statistics together make the system identifiable with regard to the first order OD-pair statistics, i.e. we are able to find solution for the likelihood equations if there exists a functional relationship between the mean and the variance of OD-pair traffic. The commonly used relation is the power-law relation

$$\boldsymbol{\Sigma} = \phi \cdot \operatorname{diag}\{\boldsymbol{\lambda}^c\}. \tag{3}$$

Here Σ is a diagonal matrix, because we assume independence between OD pairs. Let us denote the number of links by J and the number of OD-pairs by N. Then the vector form of traffic matrix $\boldsymbol{x}$ has the dimension $(N \times 1)$, link loads $\boldsymbol{y}$ has the dimension $(J \times 1)$.

First, let us define $\boldsymbol{S}^{(y)}$ as a $\frac{1}{2}J(J+1)$-vector containing diagonal and upper triangle elements of the link covariance matrix $\boldsymbol{\Sigma}^{(y)}$. Define $\boldsymbol{S}^{(x)}$ as a N-vector containing the diagonal elements of the OD-pair covariance matrix $\boldsymbol{\Sigma}^{(x)}$. $\boldsymbol{A}$ is the $(J \times N)$ routing matrix, whose element $A_{i,j}$ is 1 if OD pair x_j uses link y_i, and 0 otherwise. Then define a $(\frac{1}{2}J(J+1) \times N)$ matrix $\boldsymbol{B}$ that relates vector $\boldsymbol{S}^{(y)}$ to vector $\boldsymbol{S}^{(x)}$. A row of $\boldsymbol{B}$ is indexed by a compound index (ij) where $i = 1, \ldots, J;\ j = i, \ldots, J$, meaning that the index runs through $\frac{1}{2}J(J+1)$ values,

$$B_{(ij),k} = A_{i,k}A_{j,k} \qquad \begin{array}{l} i = 1, \ldots, J;\ j = i, \ldots, J \\ k = 1, \ldots, N. \end{array}$$

In vector form this reads,

$$
\boldsymbol{B} = \begin{pmatrix} \boldsymbol{A}_1 \star \boldsymbol{A}_1 \\ \boldsymbol{A}_1 \star \boldsymbol{A}_2 \\ \vdots \\ \boldsymbol{A}_1 \star \boldsymbol{A}_c \\ \boldsymbol{A}_2 \star \boldsymbol{A}_2 \\ \boldsymbol{A}_2 \star \boldsymbol{A}_3 \\ \vdots \\ \boldsymbol{A}_c \star \boldsymbol{A}_c \end{pmatrix},
$$

where $\boldsymbol{A}_i$ denotes the ith row of $\boldsymbol{A}$, and the componentwise product is denoted with the star ($\star$). Now the rows of $\boldsymbol{B}$ indicate the elements of $\boldsymbol{x}$ contributing to covariance between links i and j.

The measured link covariance matrix can be written as

$$
\boldsymbol{\Sigma}^{(y)} = \sum_k \sigma_k^2 \boldsymbol{a}_k \boldsymbol{a}_k^{\mathrm{T}}, \tag{4}
$$

where $\boldsymbol{a}_i$ is the ith column of $\boldsymbol{A}$. In component form we have

$$
\boldsymbol{\Sigma}_{i,j}^{(y)} = \sum_k \sigma_k^2 A_{i,k} A_{j,k}. \tag{5}
$$

Using the vector notation, the equation becomes

$$
\boldsymbol{S}^{(y)} = \boldsymbol{B}\boldsymbol{S}^{(x)}. \tag{6}
$$

This is in fact quite similar to (2) in the case where ϵ would be set very large, leading to the part $\boldsymbol{S}^{(y)} = \boldsymbol{B}\boldsymbol{\lambda}$ to dominate the equation. We just have the more general power-law relation instead of the Poisson assumption, so we cannot now just replace $\boldsymbol{S}^{(x)}$ with $\boldsymbol{\lambda}$.

Typically $\frac{1}{2}J(J+1) > N$ and equation (6) is overdetermined. The least square estimate (LSE) solution (see e.g. [5]), to the equation is

$$
\boldsymbol{S}^{(x)} = (\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}^{(y)}. \tag{7}
$$

3 Projection method

Now that we have an estimate for the variances of each OD-pair, it is straight forward to solve for an estimate of the mean by using the mean-variance relation (3).

$$
\boldsymbol{\lambda}_0 = (\phi^{-1}\boldsymbol{S})^{\frac{1}{c}}. \tag{8}
$$

If we assume c to be constant, we can solve for the ϕ that gives the best fit with regard to the link loads, i.e. which minimizes

$$
\begin{aligned}
f(\phi, c) &= (\overline{\boldsymbol{y}} - \boldsymbol{A}\boldsymbol{\lambda}_0)^{\mathrm{T}}(\overline{\boldsymbol{y}} - \boldsymbol{A}\boldsymbol{\lambda}_0) \qquad (9)\\
&= (\overline{\boldsymbol{y}} - \boldsymbol{A}(\phi^{-1}\boldsymbol{S})^{\frac{1}{c}})^{\mathrm{T}}(\overline{\boldsymbol{y}} - \boldsymbol{A}(\phi^{-1}\boldsymbol{S})^{\frac{1}{c}}).
\end{aligned}
$$

The values of ϕ and c that realize the minimum, can now be used in equation (8) to yield a preliminary estimate $\boldsymbol{\lambda}_0$. The problem with this estimate is, that it does not require the solution to satisfy the link count equation (1), which is a stronger condition than the second moment relation.

The preliminary estimate $\boldsymbol{\lambda}_0$ can be improved by imposing the $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\lambda}$ condition by projecting the result to the surface that satisfies that condition. This yields

$$
\boldsymbol{\lambda} = \boldsymbol{\lambda}_0 + \boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\lambda}_0). \tag{10}
$$

This is in a sense equivalent to giving a next to zero value for the weight coefficient ϵ in Vardi's method, but we do the moment estimation sequentially. This does not yield quite as accurate estimates, but is many time faster, as no numerical methods have to be used.

3.1 Relaxing the exponent parameter

In Cao et al. [2] the EM-algorithm is run after preselecting a convenient value for the exponent parameter c in the power law relation (3). The authors point out that convergence is guaranteed for the algorithm only for integer value c, namely 1 or 2. However, Gunnar et al. [4] in their study of the Global Crossing data find out that the correct values for c in those particular networks are 1.5 and 1.6 for the European and North American core-networks respectively. Thus being limited to integer values in the solution makes sense for only computational reasons. Our method, on the other hand, works for any preselected c. And, in fact, we can relax c to be a free parameter. Although this does mean that we have to let go of the completely explicit solution form. Numerical optimization can be used to estimate the value for c.

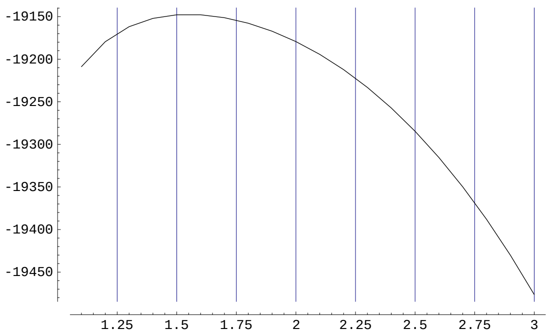

Figure 1: Log-Likelihoods for different parameter c values.

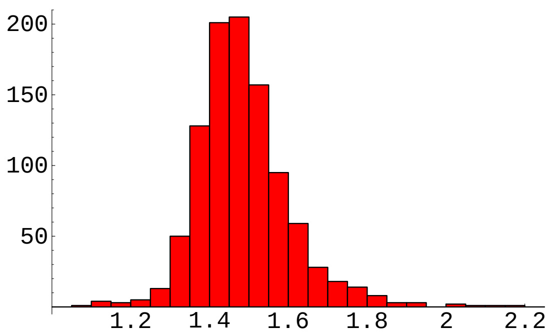

Figure 2: Estimated c-values for synthetic data sets of size 500, generated with setting $c = 1.5$.

For each value of c the traffic matrix can be solved by the quick method as before by treating that c as a fixed value, and then calculate the likelihood for that solution

$$
\begin{aligned}
l(\boldsymbol{\theta}|\boldsymbol{Y}) = -\frac{\tau}{2}\log|\boldsymbol{A\Sigma A}^{\mathrm{T}}| - \qquad (11)\\
-\frac{1}{2}\sum_{t=1}^{\tau}(\boldsymbol{y}_t - \boldsymbol{A\lambda})^{\mathrm{T}}(\boldsymbol{A\Sigma A}^{\mathrm{T}})^{-1}(\boldsymbol{y}_t - \boldsymbol{A\lambda})
\end{aligned}
$$

where $\boldsymbol{\theta} = (\boldsymbol{\lambda}, \phi)$ and $\boldsymbol{\Sigma}$ can be written as function of $\boldsymbol{\lambda}$ according to (3).

In Figure 1 we have the values of the likelihood as a function of c for different c-values between 1 and 3. The figure was generated by a set of synthetic measurements using value $c = 1.5$. Figure 2 shows a histogram of estimated values for parameter c. These are from randomly drawn normal-distribution data, generated with parameter value $c = 1.5$. We see that for this sample size ($n = 500$) the mean is close to 1.5. The standard deviation for the estimates is 0.12.

While this technique is based on the quick method, we can of course use it to solve for c to initialize any algorithm, that relies on the mean-variance relation, but does require a fixed parameter value for c.

4 Constrained minimization

Another approach is to require the condition $\boldsymbol{y} = \boldsymbol{A\lambda}$ to be satisfied from the outset, and try to satisfy the mean-variance relation in the least square sense. In general, this has to be solved numerically. However, in the special case of $c = 1$ an explicit solution can be derived.

This approach is equivalent to Vardi's method, if we set ϵ very small, so that the first moment is the dominant factor in the estimation. With the exception that we treat ϕ as a parameter to be optimized, as in (2) it is fixed as 1 by the Poisson assumption.

We get a constrained minimization problem

$$
\begin{aligned}
\min_{\boldsymbol{\lambda},\phi} \quad & \|\boldsymbol{S}^{(y)} - \boldsymbol{B}\phi\boldsymbol{\lambda}^c\| \qquad (12)\\
\text{such that} \quad & \boldsymbol{y} = \boldsymbol{A\lambda}.
\end{aligned}
$$

Introducing a vector of Lagrange multipliers $\boldsymbol{\alpha}$, the objective function to be minimized can be written as

$$
\begin{aligned}
&f(\boldsymbol{\lambda},\boldsymbol{\alpha},\phi)\\
&= (\boldsymbol{S}^{(y)} - \phi\boldsymbol{B}\boldsymbol{\lambda})^{\mathrm{T}}(\boldsymbol{S}^{(y)} - \phi\boldsymbol{B}\boldsymbol{\lambda}) + 2\boldsymbol{\alpha}^{\mathrm{T}}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\lambda})\\
&= \phi^2\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\lambda} - 2\phi{\boldsymbol{S}^{(y)}}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\lambda} - 2\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{\lambda}\\
&\quad + {\boldsymbol{S}^{(y)}}^{\mathrm{T}}\boldsymbol{S}^{(y)} + 2\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}. \qquad (13)
\end{aligned}
$$

The above expression is quadratic in $\boldsymbol{\lambda}$, and the minimum with respect to $\boldsymbol{\lambda}$ can easily be found,

$$
\boldsymbol{\lambda} = \phi^{-2}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{-1}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{\alpha} + \phi\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}^{(y)}) \qquad (14)
$$

The Lagrange multipliers $\boldsymbol{\alpha}$ are then determined such that the constraints hold:

$$
\boldsymbol{y} = \boldsymbol{A}\phi^{-2}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{-1}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{\alpha} + \phi\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}^{(y)}), \qquad (15)
$$

from which

$$
\begin{aligned}
\boldsymbol{\alpha} &= (\phi^{-2}\boldsymbol{A}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{-1}\boldsymbol{A}^{\mathrm{T}})^{-1}\cdot\\
&\quad \cdot(\boldsymbol{y} - \phi^{-1}\boldsymbol{A}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}^{(y)}). \qquad (16)
\end{aligned}
$$

Minimizing $f(\boldsymbol{\lambda},\boldsymbol{\alpha},\phi)$ with respect to ϕ yields

$$
\phi = (\boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\lambda})^{-1}{\boldsymbol{S}^{(y)}}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{\lambda}. \qquad (17)
$$

Substitution of (16) into (14) gives $\boldsymbol{\lambda}$ as a function of ϕ

$$
\boldsymbol{\lambda} = \boldsymbol{K}\boldsymbol{y} - \phi^{-1}\big(\boldsymbol{K}\boldsymbol{A}(\boldsymbol{B}^T\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{S}^{(y)} + \boldsymbol{B}^T\boldsymbol{S}^{(y)}\big),
$$

where we use the notation

$$
\boldsymbol{K} = (\boldsymbol{B}^T\boldsymbol{B})^{-1}\boldsymbol{A}^T(\boldsymbol{A}(\boldsymbol{B}^T\boldsymbol{B})^{-1}\boldsymbol{A}^T)^{-1}.
$$

Substituting $\boldsymbol{\lambda}$ further in (17) yields an quadratic equation for ϕ, which is easily solvable. This can be then substituted back to (16) and (14) to obtain the solution for $\boldsymbol{\lambda}$.

5 Comparison against MLE method

The accuracy of our quick methods are evaluated by comparing them against Maximum likelihood estimation. In the following subsection we present the Maximum likelihood estimation used. In the subsequent sections the results of accuracy on synthetic data test cases is presented.

5.1 Maximum Likelihood Estimation

We follow the approach of Cao et al. [2] in using the Expectation Maximization (EM) algorithm. For a review see also [7].

The log-likelihood for estimating $\boldsymbol{\lambda}$, the vector containing the means of $\boldsymbol{x}$ is given in equation 11. In Cao et al. c is assumed to be constant and the parameters of the model are thus

$$
\boldsymbol{\theta} = (\phi, \boldsymbol{\lambda})
$$

Now the problem can be solved numerically with the EM-algorithm. The complete data log-likelihood is of the form

$$
l(\boldsymbol{\theta}|\boldsymbol{X}) = -\frac{\tau}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\sum_{t=1}^{\tau}(\boldsymbol{x}_t - \boldsymbol{\lambda})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_t - \boldsymbol{\lambda})
$$

The EM-equation is

$$
\begin{aligned}
&Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(k)}) = \mathrm{E}[l(\boldsymbol{\theta}|\boldsymbol{X})|\boldsymbol{Y},\boldsymbol{\theta}^{(k)}]\\
&= \mathrm{E}[-\frac{\tau}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\sum_{t=1}^{\tau}(\boldsymbol{x}_t - \boldsymbol{\lambda})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_t - \boldsymbol{\lambda})|\boldsymbol{Y},\boldsymbol{\theta}^{(k)}]
\end{aligned}
$$

And since

$$
\begin{aligned}
&\mathrm{E}[(\boldsymbol{x} - \boldsymbol{\lambda})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\lambda})]\\
&= \mathrm{E}[\mathrm{Tr}\{\boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\lambda})(\boldsymbol{x} - \boldsymbol{\lambda})^{\mathrm{T}}\}]\\
&= \mathrm{Tr}\{\boldsymbol{\Sigma}^{-1}\mathrm{E}[(\boldsymbol{x} - \boldsymbol{\lambda})(\boldsymbol{x} - \boldsymbol{\lambda})^{\mathrm{T}}]\}\\
&= \mathrm{Tr}\{\boldsymbol{\Sigma}^{-1}\mathrm{E}[((\boldsymbol{x} - \boldsymbol{m}) + (\boldsymbol{m} - \boldsymbol{\lambda}))((\boldsymbol{x} - \boldsymbol{m}) + (\boldsymbol{m} - \boldsymbol{\lambda}))^{\mathrm{T}}]\}\\
&= \mathrm{Tr}\{\boldsymbol{\Sigma}^{-1}(\boldsymbol{R} + (\boldsymbol{m} - \boldsymbol{\lambda})(\boldsymbol{m} - \boldsymbol{\lambda})^{\mathrm{T}})\}\\
&= \mathrm{Tr}\{\boldsymbol{\Sigma}^{-1}\boldsymbol{R}\} + (\boldsymbol{m} - \boldsymbol{\lambda})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{m} - \boldsymbol{\lambda})
\end{aligned}
$$

we can write

$$
\begin{aligned}
Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(k)}) &= -\frac{\tau}{2}(\log|\boldsymbol{\Sigma}| + \mathrm{Tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{R}^{(k)}))\\
&\quad -\frac{1}{2}\sum_{t=1}^{\tau}(\boldsymbol{m}_t^{(k)} - \boldsymbol{\lambda})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{m}_t^{(k)} - \boldsymbol{\lambda}),
\end{aligned}
$$

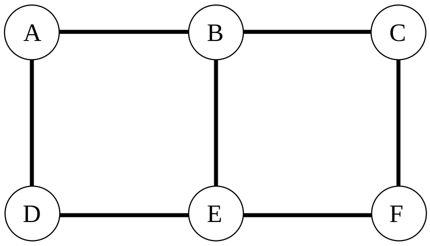

Figure 3: Six node Test topology

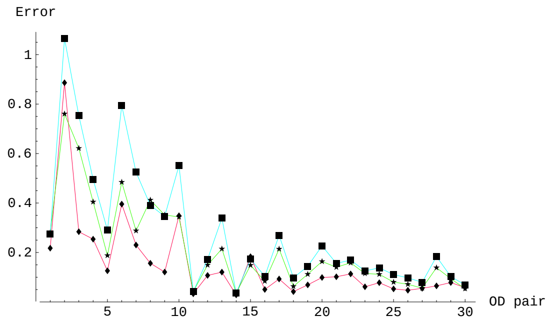

Figure 4: Errors for OD pairs in 6-node topology in ascending order of traffic amount for case $c = 1$. Diamonds denote MLE, stars projection method and rectangles constrained minimization.

where

$$\begin{aligned}
\boldsymbol{m}_t^{(k)} &= \mathrm{E}[\boldsymbol{x}_t|\boldsymbol{y}_t, \boldsymbol{\theta}^{(k)}] \\
&= \boldsymbol{\lambda}^{(k)} + \boldsymbol{\Sigma}^{(k)}\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{\Sigma}^{(k)}\boldsymbol{A}^{\mathrm{T}})^{-1}(\boldsymbol{y}_t - \boldsymbol{A}\boldsymbol{\lambda}) \\
\boldsymbol{R}^{(k)} &= \mathrm{Var}[\boldsymbol{x}_t|\boldsymbol{y}_t, \boldsymbol{\theta}^{(k)}] \\
&= \boldsymbol{\Sigma}^{(k)} - \boldsymbol{\Sigma}^{(k)}\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{\Sigma}^{(k)}\boldsymbol{A}^{\mathrm{T}})^{-1}\boldsymbol{A}\boldsymbol{\Sigma}^{(k)}.
\end{aligned}$$

According to [2], convergence to the maximum likelihood estimate is guaranteed in the special cases of $c = 1$ and $c = 2$.

5.2 Results

In the evaluation we use two topologies. A small six node topology shown in Figure 3 has 14 one way links, two between each node connected in the figure. Hence, there are 30 OD pairs in the network. In the more realistic size fictitious backbone topology OMP-USA, shown in Figure 6, there are 12 nodes, 38 links, and 132 OD pairs. For both topologies, we generate synthetic Gaussian data sets, where the power-law holds. Sample size is set as 500 measurements for each simulation.

5.2.1 A Simple six node topology

In the synthetic OD pair traffic that we use, the traffic amounts vary so that the largest are tenfold larger than the smallest ones.

In Figure 4 the results of the projection method and the constrained minimization, along with the ML estimates are displayed in the case $c = 1$. The OD pairs are arranged into ascending order based on the traffic amount, so that the smaller OD pairs are on the left and the largest on the right. The plot marked with diamonds represents the errors of the MLE method, while the projection method is denoted by the stars and the constrained minimization by the rectangles. We see that, as expected, the ML estimation performs better on average, but not overwhelmingly better. The average errors are 21%, 27% and 15% for the projection method, constrained minimization and MLE respectively. The difference is smaller if we concentrate on the largest OD pairs. For the 15 largest OD pairs the average errors were 11% and 13% for the quick methods and 7% for the MLE.

In the sequel we differ the underlying power-law parameters in our synthetic data. As the projection method seems to be better than the constrained minimization method, even when $c = 1$, which is the required value for the latter, we consider only the projection method in these evaluations.

We generate synthetic set of measurements, with the realistic value $c = 1.5$ as given in [4] for the Global Crossing network. We study the

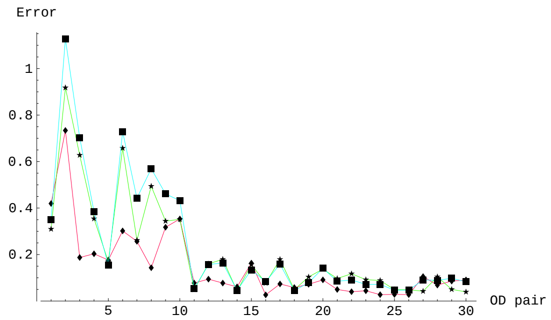

Figure 5: Errors for OD pairs in larger topology in ascending order of traffic amount. Diamonds denote MLE, stars projection method and rectangles projection method with fixed c.

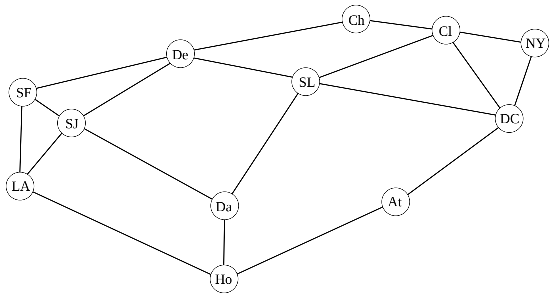

Figure 6: OMP-USA test topology

accuracy of the projection method compared to the MLE, and also the affect of relaxing c.

In Figure 5 we see the errors for MLE method, projection method, where c is fixed to value different than the real one, namely here $c = 2$, and the projection method that includes the estimation of the appropriate c-value. The average error is 15% for the MLE and 24% for the projection method, which is improved to 21% by relaxing the exponent parameter. For the 15 largest of the 30 OD pairs, the errors are 7% for MLE and 9% for projection method, and marginally larger for the projection method with fixed exponent parameter.

5.2.2 OMP-USA

In this example network, the traffic volumes for the OD pairs varies so that the largest are approximately hundred times as large as than the smallest. This creates great difficulties for the projection method regarding the estimation of the smaller OD pairs. The estimates of the projection method for the smallest OD pairs are far off the real traffic amounts. Due to the fact that the estimates for some of the smallest OD pairs have errors of several hundred percent, the mean error is also affected by these, and is 67%, while the median error is 38%. The mean error for the MLE is 9%. However, the most important thing is to estimate the largest OD pairs. If we concentrate only on the largest OD pairs comprising over 80% of total traffic in volume, the projection method is more competitive. The errors are shown in Figure 7. The mean errors are 16% for the projection method and 7% for the MLE.

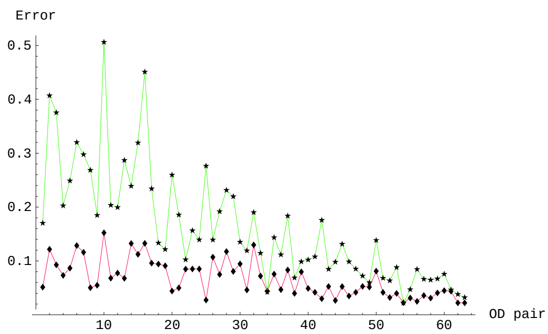

Figure 7: Errors for the largest OD pairs in ascending order of traffic amount

6 Conclusion and Future Work

This paper presented ways to obtain estimate for traffic matrix by explicit calculations utilizing the link count covariance matrix. We illustrated how to obtain the OD pair traffic variance estimates from empirical link count covariance matrix, and developed computationally light weight methods, the projection method and the constrained minimization method, to use the covariance estimate to obtain estimate for the traffic matrix, in a way that would still be consistent with the link counts.

The constrained minimization method was recognized, inn fact, to be a special case of Vardi's method. We give an explicit solution for it in the case $c = 1$ and also obtain an estimate for the second parameter ϕ in the mean-variance relation. For the projection method we have an even simpler and quicker to compute solution. Also in this case we get estimates of the parameters c and ϕ.

We evaluated the accuracy of the methods in a simulation study by comparing them against the maximum likelihood solution by Cao et al., and found that they perform reasonably well, considering they are much quicker and simpler to calculate than the MLE, which requires the use of an iterative numerical method, namely the EM-algorithm. In the worst case, the errors in the estimate of a traffic matrix element for the largest components given by the quick method were three times as large as those by the MLE method, in many cases they difference was smaller. As for the running time, the difference between the MLE method and quick methods was big. With our non-optimized Mathematica code running the MLE method took of the order of tens of minutes, while the quick methods yielded the result in a few seconds.

In this paper all comparisons were done with synthetic data. Evaluation with real data would be very important to assess the true effectiveness of the methods. For now, we have used in our evaluations a sample size of 500, which may be rather large. Accuracy of the estimated covariance matrix with various sample sizes should be studied, as well as the effects the measurement inaccuracies have on the subsequent traffic matrix estimates.

References

[1] Y. Vardi, "Network tomography: estimating source-destination traffic intensities from link data", Journal of the American Statistical Association, vol. 91, pp. 365–377, 1996.

[2] J. Cao, D. Davis, S. V. Wiel, B. Yu, "Time-varying network tomography," Journal of the American Statistical Association, vol. 95, pp. 1063–1075, 2000.

[3] A. Soule, A. Nucci, R. Cruz, E. Leonardi, N. Taft, "How to Identify and Estimate the Largest Traffic Matrix Elements in a Dynamic Environment" SIGMETRICS/PerformanceŠ04, New York, USA, pp. 73–84, 2004.

[4] A. Gunnar, M. Johansson, and T. Telkamp, "Traffic matrix estimation on a large IP backbone – A comparison on real data", IMC'04, Taormina, Italy, 2004.

[5] David G. Luenberger, "Optimization by Vector Space Methods", New York NY, John Wiley & Sons, 1969.

[6] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot, "Traffic matrix estimation: Existing techniques and new directions", ACM SIGCOMM, Pittsburg, USA. August 2002.

[7] S. Vaton, J.S. Bedo, A. Gravey, "Advanced methods for the estimation of the Origin Destination traffic matrix", Revue du 25ème anniversaire du GERAD, 2005.