Network Tomography: Estimating Source-Destination Traffic Intensities From Link Data

Y. VARDI

The problem of estimating the node-to-node traffic intensity from repeated measurements of traffic on the links of a network is formulated and discussed under Poisson assumptions and two types of traffic-routing regimens: deterministic (a fixed known path between each directed pair of nodes) and Markovian (a random path between each directed pair of nodes, determined according to a known Markov chain fixed for that pair). Maximum likelihood estimation and related approximations are discussed, and computational difficulties are pointed out. A detailed methodology is presented for estimates based on the method of moments. The estimates are derived algorithmically, taking advantage of the fact that the first and second moment equations give rise to a linear inverse problem with positivity restrictions that can be approached by an EM algorithm, resulting in a particularly simple solution to a hard problem. A small simulation study is carried out.

KEY WORDS: Communication network; Computer network; EM algorithm; Linear inverse problems; Maximum-likelihood; Moment method; Origin-destination tables; Poisson traffic; Positivity constraints; Trip matrix.

1. INTRODUCTION

We consider the problem of estimating the traffic intensity between all source-destination (directed) pairs (SD pairs heretofore) of nodes of a network from repeated measurements of traffic flow along the directed links of the network. For concreteness and simplicity of presentation, we discuss the problem in terms of "packets" or "messages" transmitted over a communication network although the development is applicable to any network for which our formal assumptions (described later) hold. We assume that the SD pairs transmit messages over a strongly connected directed network (i.e., there exists a direct path between any two nodes). The networks are divided into two categories: *fixed routing* (deterministic) and *random (Markovian) routing* networks. In fixed routing networks, the directed path between any two nodes is fixed and known and remains the same for all messages traveling between these two nodes. In random routing networks, the directed path taken by a message traveling from the source node j_1 to the destination node j_2 is determined according to a fixed known Markov chain specific to the SD pair (j_1, j_2). From a purely technical standpoint, fixed routing is a special case of random routing, much as a sequence of iid random variables is a special case of a Markovian sequence. But there are many advantages to treating the two cases separately, and so, for now, we proceed with the case of fixed routing, postponing the discussion and treatment of random routing until Section 5.

Let c denote the number of SD pairs ($c = n(n-1)$ in a network of n nodes), and let $X_j^{(k)}$ be the number of transmitted messages for SD pair j at measurement period k. We assume that $X_j^{(k)} \sim \text{Poisson}(\lambda_j), j = 1, \ldots, c, k = 1, \ldots, K$, independent. We denote vectors and matrices by boldface and transpose by a prime. $\mathbf{X}^{(k)} = (X_1^{(k)}, \ldots, X_c^{(k)})'$ is the SD transmission vector at period $k, k = 1, \ldots, K$. Let r be the number of directed links in the network, and let $\mathbf{A}$ denote the $r \times c$ routing matrix for a fixed-routing network. $\mathbf{A}$ is a zero-one matrix; its rows corresponds to the directed links, its columns correspond to the SD pairs, and its entries, a_{ij}'s, are "1" or "0" according to whether link i does or does not belong to the directed path of the SD pair j. (For simplicity, we index the SD pairs and the directed links with one dimensional indices, typically j and i. This may occasionally lead to a slight ambiguity as we may refer to the third SD pair (say) as "$j = 3$" and also as $j = (j_1, j_2) = (a, b)$ (say). The alternative, however, would be overly cluttered expressions with too many subscripts and superscripts, throughout the article.) The measured data are the traffic on all the links of the network for K time periods: $\mathbf{Y}^{(k)} = (Y_1^{(k)}, \ldots, Y_r^{(k)})', k = 1, \ldots, K$, where

$$\mathbf{Y}^{(k)} = \mathbf{A}\mathbf{X}^{(k)}, \qquad k = 1, \ldots, K. \tag{1}$$

Our goal is to estimate $\lambda \equiv (\lambda_1, \ldots, \lambda_c)'$ from $\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(k)}$ within the framework of this model. Because of the network interpretation of the model, c $(= n(n-1))$ is typically larger than r $(= O(n))$, and this raises interesting questions regarding identifiability of the parameters and consistency of the estimates. Note that although the model is linear, it is neither a "typical" linear regression nor a random-effects model, because of the zero-one structure of $\mathbf{A}$, the nonnegativity constraints on the parameters, and the Poisson assumption.

Example 1. Consider the network of Figure 1. There are $(4 \times 3 =)12$ SD pairs and 7 directed links. The routing matrix $\mathbf{A}$ is given by

Y. Vardi is Professor of Statistics, Department of Statistics, Rutgers University, Busch Campus, Piscataway, NJ 08855. The problem originated in a consulting project to DARPA (Becker and Wilks 1993) regarding the statistical analysis of computer data networks, and was brought to his attention by D. Pregibon, C. L. Mallows and P. McCullagh contributed useful comments, Jeanne Yue helped with programming, and National Science Foundation provided financial support. The works of Sherali et al., and Bell, were brought to his attention, at the proofreading stage of this article, by R. Becker.

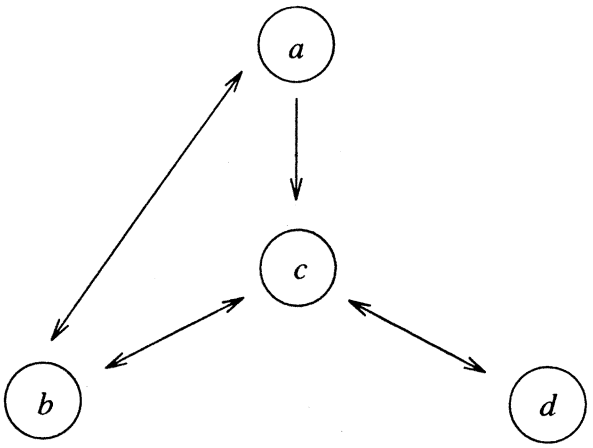

Figure 1. A Sketch of a Four-Node Directed Network. (The actual routs between any two nodes are separately specified by the matrix **A** *of Example 1.1.)*

	1 ab	2 ac	3 ad	4 ba	5 bc	6 bd	7 ca	8 cb	9 cd	10 da	11 db	12 dc
1 $(a \to b)$	1	0	0	0	0	0	0	0	0	0	0	0
2 $(a \to c)$	0	1	1	0	0	1	0	0	0	0	0	0
3 $(b \to a)$	0	0	0	1	0	1	1	0	0	1	0	0
4 $(b \to c)$	0	0	0	0	1	0	0	0	0	0	0	0
5 $(c \to b)$	0	0	0	0	0	0	1	1	0	1	1	0
6 $(c \to d)$	0	0	1	0	0	1	0	0	1	0	0	0
7 $(d \to c)$	0	0	0	0	0	0	0	0	0	1	1	1

Note: The routing is deterministic and is prespecified for each SD pair. (This assumption is relaxed in Sec. 5). For instance, although in Figure 1 both paths, $b \to c \to d$ and $b \to a \to c \to d$, connect the SD pair (bd), the routing matrix specifies that messages from b to d always travel on the path $b \to a \to c \to d$. Traffic from d to b, on the other hand, always travels on the path $d \to c \to b$.

In Section 2 we show that if the columns of A are all nonzero and different from each other, then $\boldsymbol{\lambda}$ is identifiable. In Section 3 we discuss estimation via the methods of maximum likelihood (ML) and moments. ML estimation is numerically complicated for the problem at hand. But the method of moments is simple, because the moment-equations make up a LININPOS (LINear INverse POSitive) problem and so can be solved using a simple EM/ML algorithm (Snyder, Schultz, and O'Sullivan 1992; Vardi and Lee 1993) to produce a method-of-moments estimate. We then simulate a small network to check the performance of such an estimate. The simulation results suggest that the estimate tends toward unbiasedness already for a moderate number of repeated link-traffic measurements. In Section 4 we offer some possible modifications and discuss an estimation strategy when the Poisson assumption can be viewed only as a crude approximation. The proposed method of weighted moment equations then amounts to estimation based on the first moments (which do not depend on the Poisson assumptions) regularized by the second moments (which do depend on the Poisson assumption). Finally, in Section 5 we derive method-of-moments estimates for networks with Markovian routing.

2. IDENTIFIABILITY

The conditions for identifiability of $\boldsymbol{\lambda}$ are surprisingly simple and will typically be satisfied by any real-life routing matrix **A**. This is somewhat unexpected, because in the simplest example, when $r = 1$, the Y's are sums of independent Poisson variables and so the intensity vector $\boldsymbol{\lambda}$ is not identifiable (e.g., $\mathbf{A} = (1, 1, 0, 1)$, so that $Y = X_1 + X_2 + X_4$ and hence only the sum $\lambda_1 + \lambda_2 + \lambda_4$ is identifiable but not the individual components $\lambda_1, \lambda_2, \lambda_3, \lambda_4$.)

The following holds.

Lemma. If the columns of the routing matrix **A** are all distinct, and each column has at least one nonzero entry, then $\boldsymbol{\lambda}$ is identifiable.

Note. A zero column means that the corresponding SD pair is not connected by a path. Likewise, a zero row means that the corresponding link is not a part of the network. Thus any real-life routing matrix **A** would not have either columns or rows of all zeros.

Proof. See the Appendix.

Corollary. If $c > 2^r - 1$, then some of the individual rates $\lambda_1, \ldots, \lambda_c$ cannot be estimated separately.

Proof. $2^r - 1$ is the maximal number of distinct zero-one vectors with at least one "1" in each vector.

3. ESTIMATION (FIXED ROUTING)

We start with a discussion of the maximum likelihood estimator (MLE) and an iterative procedure, based on the EM algorithm. The trouble with this approach, however, is that the iteration formula is very hard to compute. In the course of the discussion, to demonstrate the difficulty of the problem, we give a simple but constructive example where the likelihood equation has a unique solution that differs from the MLE, which is also unique. Alternative methods that are potentially more computationally tractable are discussed next. We then propose and experiment with a particularly simple method motivated by the method of moments. Given that we are estimating parameters within the exponential family, it is reasonable to expect that this method is relatively efficient. Our simulation experiment (discussed in the next section) indicates that the method produces approximately unbiased estimates of $\boldsymbol{\lambda}$ already for moderate values of K.

3.1 Maximum Likelihood Estimation

We start with an interesting "toy" example that demonstrates the difficulty of the problem due to boundary solutions.

Example 3.1: $(r = 2, c = 3, K = 1)$. Let $X_i \sim \text{Poisson}(\lambda_i), i = 1, 2, 3$ independent, and suppose that we are given

$$X_1 + X_2 = 1 \quad \text{and} \quad X_1 + X_3 = 2. \tag{2}$$

Our goal is to derive the ML estimate of $\boldsymbol{\lambda}$. In the notation of the earlier sections, we have

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{Y} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}. \tag{3}$$

The following holds: The unique solution of the likelihood equations is $\boldsymbol{\lambda} = (0, 1, 2)'$. But the unique MLE is $\hat{\boldsymbol{\lambda}} = (1, 0, 1)'$. To verify this, we first derive the likelihood of the data: (2) has only two possible solutions in natural numbers for the $\mathbf{X}$'s:

$$\mathbf{X} = (1, 0, 1)' \quad \text{and} \quad \mathbf{X} = (0, 1, 2)', \tag{4}$$

and so the likelihood of the data is

$$\begin{aligned} L(\boldsymbol{\lambda}) &\equiv P_{\boldsymbol{\lambda}}\{\mathbf{Y} = (1, 2)'\} \\ &= P_{\boldsymbol{\lambda}}\{\mathbf{X} = (1, 0, 1)'\} + P_{\boldsymbol{\lambda}}\{\mathbf{X} = (0, 1, 2)'\} \\ &= (\lambda_1\lambda_3 + \lambda_2\lambda_3^2/2)\exp(-\lambda_1 - \lambda_2 - \lambda_3). \end{aligned} \tag{5}$$

Let $l(\boldsymbol{\lambda}) = \log L(\boldsymbol{\lambda})$. Setting the partial derivatives of l equal to zero, we get the likelihood equations

$$\text{(a) } (2\lambda_1\lambda_3 + \lambda_2\lambda_3^2)^{-1}2\lambda_3 = 1,$$

$$\text{(b) } (2\lambda_1\lambda_3 + \lambda_2\lambda_3^2)^{-1}\lambda_3^2 = 1, \quad \text{and}$$

$$\text{(c) } (2\lambda_1\lambda_3 + \lambda_2\lambda_3^2)^{-1}2(\lambda_1 + \lambda_2\lambda_3) = 1. \tag{6}$$

Equations (6a) and (6b) imply that

$$\lambda_3 = 2.$$

Substituting this back into (6) results in the equations

$$\lambda_1 + \lambda_2 = 1 \quad \text{and} \quad \lambda_1 + 2\lambda_2 = 2, \tag{7}$$

which have a unique solution $\lambda_1 = 0$ and $\lambda_2 = 1$. Thus $\boldsymbol{\lambda} = (0, 1, 2)'$ is the unique solution of the likelihood equations. But the MLE is $\hat{\boldsymbol{\lambda}} = (1, 0, 1)'$. To prove this, first note that the MLE must be a boundary point of the region $\lambda_i \geq 0, i = 1, 2, 3$, because no interior point satisfies the likelihood equations (as we just proved). From (5), it is clear that any candidate for a MLE must have $\lambda_3 > 0$ and one of λ_1 or λ_2 strictly positive (but not both, because otherwise it will be an interior point). If $\lambda_1 = 0$, then we need to maximize $(\lambda_2\lambda_3^2/2)\exp(-\lambda_2 - \lambda_3)$ subject to $\lambda_2, \lambda_3 > 0$, and this maximum is attained at $\lambda_2 = 1$ and $\lambda_3 = 2$. If $\lambda_2 = 0$, then we need to maximize $\lambda_1\lambda_3\exp(-\lambda_1 - \lambda_3)$ subject to $\lambda_1, \lambda_3 > 0$, and this maximum is attained at $\lambda_1 = \lambda_3 = 1$. Thus there are two possible candidates for MLE: $(0, 1, 2)'$ and $(1, 0, 1)'$. Comparing the likelihoods of these candidates, one verifies that $L((0, 1, 2)') < L((1, 0, 1)')$, and so the unique MLE is $\boldsymbol{\lambda} = (1, 0, 1)'$.

3.2 An EM Derivation of the MLE

The likelihood equations ($0 = \partial l/\partial\lambda_j, j = 1, \ldots, c, l \equiv \log L$) in vector notation can be expressed as

$$\mathbf{0} = \frac{1}{K}\sum_{k=1}^{K} E_{\boldsymbol{\lambda}}[\mathbf{X}^{(k)}|\mathbf{Y}^{(k)} = \mathbf{A}\mathbf{X}^{(k)}] - \boldsymbol{\lambda}, \tag{8}$$

and, theoretically, the EM algorithm (Dempster, Laird, and Rubin 1977) can be used to search for a solution. But the implementation is a different story. If we treat the $\mathbf{X}^{(k)}, k = 1, \ldots, K$, as the complete (unobserved) data and $\mathbf{Y}^{(k)}, k = 1, \ldots, K$, as the incomplete (observed) data, then the EM algorithm, in vector notation, is

$$\boldsymbol{\lambda}^{(n+1)} = E[\bar{\mathbf{X}}|\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(K)}, \boldsymbol{\lambda}^{(n)}], \qquad n = 0, 1, \ldots,$$

($\boldsymbol{\lambda}^{(0)} > \mathbf{0}$, arbitrary). Because of linearity of E and independence across k's, (8) becomes

$$\boldsymbol{\lambda}^{(n+1)} = \frac{1}{K}\sum_{k=1}^{K} E[\mathbf{X}^{(k)}|\mathbf{Y}^{(k)}, \boldsymbol{\lambda}^{(n)}], \qquad n = 0, 1, \ldots. \tag{9}$$

The trouble with this iteration formula is that the summands

$$E[\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda}] \tag{10}$$

(superscripts ignored for simplicity) are extremely hard to calculate, as they require first finding all the solutions in natural numbers of $\mathbf{A}\mathbf{X} = \mathbf{Y}$. To demonstrate the details of the procedure, consider again our "toy" example, where $\mathbf{A}$ and $\mathbf{Y}$ are given by (3). From (4), we derive

$$[\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda}] \sim \begin{cases} \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} & \text{with prob. } \lambda_2\lambda_3/(2\lambda_1 + \lambda_2\lambda_3), \\ \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} & \text{with prob. } 2\lambda_1/(2\lambda_1 + \lambda_2\lambda_3), \end{cases} \tag{11}$$

and so (9) leads to the iteration

$$\begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} \leftarrow E[\mathbf{X}|\mathbf{Y}, \boldsymbol{\lambda}] = \begin{pmatrix} 2\lambda_1/(2\lambda_1 + \lambda_2\lambda_3) \\ \lambda_2\lambda_3/(2\lambda_1 + \lambda_2\lambda_3) \\ 1 + \lambda_2\lambda_3/(2\lambda_1 + \lambda_2\lambda_3) \end{pmatrix}. \tag{12}$$

The computational difficulties in replicating this process and in calculating (10) for anything but a "toy problem" are apparent. Unconnected to this computational difficulty, note here that both $(0, 1, 2)'$ and $(1, 0, 1)'$ are stationary points of the iteration (12), and so, depending on the starting point, the algorithm may converge to a non-MLE point.

Another interesting question is whether the log-likelihood is concave, which would have simplified the likelihood optimization problem. After some algebra, it is possible to express the matrix of second derivatives of the log-likelihood function l as

$$\begin{aligned} \mathbf{H} &\equiv \left(\frac{\partial^2 l}{\partial\lambda_i\partial\lambda_j}\right) \\ &= \sum_{k=1}^{K} \operatorname{cov}[\mathbf{X}^{(k)}/\boldsymbol{\lambda}|\mathbf{Y}^{(k)}, \boldsymbol{\lambda}] \\ &\quad - \operatorname{diag}\left(\sum_{k=1}^{K} E[\mathbf{X}^{(k)}/\boldsymbol{\lambda}^2|\mathbf{Y}^{(k)}, \boldsymbol{\lambda}]\right), \end{aligned} \tag{13}$$

where the quotient vectors $\mathbf{X}^{(k)}/\boldsymbol{\lambda}$ and $\mathbf{X}^{(k)}/\boldsymbol{\lambda}^2$ are the vectors of coordinate-wise ratios. It follows that

$$\sum_{i,j} z_i z_j H_{ij} = \sum_{k=1}^{K} \operatorname{var}\left[\sum_{i=1}^{c} \frac{z_i}{\lambda_i} X_i^{(k)} | \mathbf{Y}^{(k)}, \boldsymbol{\lambda}\right] - \sum_{k=1}^{K} E\left[\sum_{i=1}^{c} \frac{z_i^2}{\lambda_i^2} X_i | \mathbf{Y}^{(k)}, \boldsymbol{\lambda}\right]. \quad (14)$$

It can then be shown that this quantity is not necessarily ≤ 0 for all $\boldsymbol{\lambda}$, and so l is not necessarily concave. Note, however, that when the true $\boldsymbol{\lambda}$, say $\boldsymbol{\lambda}^*$, is an interior point, then for large K's, l is concave in the neighborhood of $\boldsymbol{\lambda}^*$. To see this, apply the law of large numbers to (14) to get

$$\frac{1}{K}\sum_{i,j} z_i z_j H_{ij} \to E_{\boldsymbol{\lambda}^*} \operatorname{var}_{\boldsymbol{\lambda}}\left[\sum_{i=1}^{c} \frac{z_i}{\lambda_i} X_i | \mathbf{Y}\right] - E_{\boldsymbol{\lambda}^*} E_{\boldsymbol{\lambda}}\left[\sum_{i=1}^{c} \frac{z_i^2}{\lambda_i^2} X_i | \mathbf{Y}\right], \quad (15)$$

where $\mathbf{Y} = \mathbf{AX}$. Setting $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$ in (15), we can apply the general formula

$$\operatorname{var}[U] = E(\operatorname{var}[U|V]) + \operatorname{var}(E[U|V])$$

to get

$$\begin{aligned}
&(\text{right side of (15) with } \boldsymbol{\lambda} = \boldsymbol{\lambda}^*) \\
&= \operatorname{var}_{\boldsymbol{\lambda}}\left(\sum_{i=1}^{c} \frac{z_i}{\lambda_i} X_i\right) \\
&\quad - \operatorname{var}_{\boldsymbol{\lambda}}\left(E_{\boldsymbol{\lambda}}\left[\sum_{i=1}^{c} \frac{z_i}{\lambda_i} X_i | \mathbf{Y}\right]\right) \\
&\quad - E_{\boldsymbol{\lambda}}\left[\sum_{i=1}^{c} \left(\frac{z_i}{\lambda_i}\right)^2 X_i\right] \\
&= -\operatorname{var}_{\boldsymbol{\lambda}}\left(E_{\boldsymbol{\lambda}}\left[\sum_{i=1}^{c} \frac{z_i}{\lambda_i} X_i | \mathbf{Y}\right]\right) < 0, \qquad (\boldsymbol{\lambda} = \boldsymbol{\lambda}^*),
\end{aligned} \quad (16)$$

where the last equation follows from the equality of the mean and the variance for Poisson random variables and from the independence of the X_i's. Thus if K is large and the algorithm is iterated in the vicinity of the true $\boldsymbol{\lambda}$ (assumed interior), then the algorithm converges to an MLE near the true $\boldsymbol{\lambda}$. In fact, irrespective of the numerical difficulties in calculating the MLE, for large K the standard asymptotic properties of the MLE should hold for the case at hand.

3.3 MLE and Normal Approximations

Two types of normal approximations can be considered "natural" here. The first is a normal approximation of each of the summands in (9), which amounts to approximating (10) using multivariate normal theory. The second is based on the central limit theory (CLT) approximation of $\bar{Y}$ for large K. We start with the first. Assume that

$$\mathbf{X} \sim N_c(\boldsymbol{\lambda}, \boldsymbol{\Lambda}), \qquad \boldsymbol{\Lambda} \equiv \operatorname{diag}(\boldsymbol{\lambda}), \quad (17)$$

and consider the joint distribution of $(\mathbf{X}', \mathbf{Y}')'$ when $\mathbf{Y} = \mathbf{AX}$. Then

$$\begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix} \sim N_{c+r}\left\{\begin{pmatrix} \boldsymbol{\lambda} \\ \mathbf{A}\boldsymbol{\lambda} \end{pmatrix}, \begin{pmatrix} \boldsymbol{\lambda}, & \boldsymbol{\Lambda}\mathbf{A}' \\ \mathbf{A}\boldsymbol{\Lambda}, & \mathbf{A}\boldsymbol{\Lambda}\mathbf{A}' \end{pmatrix}\right\}, \quad (18)$$

and

$$\mathbf{X}|\mathbf{Y} \sim N_c\{\boldsymbol{\lambda} + \boldsymbol{\Lambda}\mathbf{A}'(\mathbf{A}\boldsymbol{\Lambda}\mathbf{A})^{-1}(\mathbf{Y} - \mathbf{A}\boldsymbol{\lambda}), \boldsymbol{\Lambda} - \boldsymbol{\Lambda}\mathbf{A}'(\mathbf{A}\boldsymbol{\Lambda}\mathbf{A}')^{-1}\mathbf{A}\boldsymbol{\Lambda}\}. \quad (19)$$

In particular, (10) is approximated as $\boldsymbol{\lambda} + \boldsymbol{\Lambda}\mathbf{A}'(\mathbf{A}\boldsymbol{\Lambda}\mathbf{A}^{-1})(\mathbf{Y} - \mathbf{A}\boldsymbol{\lambda})$. Applying this to (9) results in the iteration formula

$$\boldsymbol{\lambda}^{(n+1)} = \frac{1}{K}\sum_{k=1}^{K} [\boldsymbol{\lambda}^{(n)} + \boldsymbol{\Lambda}^{(n)}\mathbf{A}'(\mathbf{A}\boldsymbol{\Lambda}^{(n)}\mathbf{A}')^{-1} \times (\mathbf{Y}^{(k)} - \mathbf{A}\boldsymbol{\lambda}^{(n)})]. \quad (20)$$

We have not experimented with this method, but we suspect that it may perform badly for some cases because the normal approximations (17), on which the iteration is based, is a poor approximation unless we know a priori that *all* of the λ_i's are large. Other potential difficulties with this approach are the cumulative approximation error in each step of the iteration (because we apply the normal approximation to each term in (9) separately), potential numerical instability from the matrix inversion in each iteration, and the possibility of converging to negative values of λ_i's.

In the second normal approach, we approximate the distribution of $\bar{\mathbf{Y}}$ when K is large with a normal distribution:

$$\bar{\mathbf{Y}} = \frac{1}{K}\sum_{k=1}^{K} \mathbf{Y}^{(k)} \overset{\text{dist.}}{\to} N_r(\mathbf{A}\boldsymbol{\lambda}, K^{-1}\mathbf{A}\boldsymbol{\Lambda}\mathbf{A}'), \qquad \boldsymbol{\Lambda} = \operatorname{diag}(\boldsymbol{\lambda}). \quad (21)$$

Treating the right side of (21) as the exact distribution of $\bar{\mathbf{Y}}$, the log-likelihood of $\bar{\mathbf{Y}}$ (modulo multiplicative and additive constants) is

$$l(\boldsymbol{\lambda}) = -\log|\mathbf{A}\boldsymbol{\Lambda}\mathbf{A}'| - K(\bar{\mathbf{Y}} - \mathbf{A}\boldsymbol{\lambda})'(\mathbf{A}\boldsymbol{\Lambda}\mathbf{A}')^{-1}(\bar{\mathbf{Y}} - \mathbf{A}\boldsymbol{\lambda}). \quad (22)$$

A MLE based on this approximation would seek to maximize $l(\boldsymbol{\lambda})$ of (22) subject to the constraints $\lambda_i \geq 0, i = 1, \ldots, c$. When K is large, the second term is the dominant term in (22), and this suggests $\operatorname{argmin}_{\lambda \geq 0}\mathbf{K}(\bar{\mathbf{Y}} - \mathbf{A}\boldsymbol{\lambda})'(\mathbf{A}\boldsymbol{\Lambda}\mathbf{A})^{-1}(\bar{\mathbf{Y}} - \mathbf{A}\boldsymbol{\lambda})$ as a reasonable large-sample substitute for the MLE. Note that this is a weighted least square with positivity constraints and with weights depending on $\boldsymbol{\lambda}$, which can be estimated by the sample covariance matrix of the Y's.

3.4 Estimation Based on Sample Moments

3.4.1 First and Second Moment Equations. The approximate normal distribution of $\bar{\mathbf{Y}}$ is completely determined by the mean vector, $\mathbf{A}\boldsymbol{\lambda}$, and covariance matrix, $\mathbf{A}\boldsymbol{\Lambda}\mathbf{A}'$, of $\mathbf{Y}$. Thus we can equate the sample's first and second moments to their theoretical values to obtain a *linear* (in $\boldsymbol{\lambda}$) *system of estimating equations:*

$$\text{(a)}\quad \hat{E}(Y_i) = \bar{Y}_i = \sum_{l=1}^{c} a_{il}\lambda_l, \qquad i = 1, \ldots, r$$

and

$$\text{(b)}\quad \widehat{\text{cov}}(Y_i, Y_{i'}) = \frac{1}{K}\sum_k Y_i^{(k)} Y_{i'}^{(k)} - \bar{Y}_i\bar{Y}_{i'}$$

$$= \sum_{l=1}^{c} a_{il}a_{i'l}\lambda_l, \qquad 1 \le i \le i' \le r. \quad (23)$$

These are $r(r+3)/2$ linear equations that can be written, in vector notation, as

$$\begin{pmatrix} \bar{\mathbf{Y}} \\ \mathbf{S} \end{pmatrix} = \begin{pmatrix} \mathbf{A} \\ \mathbf{B} \end{pmatrix}\boldsymbol{\lambda}, \quad (24)$$

where $\mathbf{S}$ is the sample covariance matrix stretched out as a vector of length $r(r+1)/2$ (which, for simplicity of presentation here, we index by a double subscript (i, i') $1 \le i \le i' \le r$, ordered lexicographically, say),

$$S_{ii'} = \frac{1}{K}\sum_k Y_i^{(k)} Y_{i'}^{(k)} - \bar{Y}_i\bar{Y}_{i'},$$

and $\mathbf{B}$ is an $(r(r+1)/2) \times c$ matrix (rows indexed by $(i, i'), 1 \le i \le i' \le r$, to match the indexing of $\mathbf{S}$) with the (i, i')th row of $\mathbf{B}$ being the element-wise product of row i and row i' of the matrix $\mathbf{A}$.

Example 3.2. Suppose that

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Then

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix}, \qquad \mathbf{S} = \begin{pmatrix} S_{11} \\ S_{12} \\ S_{13} \\ S_{22} \\ S_{23} \\ S_{33} \end{pmatrix}.$$

3.4.2 A Closer Look at the Moment Equations (24). First we recall that because the X_i's are independent and the a_{ij}'s are zeros or 1s, the marginal distributions of the Y_i's are also Poisson and so

$$EY_i = \text{var } Y_i = \sum a_{ij}\lambda_j, \qquad i = 1, \ldots, r. \quad (25)$$

On the other hand, sampling variability will typically result in $\bar{Y}_i \ne S_{ii}, i = 1, \ldots, r$, and so (24) will be an algebraically inconsistent system of equations (with probability $\simeq 1$). Furthermore, due to sampling variability, we can also expect to observe some negative sample covariance values, $S_{ii'}$'s, whereas the theoretical covariance values, $\sum_l a_{il}a_{i'l}\lambda_l$, are always nonnegative. Finally, when $\sum_l a_{il}a_{i'l} = 0$ and $S_{ii'} \ne 0$, we get a degenerate false identity $S_{ii'} = 0$, which, as before, is attributable to sampling variability and should be deleted. These algebraically inconsistent equations are to be expected, and discussion on how to handle them follows the simulation experiment discussed later. For now, suppose that all the constants on the left side of (24) are strictly positive (i.e., $\bar{Y}_i, S_{ii'} > 0$) and that $\mathbf{B}$ has no rows of zeros. (Clearly, $\mathbf{A}$ has no such rows.) Then, because all of the entries of $\mathbf{A}$ and $\mathbf{B}$ are nonnegative and because $(\bar{\mathbf{Y}}', \mathbf{S}')' > \mathbf{0}$ and $\boldsymbol{\lambda}$ is constrained to be $\ge \mathbf{0}$, (24) is of the general form of a LININPOS problem, and we propose using an EM algorithm to "solve" it. Formally, this solution is the minimizer of the Kullback–Leibler-directed distance between the observed sample moments and their theoretical values (modulo a certain natural scaling), and it can also be viewed as an MLE of the algebraic solution under a multinomial model. (Details have been given in Snyder et al. 1992 and in Vardi and Lee 1993.)

3.4.3 On the Relation to Maximum Likelihood Estimation. Left-multiplying the likelihood equation (8) by $\mathbf{A}$, we see that any stationary point of the log-likelihood function, l, also satisfies the first-moment equations $\bar{\mathbf{Y}} = \mathbf{A}\boldsymbol{\lambda}$. The reverse implication is not true, but a partial converse holds. Specifically, if the columns of $\mathbf{A}$ are linearly independent (which is not the case in our typical application where $c > r$), and $\boldsymbol{\lambda}^*$ solves the first-moment equations, $\bar{\mathbf{Y}} = \mathbf{A}\boldsymbol{\lambda}^*$, then $\boldsymbol{\lambda}^*$ solves (8) and hence is a stationary point of l. To prove this, denote the right side of (8) by $\boldsymbol{\eta}(\boldsymbol{\lambda})$ and note that $\bar{\mathbf{Y}} = \mathbf{A}\boldsymbol{\lambda}^*$ implies $\mathbf{A}\boldsymbol{\eta}(\boldsymbol{\lambda}^*) = \mathbf{0}$, which implies $\boldsymbol{\eta}(\boldsymbol{\lambda}^*) = \mathbf{0}$ for $\mathbf{A}$ with linearly independent columns, so that $\boldsymbol{\lambda}^*$ is a solution of (8).

It is important to note, however, that because of the nonnegativity constraints on $\boldsymbol{\lambda}$, the MLE may be a boundary point and so is not necessarily a stationary point of l, as demonstrated in Example 3.1.

3.4.4 Details of the Method: Implementation for Matrices in Block Form. The canonical form of the EM iteration for solving the LININPOS problem $\bar{\mathbf{Y}} = \mathbf{A}\boldsymbol{\lambda}$ is

$$\lambda_j \leftarrow \frac{\lambda_j}{\sum_{i=1}^{r} a_{ij}} \sum_{i=1}^{r} \frac{a_{ij}\bar{Y}_i}{\sum_{k=1}^{c} a_{ik}\lambda_k}, \qquad j = 1, \ldots, c. \quad (26)$$

If the linear system is given in a block form as

$$\begin{pmatrix} \bar{\mathbf{Y}} \\ \mathbf{S} \end{pmatrix} = \begin{pmatrix} \mathbf{A} \\ \mathbf{B} \end{pmatrix}\boldsymbol{\lambda}, \quad (27)$$

where $\mathbf{A}$ is $r \times c$, $\bar{\mathbf{Y}}$ is $r \times 1$, and $\mathbf{S}$ is $m \times 1$ (indexed as r

$+ 1, \ldots, r + m$) and $\mathbf{B}$ is $m \times c$ (rows indexed as $r + 1, \ldots, r + m$), then (26) becomes the following iteration:

$$\lambda_j \leftarrow \frac{\lambda_j}{\sum_{i=1}^{r} a_{ij} + \sum_{i=r+1}^{r+m} b_{ij}} \times \left\{ \sum_{i=1}^{r} \frac{a_{ij}\bar{Y}_i}{\sum_{k=1}^{c} a_{ik}\lambda_k} + \sum_{i=r+1}^{r+m} \frac{b_{ij}S_i}{\sum_{k=1}^{c} b_{ik}\lambda_k} \right\}, \quad j = 1, \ldots, c. \tag{28}$$

A similar block decomposition was first utilized by Iusem and Svaiter (1994). Equation (28) can now be written as

$$\lambda_j \leftarrow \frac{a_{\cdot j}}{a_{\cdot j} + b_{\cdot j}} \hat{\lambda}_j(\mathbf{A}, \mathbf{Y}, \boldsymbol{\lambda}) + \frac{b_{\cdot j}}{a_{\cdot j} + b_{\cdot j}} \hat{\lambda}_j(\mathbf{B}, \mathbf{S}, \boldsymbol{\lambda}), \tag{29}$$

with "$\cdot$" denoting summation over the subscript that it replaces (e.g., $a_{\cdot j} = \sum_i a_{ij}$). Here $\hat{\lambda}_j$ is defined by the canonical EM iteration for LININPOS problems (as noted in Vardi and Lee 1993),

$$\hat{\lambda}_j(\mathbf{A}, \mathbf{Y}, \boldsymbol{\lambda}) \equiv \frac{\lambda_j}{a_{\cdot j}} \sum_i \frac{a_{ij}Y_i}{\sum_k a_{ik}\lambda_k}, \qquad j = 1, \ldots, c, \tag{30}$$

for any matrix $\mathbf{A}$ and vectors $\mathbf{Y}$ and $\boldsymbol{\lambda}$, with their respective positivity constraints.

3.4.5 A Simulation Experiment. To simulate the network described in Example 1.1, we took $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_{12}) = (1, 2, \ldots, 12)$ to be the "daily" transmission rate between all 12 pairs and generated daily data on the seven links for 100 days, $\mathbf{Y}^{(1)}(\equiv (Y_1^{(1)}, \ldots, Y_7^{(1)})'), \ldots, \mathbf{Y}^{(100)}$. We then calculated the sample means, $\bar{Y}$, and covariances, $\mathbf{S}$, and applied the algorithm (29). This led to an estimate $\hat{\boldsymbol{\lambda}} = (\hat{\lambda}_1, \hat{\lambda}_2, \ldots, \hat{\lambda}_{12})'$. We repeated this whole process 50 times to produce 50 such estimates, $\hat{\boldsymbol{\lambda}}^{(1)}, \ldots, \hat{\boldsymbol{\lambda}}^{(50)}$. The mean vector and covariance matrix based on these 50 estimates were $\bar{\hat{\boldsymbol{\lambda}}} = (1.01, 2.37, 2.68, 4.72, 5.06, 5.79, 6.84, 7.92, 9.25, 9.87, 11.33, 12.14)$ and

$$\widehat{\mathrm{cov}}(\hat{\boldsymbol{\lambda}}) = \begin{bmatrix} .01 & -.01 & -.01 & -.10 & -0 & .03 & -.01 & -.09 & -.01 & -.07 & .08 & -.02 \\ & 1.41 & -.62 & .44 & .02 & -1.22 & -.40 & .15 & .89 & -.10, & -.81 & -.07 \\ & & 3.00 & .88 & -.27 & -1.48 & -.18 & -.04 & -.51 & .10 & .69 & -.62 \\ & & & 8.62 & .15 & -1.55 & -2.15 & 6.56 & .46 & -7.05 & -1.88 & 3.87 \\ & & & & .22 & .20 & -.24 & .33 & .10 & .08 & -.27 & .05 \\ & & * * \text{reflect} * * & & & 3.78 & -.51 & -.27 & -.97 & -.01 & .67 & .55 \\ & & & & & & 6.96 & -5.03 & .46 & -2.84 & 2.46 & -1.15 \\ & & & & & & & 10.79 & -.13 & -3.48 & -7.38 & 5.76 \\ & & & & & & & & 1.78 & .24 & -.87 & .07 \\ & & & & & & & & & 12.69 & -1.60 & -4.52 \\ & & & & & & & & & & 11.50 & -5.93 \\ & & & & & & & & & & & 8.54 \end{bmatrix}.$$

The simulation results suggest that the method leads to unbiased estimation of $\boldsymbol{\lambda}$ and that the variance of the estimated λ_j increases monotonically with λ_j.

3.4.6 On Algebraic Inconsistency and Bad Equations. Algebraic inconsistency of (24) of the type described in connection with Equation (25) is not a real problem. In fact, this inconsistency often implies that the ML/EM solution of the LININPOS problem (24) has a unique solution. (This follows from Byrne 1993 and from Roeder, Devlin, and Lindsay 1989.) Intuitively, the point is that both equations $\bar{Y}_i = EY_i$ and $S_{ii} = EY_i$ are believable and valid as moment equations under the Poisson model, and so we let the "geometry of Kullback–Leibler distances" determine the best compromise between all these inconsistent equations. But the situation is different when we have infeasible equations (e.g., when $S_{ii'} < 0$ but $\sum_l a_{il}a_{i'l} > 0$) or false identities (e.g., when $S_{ii'} \neq 0$ but $\sum_l a_{il}a_{i'l} = 0$). These are known to be impossible moment equations under the Poisson model and should be given no weight by comparison to feasible moment equations. But assigning no weight to these equations amounts to simply deleting them. Because there is no shortage of moment equations, deleting the bad ones seems a reasonable strategy. It is also related to our later discussion of reducing weights to some moment equations.

4. REGULARIZATION: WEIGHTED MOMENT-EQUATIONS

The first-moment equations $\bar{\mathbf{Y}} = \mathbf{A}\boldsymbol{\lambda}$ follow from the linearity of the system (1) and is independent of the Poisson assumption. The second-moment equations depend strongly on the Poisson model, and so in applications where this model is only a crude approximation, one may want to give these equations smaller weight relative to the first-moment equations. This is achieved by replacing $\mathbf{S}$ and $\mathbf{B}$ in (27) with $\varepsilon\mathbf{S}$ and $\varepsilon\mathbf{B}$, for $0 < \varepsilon < 1$. This in turn changes (29) into

$$\lambda_j \leftarrow \frac{a_{\cdot j}}{a_{\cdot j} + \varepsilon b_{\cdot j}} \hat{\lambda}_j(\mathbf{A}, \mathbf{Y}, \boldsymbol{\lambda}) + \frac{\varepsilon b_{\cdot j}}{a_{\cdot j} + \varepsilon b_{\cdot j}} \hat{\lambda}_j(\mathbf{B}, \mathbf{S}, \boldsymbol{\lambda}), \quad 0 < \varepsilon < 1. \tag{31}$$

Note that when $\varepsilon \to 0$, the solution is based entirely on the first moments, and as such it is independent of any proba-

bilistic assumption (but most likely is not a good estimate of $\boldsymbol{\lambda}$ because of shortage of estimating equations). On the other hand, when $\varepsilon \to 1$, the solution is based on the first- and second-moment equations for the Poisson model. Thus ε can be interpreted as a regularization parameter, when the regularization in this case attempts to pull the model toward a Poisson model. This approach, in a different context and with a goal similar in spirit to Silverman, Jones, Wilson, and Nychka's (1990) EMS and Green's (1990a,b) one-step-late EM, was suggested by Iusem and Svaiter (1994), who discussed convergence properties of (31).

5. RANDOM (MARKOVIAN) ROUTING

Here we consider the same problem under the same model assumptions as in Sections 1–4, except that the assumption of fixed routing is replaced by Markovian routing. For specificity and ease of description, we use the term *SD address* (and sometimes just *address*) to mean the combined sender and receiver addresses, so that a packet traveling from source j_1 to destination j_2 is said to have an SD address $j\,(=(j_1,j_2))$. (This is like a standard letter, which usually includes both sender and receiver addresses.) As before, the routing regimen is specified by an $r \times c$ matrix $\mathbf{A}$, but now the entries are conditional probabilities:

a_{ij} is the conditional probability that a packet with SD-address j passes through link $i = (i_1, i_2)$, conditioned on the packet being at node i_1. (It is assumed that this probability is independent of how the packet arrived at i_1 but depends only on its SD address j.)

For any fixed SD pair, the Markovian nature of the routing is a consequence of the assumption that the routing of any packet at each particular node does not depend on the path through which it arrived at the node, but depends only on it being there and on its SD address. Thus each column of the matrix $\mathbf{A}$ corresponds to a directed network with c nodes and r links, with probabilities attached to each link. Typically, if the probability of passing through a given link is zero, then we do not include the link in the diagram of the network. We further assume that the routing matrix $\mathbf{A}$ is such that a packet cannot revisit a node, and that for each SD address $j = (j_1, j_2)$, the corresponding column of $\mathbf{A}$ specifies a network of paths all leading from j_1 (source) to j_2 (destination).

Example 5.1. Consider a network of 4 nodes (and hence 12 SD addresses) and 9 possible links given by the following matrix $\mathbf{A}$ (blank entries are equal to zero):

$\mathbf{A} =$

	1 *ab*	2 *ac*	3 *ad*	4 *ba*	5 *bc*	6 *bd*	7 *ca*	8 *cb*	9 *cd*	10 *da*	11 *db*	12 *dc*
1 $a \to b$	.8	.2	.2									
2 $a \to c$	.2	.8	.8		1	1						
3 $b \to a$				1	.2	.1	1			1		
4 $b \to c$		.8			.8	.1						1
5 $b \to d$		.2	1			.8			1			
6 $c \to b$	.8		.2				.8	.8	.2	1	1	
7 $c \to d$	.2		.8			1	.2	.2	.8			
8 $b \to d$	1						1	1		.8	.8	.2
9 $d \to c$		1								.2	.2	.8

The 12 columns of $\mathbf{A}$ correspond to the 12 Markovian networks depicts in Figure 2.

As in the earlier sections, we observe the traffic on the links and are interested in estimating the transmission rates for all SD addresses, based on these data and the matrix $\mathbf{A}$. The problem at hand resembles the estimation problem in PET (Shepp and Vardi 1982; Vardi, Shepp, and Kaufman 1985), where the link data (also Poisson, as sums of "thinned" Poisson variables) are the equivalent of the "detector tubes" data in the PET problem. But there is a crucial difference that makes the current problem much harder. Although the detector tubes data in PET are mutually independent, the link data are highly dependent. In fact, the current problem is similar to a PET problem in which the detector tubes overlap. Thus ML estimation appears to be very complicated. Because of this, we focus again on the method of moments.

Despite the significantly more complicated nature of the random routing regimen, it turns out that the first and second moments of the observed link data are linear in $\boldsymbol{\lambda}$, and, as in the case of fixed routing, the moment equations constitute a LININPOS problem that can be "solved" using an EM/ML algorithm. The hard part is then the calculation of EY_i and $\operatorname{cov}(Y_i, Y_{i'}), 1 \le i, i' \le r$, which are needed to set up the moment equations. These are derived later.

5.1 Estimation: First and Second Moments

5.1.1 Network Thinning of Poisson Variables. In general, "thinning" of Poisson variables is a reference to the following general property: If $X \sim \text{Poisson}(\lambda)$ and $[Y_1, \ldots, Y_k | X] \sim \text{multinomial}(X; \pi_1, \ldots, \pi_k)$, then $Y_1, \ldots, Y_k$ are mutually independent and $Y_i \sim \text{Poisson}(\lambda\pi_i)$, $i = 1, \ldots, k$. (The index k here should not be confused with the index k for the measurement period in earlier sections, as the derivation of the moments in this section requires consideration of only a single measurement period.)

We can use this property as follows: Let X_j be the number of packets with SD address j; $X_j \sim \text{Poisson}(\lambda_j)$. Let Y_i^j be the number of packets with address j passing through link $i\ (=(i_1,i_2))$. It follows from the "thinning property" that

$$Y_i^j \sim \text{Poisson}(\lambda_j P_i^j), \tag{32}$$

where

$$P_i^j = \text{probability that a packet with address } j \text{ passes through link } i.$$

Furthermore, because of the independence of the X_j's, the Y_i^j's are independent across j for fixed i, and so

$$Y_i = \sum_j Y_i^j \sim \text{Poisson}\left(\sum_j \lambda_j P_i^j\right), \qquad i = 1, \ldots, r. \tag{33}$$

Next, we express P_i^j in terms of the routing matrix $\mathbf{A}$. We define a chain of connected links as a *path,* and we say that a path is *possible* if it has positive probability. Specifically, a path $(i_1, \ldots, i_k)$ is *possible for* SD *address* j if it

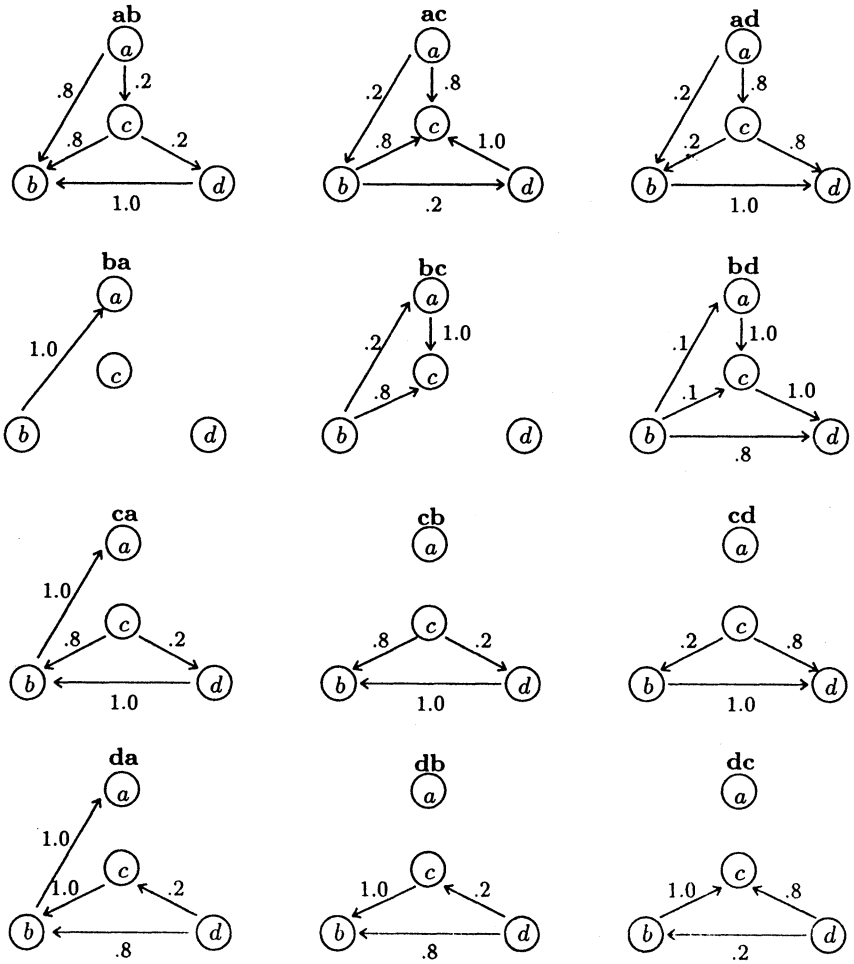

Figure 2. The Four-Node Network of Example 5.1 With Its Markovian Routing Scheme (as Specified by the Matrix **A***) Gives Rise to 12 Markov Networks Corresponding to the 12 Possible SD Addresses, as Depicted Here.*

is possible to travel in sequence through $i_1, i_2, \ldots, i_k$ under the network probabilities for SD address j. We say that a path is *complete for SD address* j if it is possible and leads from the source (j_1) to the destination (j_2). Let B_i^j denote the set of all complete paths for address j that pass through link i. Then the chain rule for probabilities $(P(D_3D_2D_1) = P(D_3|D_2D_1)P(D_2|D_1)P(D_1))$ and the Markovian nature of the routing imply that the probability for a packet to travel on a possible path $(i_1, \ldots, i_k)$ in B_i^j is the product $a_{i_1 j} \cdots a_{i_k j}$. Hence

$$P_i^j = \sum_{(i_1^m, \ldots, i_{k_m}^m) \in B_i^j} a_{i_1^m j} \cdots a_{k_{k_m}^m j}, \tag{34}$$

$(i = 1, \ldots, r, j = 1, \ldots, c)$.

Example 5.1 (continued). We get the following P_i^j's, $i = 1, \ldots, 9$ and $j = 1, \ldots, 12$ (blank entries are equal to zero):

$\mathbf{P} =$

	1 ab	2 ac	3 ad	4 ba	5 bc	6 bd	7 ca	8 cb	9 cd	10 da	11 db	12 dc
1 $a \to b$	.8	.2	.2									
2 $a \to c$	.2	.8	.8		.2	.1						
3 $b \to a$				1	.2	.1	1			1		
4 $b \to c$		.16			.8	.1						.2
5 $b \to d$		.04	.36			.8			.2			
6 $c \to b$	.16		.16				.8	.8	.2	.2	.2	
7 $c \to d$	.04		.64			.2	.2	.2	.8			
8 $d \to b$	.04						.2	.2		.8	.2	.2
9 $d \to c$		.04								.2	.2	.8

For example,

$$P_5^3 = P_{b\to d}^{ad} = .2 \times 1.0 + .8 \times .2 \times 1.0 = .36 \; (= a_{13} \times a_{53} + a_{23} \times a_{63} \times a_{53}).$$

5.1.2 Calculating the First and Second Moments of $\mathbf{Y}$. As it stands, formula (34) is somewhat hard to calculate because of the complexity of constructing the set of paths B_i^j. Here we give a relatively simple method for deriving $\mathbf{P}$ from $\mathbf{A}$, using Markov chains techniques, and demonstrate it on the foregoing example. Expressions (33) and (34) give us the necessary ingredients for the moment equations based on the mean and variance, because

$$E\mathbf{Y} = \mathbf{P}\boldsymbol{\lambda}, \qquad \text{Var } \mathbf{Y} = \mathbf{P}\boldsymbol{\lambda}, \tag{35}$$

where var $\mathbf{Y} = (\text{var}Y_1, \ldots, \text{var}Y_r)'$ and $\mathbf{P}$ is given in (34). To construct additional second-moment equations corresponding to $\text{cov}(Y_i, Y_{i'})$, we need the following. Let

$$Y_{ii'}^j = \text{the number of packets with SD address } j \text{ passing through both link } i \text{ and link } i'.$$

Then, again from the "thinning property" of Poisson variables, we have

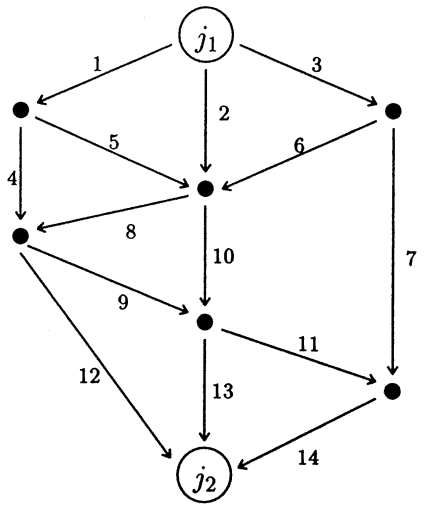

Figure 3. A Diagram of a Network Corresponding to SD Address j (= (j_1, j_2) With 14 Links Labeled 1, . . . , 14.

$$Y^j_{ii'} \sim \text{Poisson}(\lambda_j P^j_{ii'}), \qquad 1 \le i, i' \le r, \qquad j = 1, \ldots, c, \tag{36}$$

where

$P^j_{ii'}$ = probability that a packet with SD address j passes through both link i and link i'.

Now we can decompose Y^j_i and $Y^j_{i'}$ as follows: $Y^j_i = Y^j_{ii'} + U^j_{ii'}$ and $Y^j_{i'} = Y^j_{ii'} + U^j_{i'i}$, where $U^j_{ii'}$ is the number of packets with address j passing through link i but not i', and $U^j_{i'i}$ is the number of packets with address j passing through link i' but not i. Because of the thinning property, $U^j_{ii'}, U^j_{i'i}$, and $Y^j_{ii'}$ are statistically independent, and thus we get

$$\begin{aligned}\text{cov}(Y_i, Y_{i'}) &= \text{cov}\left(\sum_j Y^j_i, \sum_k Y^k_{i'}\right) \overset{(1)}{=} \sum_j \text{cov}(Y^j_i, Y^j_{i'}) \\ &= \sum_j \text{cov}(Y^j_{ii'} + U^j_{ii'}, Y^j_{ii'} + U^j_{i'i}) \\ &\overset{(2)}{=} \sum_j \text{cov}(Y^j_{ii'}, Y^j_{ii'}) \\ &= \sum_j \text{var}(Y^j_{ii'}, Y^j_{ii'}) \overset{(3)}{=} \sum_j \lambda_j P^j_{ii'}. \end{aligned} \tag{37}$$

(Here (1) follows from the independence of Y^j_i's across j; (2) follows from the independence of $Y^j_{ii'}, U^j_{ii'}$, and $U^j_{i'i}$, as explained earlier; and (3) follows from (36).)

As in the derivation of (34), the chain rule for probabilities and the Markovian nature of the routing lead to

$$P^j_{ii'} = \sum_{(i^m_1, \ldots, i^m_{k_m}) \in B^j_{ii'}} a_{i^m_1 j} \cdots a_{i^m_{k_m} j}, \qquad 1 \le i, i' \le r, \quad j = 1, \ldots, c, \tag{38}$$

where

$B^j_{ii'}$ = set of all complete paths for address j, which pass through both link i and link i'.

Example 5.2. Consider the network depicted in Figure 3. We have

$$\begin{aligned} B^j_{9,8} = \{&(1,5,8,9,13), (2,8,9,13), (3,6,8,9,13), \\ &(1,5,8,9,11,14), (2,8,9,11,14), (3,6,8,9,11,14)\}, \end{aligned}$$

$$B^j_{3,5} = \phi, B^j_{5,6} = \phi,$$

$$\begin{aligned} B^j_{8,8} = B^j_8 = \{&(1,5,8,9,13), (1,5,8,9,11,14), (1,5,8,12), \\ &(2,8,9,13), (2,8,9,11,14), (2,8,12), (3,6,8,9,13), \\ &(3,6,8,9,11,14), (3,6,8,12)\}, \end{aligned}$$

and

$$B^j_{6,10} = \{(3,6,10,13), (3,6,10,11,14)\}.$$

Note: Because the sum of the probabilities over all (partial) paths originating at any given node and terminating at the destination is 1, we can modify the definition of $B^j_{ii'}$ to disregard the tails of all paths in $B^j_{ii'}$ beyond the point at which they pass both link i and link i'. (This is so because the absence of loops implies that *all* such paths either pass first through i and then through i', or the other way around for all such paths. Otherwise, there would be a loop between i and i', contradicting our assumption. This permits modifying the definition of $B^j_{ii'}$ to be *the set of all possible paths for address j starting at j_1 and passing through both link i and i' and terminating at either link i or link i' (whichever is furthest from j_1).*) An analogous statement is of course true for B^j_i. With this modified definition, we get in Example 5.2

$$B^j_{9,8} = \{(1,5,8,9), (2,8,9), (3,6,8,9)\},$$

$$B^j_{3,5} = \phi, \qquad B^j_{5,6} = \phi,$$

$$B^j_{8,8} = B^j_8 = \{(1,5,8), (2,8), (3,6,8)\},$$

and

$$B^j_{6,10} = \{3,6,10\}.$$

This simplifies the calculation of (34) and (38) considerably, as it reduces the number of summand and the length of the products. Putting (37) and (38) together, we get

$$\text{cov}(Y_i, Y_{i'}) = \sum_j \lambda_j \left[\sum_{(i^m_1, \ldots, i^m_{k_m}) \in B^j_{ii'}} a_{i^m_1 j} \cdots a_{i^m_{k_m} j} \right], \tag{39}$$

which is of the same general form as (35) combined with (34).

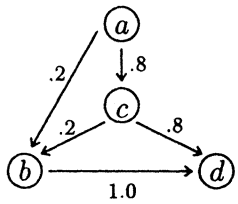

Figure 4. The Markov Network of SD Address (a, d) of Example 5.1.

5.1.3 A Simple Method for Constructing P of (34). For the following discussion, we fix the address j, and without loss of generality assume that the "active" links (i.e., links with positive probabilities) are labeled $1, \ldots, M$. Consider a Markov chain with $M+1$ states, where the states are the M links plus an additional dummy link that is an absorbing state labeled $M+1$ (or "end"), corresponding to an imaginary state that follows the last link of the path. (All paths are eventually absorbed at this state.) Let $\mathbf{Q}^{(j)}$ be the $(M+1) \times (M+1)$ transition probability matrix between these links. To reduce the notational clutter, we suppress the superscript j and use $\mathbf{Q}$ for $\mathbf{Q}^{(j)}$. $\mathbf{Q}$ is derived directly from the matrix $\mathbf{A}$ as follows: For two links, $i = (i_1, i_2)$ and $i' = (i'_1, i'_2)$, if a direct transition from i to i' is possible, so that $i_2 = i'_1$, then we set $q_{i,i'} = a^j_{i'}$. If $i_2 \neq i'_1$ (i.e., i is not directly connected to i'), then $q_{i,i'} = 0$. If $i_2 = j_2$ (i.e., = the destination node), then the next state is necessarily $M+1$, so that $q_{i,M+1} = 1$ if $i_2 = j_2$. Also $q_{M+1,M+1} = 1$. This is somewhat hard to see formally but easier to see via an example.

Example 5.1 (continued). Consider the address $j = (ad)(=$ "3") corresponding to the third column of $\mathbf{A}$ in Example 5.1. The corresponding network is depicted in Figure 4. There are five active links, and the matrix $\mathbf{Q}$ is the following 6×6 transition probability matrix:

$$\mathbf{Q} = \begin{array}{c} \\ 1\ (a \to b) \\ 2\ (a \to c) \\ 3\ (b \to d) \\ 4\ (c \to b) \\ 5\ (c \to d) \\ 6\ (end) \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \\ (a \to b) & (a \to c) & (b \to d) & (c \to b) & (c \to d) & (end) \end{array} \\ \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & .2 & .8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \tag{40}$$

We shall also need the two-step transition probability matrix

$$\mathbf{Q}^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & .2 & 0 & 0 & .8 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{41}$$

and the three-step transition probability matrix

$$\mathbf{Q}^3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{42}$$

Note that because there are no loops (by assumption), such a chain is always absorbed at state $M+1$ after *at most* M steps, and so $\mathbf{Q}^M$ is a matrix with columns of all zeros except the last column, which is all 1s.

The initial probabilities of the chain, say $\boldsymbol{\pi}$, are also read off the matrix $\mathbf{A}$. These probabilities are positive only for links connected to the source (j_1) of address j, and in this case $\pi_i = a^j_i$. In Example 5.1, for $j = (a, d) =$ "3", we get

$$\begin{aligned} \boldsymbol{\pi}' &\equiv (\pi_{a\to b}, \pi_{a\to c}, \pi_{b\to d}, \pi_{c\to b}, \pi_{c\to d}, \pi_{\text{end}}) \\ &= (.2, .8, 0, 0, 0, 0) \qquad (\pi_{\text{end}} \equiv 0). \end{aligned} \tag{43}$$

The probability that the chain is in "state i," $i = 1, \ldots, M$, after k steps is then $(\boldsymbol{\pi}'\mathbf{Q}^k)_i$, and the probability that the chain ever visits state i is then

$$\left(\boldsymbol{\pi}' \sum_{k=0}^{M} \mathbf{Q}^k\right)_i, \qquad i = 1, \ldots, M, \qquad (\mathbf{Q}^0 = \mathbf{I}).$$

This is seen as follows: Because there are no loops in the chain, the event [chain visits state i] is a union of mutually exclusive events, $\cup_{k=0}^{M}$ [chain visits state i at step k], and so we get (cf. (34))

$$\begin{aligned} &P[\text{chain visits state } i] \\ &\quad = \sum_{k=0}^{M} P[\text{chain visits state } i \text{ at step } k] \\ &\quad = \left(\boldsymbol{\pi}' \sum_{k=0}^{M} \mathbf{Q}^k\right)_i. \end{aligned} \tag{44}$$

Bringing back into (44) the suppressed dependency of $\mathbf{Q}$ and $\boldsymbol{\pi}$ on the SD address j, and noting that the event "chain visits state i" is equivalent to a "packet with SD address j passing through link i," we get

$$P^j_i = \left(\boldsymbol{\pi}^{(j)\prime} \sum_{k=0}^{M} [\mathbf{Q}^{(j)}]^k\right)_i. \tag{45}$$

Example 5.1 (continued). For the address $j = (ad)$, we get

$$(P^{ad}_{a\to b}, P^{ad}_{a\to c}, P^{ad}_{b\to d}, P^{ad}_{c\to b}, P^{ad}_{c\to d}) = (.2, .8, .36, .16, .64).$$

This is obtained as the first five $(= M)$ coordinates of $\pi^{(j)'}[\mathbf{Q}^0 + \mathbf{Q}^1 + \mathbf{Q}^2]$:

$$(.2, .8, 0, 0, 0, 0)[\mathbf{I} + \mathbf{Q}^1 + \mathbf{Q}^2] = (.2, .8, 0, 0, 0, 0) + (0, 0, .2, .16, .64, 0) + (0, 0, .16, 0, 0, .84) = (.2, .8, .36, .16, .64, .84).$$

Compare the vector (.2, .8, .36, .16, .64) with the third column of the matrix $\mathbf{P}$ in Example 5.1 and note that they are the same, except that we deleted the inactive links of the network in the current calculation, because for these links $p_i^{ad} = 0$.

5.1.4 A Simple Method for Constructing P of (38). The absence of loops induces a partial order on the network with respect to each address j. For any pair of links, i and i', they may be unconnected, in which case they are unordered, or they may be connected by a possible path, in which case they are ordered relative to their distance from j_1. We say that $i <_j i'$ if i and i' are connected and i precedes i' relative to the address j. If i and i' are unconnected, then $p_{ii'}^j = 0$. If they are connected, and supposing (without loss of generality) that $i <_j i'$, then a conditional probability argument gives the following factorization:

$$P_{ii'}^j = P_i^j \tilde{P}_{i'}^{(i_2, j_2)}, \tag{46}$$

where $i = (i_1, i_2), i' = (i_1', i_2')$, and $j = (j_1, j_2)$. Here P_i^j is as defined before (33), and $\tilde{P}_{i'}^{(i_2,j_2)}$ is the probability that a packet with address (i_2, j_2) (starting at node i_2 and ending at node j_2) passes through link i', under the transition probabilities of the network with the original address $j = (j_1, j_2)$. Both P_i^j and $\tilde{P}_{i'}^{(i_2,j_2)}$ can be calculated using the method given in (45).

5.1.5 Putting it Together. Expressions (45) and (35) together give the mean and variance of the Y_i's, and expressions (46) and (37) together give the covariances of the Y_i's. For estimation, we compare these quantities to the empirical moments, based on repeated measurements at the links (exactly as in Sec. 3), and treat the resulting system of linear equations as a LININPOS problem. Much of the discussion in Sections 1–4, which relates to implementation of the method in connection with bad equations, regularization toward the Poisson model via reduced weights to second-moment equations, and so on, is all relevant to the random-routing case.

6. CONCLUSIONS AND FINAL REMARKS

6.1 On the Problem (Formulation and Methodology)

The problem of estimating the node-to-node traffic intensity from repeated measurements of traffic on the links of a network is formulated and discussed under Poisson assumptions. The MLE and related approximations are discussed, and numerical difficulties are pointed out. A detailed methodology is presented for the method of moments. For the Poisson model, estimates based on the method of moments are consistent and asymptotically normal. The estimates are derived algorithmically, taking advantage of the fact that the first- and second-moment equations give rise to a linear inverse problem with positivity restrictions, which can be approached with an EM algorithm. This is done for two different types of network routing schemes: fixed (deterministic) and random (Markovian). A small simulation study is presented for a network with fixed routing, and the results indicate tendency toward unbiasedness already for moderate sample sizes.

6.2 On the Poisson Assumption

Without any probability model, one is left only with the linear equations (1), in which the $\mathbf{X}^{(k)}$ are from unspecified distribution(s). This cannot lead to anything useful. Thus a model is needed to tie all these equations together and relate the data to some underlying SD traffic pattern. A Poisson model is the most natural starting point for various reasons including the popularity of Poisson-related models for time-dependent processes such as conventional telephone and packet traffic (Heffes and Lucantoni, 1986), as well as vehicular traffic; the simplicity of transition from inference for Poisson variables (as in this article) to inference for Poisson processes; and the linearity and positivity in $\boldsymbol{\lambda}$ of the first two moments, which facilitate a simple estimation procedure. The method of weighted moment equations of Section 4 addresses estimation under deviations from the Poisson assumption and can also be viewed as a regularization technique for the Poisson model estimation. Clearly, however, there are situations of network data where Poisson-related models are totally inappropriate (see, for example, Leland et al. 1993). Such cases are governed by different probability models, and, accordingly, different estimation procedures, tailored to the specific models, would need to be derived.

6.3 On Applications to Road Planning

Knowing the node-to-node traffic intensity is a great design tool for traffic engineers and urban planners, as it is essential for deciding which nodes need to be linked directly to reduce network congestion. Historically, road side interviews and other types of surveys have been used for estimating the SD traffic intensities in transportation studies. In recent years more algebraic and statistical methods based on link data have been proposed. See Sherali, Sivanandan, and Hobeika (1994), Bell (1991), and the references therein, for important recent work and current practices. A forthcoming paper will discuss our methodology in the context of these applications, and its relation to the methods proposed in this field.

6.4 On an Early Interest in Networks

Finally, we would like to include earlier references (suggested by W. Whitt and M. Segal) to a related problem. Kruithof (1937) posed the problem of projecting from measured node-to-node teletraffic data to some future values, based on estimates of total originating and terminating traffic only. Krupp (1979) presented Kruithof's method, treated it with modern mathematical tools, and related it to more recent mathematical concepts, such as Kullback–Leibler in-

formation divergence. Kruithof's method can be considered a dual of the EM algorithm for LININPOS problems, as it also minimizes a Kullback–Leibler criterion function for such problems, but it does the minimization with respect to the first argument of the Kullback–Leibler function, whereas the EM does it with respect to the second argument (see also Byrne 1993 for a related analysis.)

APPENDIX: PROOF OF LEMMA OF SECTION 2

Let

$$s_j = \sum_{i=1}^{r} \mathbf{A}_i^{(j)} = \text{number of 1's in column } \mathbf{A}^{(j)}. \quad \text{(A.1)}$$

Without loss of generality, assume that the columns of $\mathbf{A}$ are arranged monotonically in $s_j, 1 \le s_1 \le \cdots \le s_c$. To prove identifiability, we need to show that if

$$P_{\mathbf{Y}}(\mathbf{y}|\boldsymbol{\lambda}) = P_{\mathbf{Y}}(\mathbf{y}|\tilde{\boldsymbol{\lambda}}) \quad \text{for a set of } \mathbf{y}\text{'s of measure } 1(\boldsymbol{\lambda}), \quad \text{(A.2)}$$

then necessarily $a\boldsymbol{\lambda} = \tilde{\boldsymbol{\lambda}}$. The proof has three steps. In Step 1 we show that (A.2) implies $\sum \lambda_i = \sum \tilde{\lambda}_i$, in Step 2 we show that it implies $\lambda_1 = \tilde{\lambda}_1$, and in Step 3 we use induction to complete the proof. In all three steps we calculate the probabilities of the possible solution vectors, $\mathbf{x}$, to the equations $\mathbf{y} = \mathbf{Ax}$ under the Poisson model.

Step 1: Take $\mathbf{y} = \mathbf{0}$ (vector of all zeros). Then, because the components of $\mathbf{x}$ must be natural numbers (because the x_i's are Poisson variables) and because $\mathbf{A}$ has no zero columns, the only feasible solution of $\mathbf{y} = \mathbf{Ax}$ is $\mathbf{x} = \mathbf{0}$. Combining this with (A.2), we have $e^{-\Sigma\lambda_i} = P_{\mathbf{Y}}(\mathbf{0}|\boldsymbol{\lambda}) = P_{\mathbf{Y}}(\mathbf{0}|\tilde{\boldsymbol{\lambda}}) = e^{-\Sigma\tilde{\lambda}_i}$, so that

$$\sum \lambda_i = \sum \tilde{\lambda}_i. \quad \text{(A.3)}$$

Step 2: Now take $\mathbf{y} = \mathbf{A}^{(1)}$ and consider again the solution, in $\mathbf{x}$, to $\mathbf{y} = \mathbf{Ax}$. Clearly, $\mathbf{x} = \boldsymbol{\delta}_1 \equiv (1, 0, \ldots, 0)'$ is a solution. To see that this is the unique solution, suppose (by negation) that there is another solution. Because $\mathbf{A}^{(1)}$ is a vector of zeros and 1s, so must be any feasible solution $\mathbf{x}$. If the first coordinate of $\mathbf{x}$ is 1, then the remaining must all be zeros, because we have $\mathbf{A}^{(1)} = \sum_{i=1}^{c} x_i \mathbf{A}^{(i)}$, so that $s_1 = \sum_{i=1}^{c} x_i s_i = s_1 + \sum_{i=2}^{c} x_i s_i$, and so $x_2 = \cdots = x_c = 0$. If the first coordinate of $\mathbf{x}$ is zero, then we get $s_1 = \sum_{i=2}^{c} x_i s_i$, and because $s_1 \le s_i, i = 2, \ldots, c$, we can have at most one $x_i (2 \le i \le c)$ equal to 1, but because $\mathbf{x}$ is a solution to $\mathbf{A}^{(1)} = \mathbf{Ax}$, it would imply $\mathbf{A}^{(1)} = \mathbf{A}^{(i)}$ for some $i \in \{2, \ldots, c\}$, contradicting the assumption that all the columns of $\mathbf{A}$ are different. Thus we have

$$P_{\mathbf{Y}}(\mathbf{A}^{(1)}|\boldsymbol{\lambda}) = P_{\mathbf{X}}(\delta_1|\boldsymbol{\lambda}) = e^{-\Sigma\lambda_i}\lambda_1$$

and, similarly, $P_{\mathbf{Y}}(\mathbf{A}^{(1)}|\tilde{\boldsymbol{\lambda}}) = e^{-\Sigma\tilde{\lambda}_i}\tilde{\lambda}_1$. It then follows from (A.2) and (A.3) that

$$\lambda_1 = \tilde{\lambda}_1. \quad \text{(A.4)}$$

Step 3 (induction step): Suppose that

$$\lambda_i = \tilde{\lambda}_i, \qquad i = 1, \ldots, j-1, \quad \text{for any} \quad j \in \{2, \ldots, c\}. \quad \text{(A.5)}$$

We want to show that $\lambda_j = \tilde{\lambda}_j$. Take $\mathbf{y} = \mathbf{A}^{(j)}$ and consider again the equation $\mathbf{y} = \mathbf{Ax}$. Clearly, $\mathbf{x} = \boldsymbol{\delta}_j \equiv (0, \ldots, 0, 1, 0, \ldots, 0)'$ (1 at position j, and zero elsewhere) is a solution. Also, because $\mathbf{A}^{(j)}$ is a vector of zeros and 1s, and because the x_i's are Poisson variables, any feasible solution must be a vector of zeros and 1s. For similar reasons as in Step 2, any solution must have $x_{j+1} = \cdots = x_c = 0$. Furthermore, if a solution has $x_j = 1$, then it must also have $x_1 = \cdots = x_{j-1} = 0$, because for such a solution, $s_j = s_j + \sum_{i=1}^{j-1} x_i s_i$, and hence $x_1 = \cdots = x_{j-1} = 0$ too. In other words, the only feasible solution of $\mathbf{A}^{(j)} = \mathbf{Ax}$ with $x_j = 1$ is $\mathbf{x} = \boldsymbol{\delta}_j$. Therefore, if $\mathbf{A}^{(j)} = \mathbf{Ax}$ has *any* solution in addition to $\mathbf{x} = \boldsymbol{\delta}_j$, then it must be a vector of zeros and 1s and have a tail of zeros starting at position j or earlier, so that $x_j = x_{j+1} = \cdots = x_c = 0$. Let $\mathbf{x}_1, \ldots, \mathbf{x}_k$ be all such distinct feasible solutions to $\mathbf{A}^{(j)} = \mathbf{Ax}$, and let $b_i \subseteq \{1, \ldots, j-1\}$ index the coordinates with value 1 of $\mathbf{x}_i, i = 1, \ldots, k$. (For example, if $\mathbf{x}_1 = (1, 0, 1, 0 \ldots 0)$ and $\mathbf{x}_2 = (1, 1, 0 \ldots 0)$, then $b_1 = \{1, 3\}$ and $b_2 = \{1, 2\}$.) We then have

$$P_{\mathbf{Y}}(\mathbf{A}^{(j)}|\boldsymbol{\lambda}) = P_{\mathbf{X}}(\boldsymbol{\delta}_j \text{ or } \mathbf{x}_1 \text{ or } \mathbf{x}_2 \ldots \text{or}$$

$$\mathbf{x}_k|\boldsymbol{\lambda}) = e^{-\Sigma\lambda_i}\left(\lambda_j + \prod_{i\in b_1}\lambda_i + \prod_{i\in b_2}\lambda_i + \cdots + \prod_{i\in b_k}\lambda_i\right).$$

(Note that $\boldsymbol{\delta}_j$ and $\mathbf{x}_1, \ldots, \mathbf{x}_k$ are all distinct.) Similarly,

$$P_{\mathbf{Y}}(\mathbf{A}^{(j)}|\tilde{\boldsymbol{\lambda}}) = e^{-\Sigma\tilde{\lambda}_i}\left(\tilde{\lambda}_j + \prod_{i\in b_1}\tilde{\lambda}_i + \prod_{i\in b_2}\tilde{\lambda}_i + \cdots + \prod_{i\in b_k}\tilde{\lambda}_i\right).$$

It follows from (A.2), (A.3), and the induction assumption (A.5) that

$$\lambda_j - \tilde{\lambda}_j = \sum_{l=1}^{k}\left(\prod_{i\in b_l}\tilde{\lambda}_i - \prod_{i\in b_l}\lambda_i\right) = 0, \quad \text{(A.6)}$$

where the last equality follows from the induction assumption because

$$b_l \subseteq \{1, \ldots, j-1\}, \qquad l = 1, \ldots, k.$$

This completes the proof of the induction step and of the lemma.

[Received January 1994. Revised May 1995.]

REFERENCES

Becker, R. A., and Wilks, A. R. (1993), "Network Traffic Analysis," AT&T Bell Labs technical report.

Bell, M. G. (1991), "The Estimation of Origin-Destination Matrices by Constrained Generalised Least Squares," *Transportation Research B*, 25B, pp. 13–22.

Byrne, C. L. (1993), "Iterative Image Reconstruction Algorithms Based on Cross-Entropy Minimization," *IEEE Transactions on Imaging Processes*, 2, 96–103.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977), "Maximum Likelihood from Incomplete Data via the EM Algorithm" (with discussion), *Journal of the Royal Statistical Society*, Ser. B, 39, 1–38.

Green, P. J. (1990a), "Bayesian Reconstructions From Emission Tomography Data Using a Modified EM Algorithm," *IEEE Transactions in Medical Imaging*, 9, 84–93.

——— (1990b), "On Use of the EM Algorithm for Penalized Likelihood Estimation," *Journal of the Royal Statistical Society*, Ser. B, 52, 443–452.

Heffes, H., and Lucantoni, D. M. (1986), "A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance," *IEEE Journal on Selected Areas in Communications*, 4, 856–868.

Iusem, A. N., and Svaiter, B. F. (1994), "A New Smoothing-Regularization Approach for a Maximum-Likelihood Estimation Problem," *Applied Mathematics & Optimization: An International Journal*, 29, 225–241.

Kruithof, J. (1937), "Telefoonverkeersrekening," *De Ingenieur*, 52, no. 8, E15–E25.

Krupp, R. S. (1979), "Properties of Kruithof's Projection Method," *The Bell Systems Technical Journal*, 58, 517–538.

Leland, W. E., Taqqu, M. S., Willinger, W., and Wilson, D. V. (1993), "On the Self-Similar Nature of Ethernet Traffic," presented at SIGCOMM '93.

Roeder, K., Devlin, B., and Lindsay, B. G. (1989), "Application of Maximum Likelihood Methods to Population Genetic Data for the Estimation of Individual Fertilities," *Biometrics*, 45, 363–379.

Sherali, H. D., Sivanandan, R., and Hobeika, A. G. (1994), "A Linear Programming Approach for Synthesizing Origin-Destination Trip Tables from Link Traffic Volumes," *Transportation Research B*, 28B, pp. 213–233.

Shepp, L. A., and Vardi, Y. (1982), "Maximum Likelihood Reconstruction for Emission Tomography," *IEEE Transactions on Medical Imaging*, 1, 113–122.

Silverman, B. W., Jones, M. C., Wilson, J. D., and Nychka, D. W. (1990), "A Smoothed EM Approach to Indirect Estimation Problems, With Particular Reference to Stereology and Emission Tomography" (with discussion), *Journal of the Royal Statistical Society*, Ser. B, 52, 271–324.

Snyder, D. L., Schultz, T. J., and O'Sullivan, J. A. (1992), "Deblurring Subject to Nonnegativity Constraints," *IEEE Transactions on Signal Processes*, 40, 1143–1150.

Vardi, Y., and Lee, D. (1993), "From Image Deblurring to Optimal Investments: Maximum Likelihood Solutions for Positive Linear Inverse Problems" (with discussion), *Journal of the Royal Statistical Society*, Ser. B, 55, 569–612.

Vardi, Y., Shepp, L. A., and Kaufman, L. (1985), "A Statistical Model for Positive Emission Tomography," (with comments), *Journal of the American Statistical Association*, 80, 8–37.