

DELAY AND TRAFFIC RATE ESTIMATION IN NETWORK TOMOGRAPHY

by

Neshat Etemadi Rad  
A Dissertation  
Submitted to the  
Graduate Faculty  
of  
George Mason University  
In Partial fulfillment of  
The Requirements for the Degree  
of  
Doctor of Philosophy  
Electrical and Computer Engineering

Committee:

\_\_\_\_\_ Dr. Yariv Ephraim, Co-director  
\_\_\_\_\_ Dr. Brian L. Mark, Co-director  
\_\_\_\_\_ Dr. Jill K. Nelson, Committee Member  
\_\_\_\_\_ Dr. James Gentle, Committee Member  
\_\_\_\_\_ Dr. Monson H. Hayes, Department Chair  
\_\_\_\_\_ Dr. Kenneth S. Ball, Dean, Volgenau School  
of Information Technology and Engineering

Date: \_\_\_\_\_ Fall Semester 2015  
George Mason University  
Fairfax, VA

Delay and Traffic Rate Estimation in Network Tomography

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

By

Neshat Etemadi Rad

Master of Science

Sharif University of Technology, Tehran, Iran, 2010

Bachelor of Science

Amirkabir University of Technology, Tehran, Iran, 2008

Co-director: Dr. Yariv Ephraim, Professor

Co-director: Dr. Brian L. Mark, Professor

Department of Electrical and Computer Engineering

Fall Semester 2015

George Mason University

Fairfax, VA

Copyright © 2015 by Neshat Etemadi Rad  
All Rights Reserved

## Dedication

To my parents, Avisa and Hamid, for without their early inspiration and coaching, none of this would have happened.

To the love of my life, Abbas, for without his support and enthusiasm, none of this would have been accomplished.

## Acknowledgments

I would like to thank my dissertation advisors, Professor Yariv Ephraim and Professor Brian L. Mark, for being supportive and patient throughout my PhD research at George Mason University. I am very grateful to them for their in-depth technical knowledge, guidance and insightful discussions. I would like to also thank the rest of my committee members, Professor Jill K. Nelson and Professor James Gentle, who have reviewed the dissertation carefully, provided thoughtful suggestions, and challenged me with insightful questions.

I will forever be thankful to my parents, my sister, Negar, and my beloved husband, Abbas. My parents have devoted their lives for me and my sister. I would not have reached it this far without their unconditional love, support, and encouragement. My sister has always been the one I could count on at tough times. My husband, Abbas, has been always my best friend and provided his support, care and love. I truly thank Abbas for having faith in me and sticking by my side during my difficult times.

This work was supported in part by the U.S. National Science Foundation under grant CCF-0916568.

# Table of Contents

	Page
List of Tables . . . . .	vii
List of Figures . . . . .	viii
Abstract . . . . .	x
1 Introduction . . . . .	1
1.1 Overview of Network tomography . . . . .	1
1.2 Thesis contribution . . . . .	2
1.3 Thesis outline . . . . .	4
2 Literature review . . . . .	5
2.1 Delay network tomography . . . . .	5
2.1.1 Estimation of propagation link delay . . . . .	5
2.1.2 Estimation of link delay density . . . . .	6
2.2 Traffic network tomography . . . . .	9
2.2.1 Vardi’s model . . . . .	9
2.2.2 Vanderbei and Iannone’s model . . . . .	15
2.2.3 Maximum likelihood parameter estimation . . . . .	16
2.2.4 Vardi’s moment matching approach . . . . .	19
2.2.5 Tebaldi and West Bayesian approach . . . . .	22
2.2.6 Other related works . . . . .	23
2.3 Other aspects of network tomography . . . . .	24
3 Background on bivariate Markov chains . . . . .	25
3.1 Continuous-time Bivariate Markov chain . . . . .	25
4 Delay Network Tomography using a Partially Observable Bivariate Markov Chain	33
4.1 Partially observable bivariate Markov chain model . . . . .	33
4.2 Maximum likelihood parameter estimation . . . . .	38
4.2.1 Identifiability . . . . .	38
4.2.2 EM algorithm . . . . .	39
4.3 Tree-Structured Networks . . . . .	46
4.4 Numerical Results . . . . .	48
4.4.1 Unstructured Network . . . . .	49

4.4.2	Tree-Structured Networks . . . . .	54
4.4.3	Recursive implementation of the EM algorithm . . . . .	56
5	Source-destination traffic rates estimation . . . . .	61
5.1	Covariance-based rate estimation . . . . .	61
5.2	Maximum-entropy approach . . . . .	64
5.2.1	Shannon's Entropy . . . . .	64
5.2.2	Underlying framework . . . . .	64
5.3	Numerical results . . . . .	68
5.3.1	Simulation set-up . . . . .	68
5.3.2	Results . . . . .	69
6	Conclusions and Future directions . . . . .	73
6.1	Conclusion . . . . .	73
6.2	Future directions . . . . .	74
A	An Appendix . . . . .	75
	Bibliography . . . . .	77

## List of Tables

Table	Page
2.1 Routing matrix $A$ for network in Fig. 2.3. . . . .	10
2.2 Example of a random routing matrix $A$ for the network of Fig. 2.4. . . . .	12
2.3 Link usage probabilities $\{P_i^j\}$ for source-destination $(a, d)$ in Fig. 2.5. . . . .	13
2.4 The embedded zero-one matrix from the random routing matrix of Table 2.2 $\tilde{A}, [1]. . . . .$	15
2.5 Matrix $A$ of the network in Fig. 2.6 . . . . .	16
4.1 Divergence between true and estimated densities in the tree-structured network.	55
4.2 Divergence between true and estimated densities in the unstructured network obtained from the online algorithm. . . . .	59

## List of Figures

Figure	Page
2.1 An unstructured network with $N = 5$ nodes . . . . .	6
2.2 A tree network topology . . . . .	8
2.3 An example of a 4-node directed network [2, Example 1]. . . . .	10
2.4 A sketch of 4-node directed network with random routing regime. . . . .	11
2.5 Markov chain for SD $(a, d)$ . . . . .	12
2.6 A network of 4 nodes. . . . .	16
4.1 True diagonal elements of $H_{cc}$ depicted in ascending order, and their estimates. . . . .	50
4.2 True and estimated overall source-destination delay densities. . . . .	51
4.3 True and estimated link delay densities are shown in (a) and (b) for two of the links. . . . .	53
4.4 Divergence values for the estimated link delay densities. . . . .	53
4.5 Mean squared error for estimated packet routing probabilities on the various links. . . . .	54
4.6 True and estimated source-destination delay densities in the tree-structured network. . . . .	56
4.7 True and estimated link delay densities for the five different links in the tree-structured network. . . . .	57
4.8 True and estimated overall source-destination delay densities obtained from online algorithm. . . . .	58
4.9 Divergence values for the estimated link delay densities obtained from online algorithm with (a) $L = 10$ (b) $L = 100$ and (c) $L = 1000$ . . . . .	59
4.10 Mean squared error for estimated packet routing probabilities on the various links obtained from online algorithm with (a) $L = 10$ (b) $L = 100$ and (c) $L = 1000$ . . . . .	60
5.1 The mean of estimated source-destination rates in a network with $N = 4$ nodes and $c = 12$ source-destination pairs. . . . .	70
5.2 The mean squared error of estimated source-destination rates in a network with $N = 4$ nodes and $c = 12$ source-destination pairs. . . . .	70

5.3	The mean of estimated source-destination rates in a network with $N = 10$ nodes and $c = 90$ source-destination pairs. . . . .	71
5.4	The mean squared error of estimated source-destination rates in a network with $N = 10$ nodes and $c = 90$ source-destination pairs . . . . .	72
5.5	The mean and the mean squared error of the estimated rates as obtained from the covariance-based scheme using $K = 100$ and $K = 100,000$ . . . . .	72

## Abstract

DELAY AND TRAFFIC RATE ESTIMATION IN NETWORK TOMOGRAPHY

Neshat Etemadi Rad, Ph.D.

George Mason University, 2015

Dissertation Co-director: Dr. Yariv Ephraim

Dissertation Co-director: Dr. Brian L. Mark

Network tomography deals with estimation of computer network features from measurements on links or terminal nodes. The area was pioneered with the work of Vanderbei and Iannou in 1994 and Vardi in 1996. Of particular interest are estimation of source-destination traffic rates from link packet counts or from aggregated packet counts in input and output nodes, and estimation of link delay from source-destination delay measurements. Traffic rate estimation, and link propagation delay estimation, are inverse problems which require the solution of under-determined sets of linear equations. Iterative solutions based on moment matching and the expectation-maximization algorithm were proposed for traffic rate estimation, and a maximum entropy approach was developed for link propagation delay estimation. Traffic rate estimation was also performed using a Bayesian estimation approach. Estimation of link delay densities commonly involves exponential mixture models which entail independence of the delay on various links. Network tomography is useful for monitoring the performance of a network, and thus maintaining and expanding the network. Network tomography is equally applicable to other networks such as rails, roads, or social network.

The main contribution of this thesis is a new approach for estimating the aggregated delay density on each link of the network from source-destination delay measurements. Our approach is based on modeling traffic over the network as a continuous-time bivariate Markov chain, whereas nodes of the network are associated with states of one of the two chains comprising the bivariate process. The sojourn time in each node of the network is determined by the other chain of the bivariate Markov process. The density of this sojourn time is phase type. The family of phase type densities is very rich and it is closed under mixture and convolution operations. This approach is more general than existing approaches which rely on independent link delays that are modeled as mixtures of exponentials. Mixtures of convolutions of exponential densities are particular phase type densities. We develop an expectation-maximization algorithm for estimating the parameter of the model, which in turn is used to evaluate the density for each link. As by products of this approach, the estimated parameter of the model can also be used to estimate routing probabilities in each node as well as the probability of any source-destination path in the network. Another contribution of this thesis addresses estimation of source-destination traffic rates from aggregated packet counts in input and output nodes of the network. We develop a simple covariance-based solution, and an alternative approach which invokes the maximum entropy principle. We have simulated a computer network and numerically studied the algorithms we developed and compared the results with other benchmarks in the literature.

# Chapter 1: Introduction

## 1.1 Overview of Network tomography

A telecommunication network consists of nodes and links. Each node may represent an element such as a computer or a router. A link is a path connecting two nodes directly and does not contain any other node. A source-destination route between a pair of nodes can be composed of several links and nodes. The routing regime in the networks is either random or deterministic. In a random routing network, there exist multiple routes for each source-destination pair, and routes are chosen according to some distribution. In a deterministic routing regime, there is a single route for each source-destination pair. Clearly, the deterministic routing regime may be considered as a particular case of a random routing regime. Network management requires monitoring the internal parameters of the network such as source-destination traffic rates, link delay densities and link loss rates.

Packets traveling through the network may undergo four types of link delays: 1) processing delay 2) queueing delay 3) transmission delay, and 4) propagation delay [3]. While queueing delay is random, the other delay types can be treated as deterministic. Link delay can adversely affect communication over the network such as that of video or speech signals. Therefore, assessment and characterization of link delay are of great importance. The source-destination traffic rates can also provide network administrators with vital information for re-routing and balancing traffic across the network. For example, the source-destination traffic rates may be used by the internet service providers to charge the customers according to their use of data. Hence, studying, modeling and analyzing of the traffic is of considerable importance. While it is essential to study network internal characteristics, it is not always feasible and efficient to measure the internal parameters directly in today's vast networks. For example, due to security reasons, some of the internal nodes may not

be accessible. Also, it is not computationally and economically efficient to measure and store all the information provided by the internal routers. Therefore, the area of network tomography has emerged. Vanderbei and Iannone [4], and Vardi [2], studied estimation of internal parameters of a computer network from some measurements taken from accessible nodes or links of that network. Vardi [2] coined the term “network tomography” following Shepp and Vardi’s earlier work on positron emission tomography (PET) [5].

Network tomography has several aspects. In this thesis, we only study the two main aspects, namely, estimation of source-destination traffic rates from traffic count measurements at some nodes or links of the network, and inference of link delay from source-destination delay measurements. Other aspects of network tomography include link loss rate estimation and topology identification from source-destination measurements. The link or source-destination measurements can be done either passively or actively. In passive measurement, the traffic flow of the network is monitored and used to obtain the measurements. In active measurement, test probes are transmitted across the network to obtain the measurements. We provide more details on active probing techniques in Section 2.1.2. A survey of research related to various aspects of network tomography can be found in [6].

## 1.2 Thesis contribution

The present work makes several contributions to the problem of link delay and source-destination traffic rates estimation from source-destination measurements, as follows.

- We formulate the problem of link delay density estimation from source-destination delay measurements in an unstructured network with random routing regime. An unstructured network may have any desirable structure such as the tree structure. We do not distinguish between the random and deterministic components of delay on each link. We propose to model traffic over the network as a partially observable bivariate Markov chain. A bivariate Markov chain  $Z$  comprises a pair of random processes  $(X, S)$  which are jointly Markov. The states of  $X$ -chain represent the nodes of

the network, while the  $S$ -chain controls the statistical properties of the  $X$ -chain, for example, the sojourn time in each state of the  $X$ -chain. The two chains are hidden since we only measure the source-destination delay. The sojourn time of the process in each state of the  $X$ -chain is phase-type [7]. Phase-type distributions are rather general and may be used to approximate any desirable sojourn time distribution. For example, mixtures of convolutions of exponential distributions are particular phase-type distributions. In the proposed model, the link delay density has matrix exponential distribution which is a generalization of phase-type distribution [8]. The proposed approach also provides the routing probabilities in the random routing regime, as well as the probability of each source-destination path.

- We develop an EM algorithm to estimate the parameter of the bivariate Markov chain based on the work in [9]. In [9], the parameter of a Markov chain with a single absorbing state was estimated from independent absorbing times. We do not use numerical integration in the E-step as in [9]; instead a fundamental result from [10] is used to evaluate the integrals. Our proposed approach for estimating link delay density from source-destination delay measurements is fairly general and does not require any prior assumption regarding the topology of the network and the form of the density on the link delay.
- We test the proposed model for link delay density estimation in both unstructured and tree-structure network. We simulate the real data using a *high order* bivariate Markov chain model. We use the simulated data to estimate the parameter of a *low order* bivariate Markov chain. Then, we use the estimated parameter to evaluate the delay density over various links as well as the packet routing probabilities. The present work is compared with the work in [11] where tree-structured network with mixture of exponentials were used. The numerical results show that the source-destination delay density, link delay densities and routing probabilities are very well represented using the proposed approach. We also discuss recursive implementation of the EM

algorithm using blocks of data in a sequential manner.

- We study the problem of source-destination traffic rates estimation from aggregated traffic flow at some nodes of the network. We discuss the two main approaches presented in [2] and [4]. We re-formulate the model of [4] to estimate source-destination traffic rates from aggregated traffic counts at input and output nodes, and discuss an interesting feature of that model. In particular, we show that estimation of source-destination rates could follow from a simple covariance matching approach. We also develop a maximum entropy solution for the rate estimation problem of [4], and provide numerical results to evaluate and compare the performance of the proposed solutions with the work of [4] and [2].

### 1.3 Thesis outline

The remainder of this thesis is organized as follows. In Chapter 2, we review existing models and approaches for the two aspects of network tomography studied here. Chapter 3 provides a review on the continuous-time bivariate Markov chains. In Chapter 4, we study the problem of link delay density estimation from source-destination delay measurements and develop a new parametric model. The performance of the proposed approach is evaluated through several numerical examples. In Chapter 5, we re-formulate the rate estimation problem of [4], and develop the covariance-based approach as well as the maximum entropy solution. Numerical results are provided to demonstrate the performance of the proposed solutions. Finally, Chapter 6 concludes the thesis and discusses additional research areas which can be extended from the present work.

## Chapter 2: Literature review

In this chapter, we address several aspects of network tomography including estimation of source-destination traffic rates, link delay and link loss rate, and topology identification from accessible measurements in the network. Some of the existing challenges and techniques to approach each of these aspects are discussed.

### 2.1 Delay network tomography

In this section, we address an important aspect of network tomography, namely, delay network tomography which aims at link delay inference from source-destination delay measurements. These measurements are collected through either passive or active probing the network. In both passive and active probing schemes, a probe time-stamped packet is sent. The destination node marks its own time stamp on the probe packet once it reaches the destination. The difference between these two time stamps is recorded as the source-destination delay, if the clocks of the nodes are synchronized.

#### 2.1.1 Estimation of propagation link delay

Several authors studied the problem of estimation of link propagation delay from source-destination delay measurements, see, e.g, [12] and [13]. The propagation delay, as mentioned earlier, is one of the deterministic components of link delay.

In [12], an unstructured network was studied where the clocks of the nodes were not synchronized, and the propagation delay on the links were estimated. In such a network, the clocks of the nodes have some offset with respect to a “Universal Time”, and the time it takes for a packet to travel along a closed path was referred to as the *round-trip* delay [12]. Consider the network depicted in Fig. 2.1. The round-trip delay, however, does not

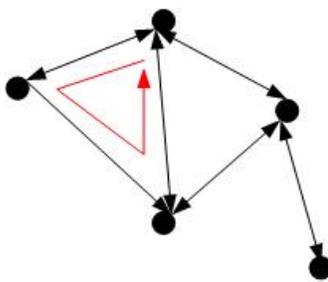


Figure 2.1: An unstructured network with  $N = 5$  nodes

include any clock offset since the clock drifts are cancelled along a closed path. In [12], the propagation round-trip delay associated with each closed path was identified as the smallest round-trip delay measured for multiple different packets along that path. Next, the authors in [12] identified as many closed paths as possible, and constructed a set of under-determined linear equations. Least square and maximum entropy approaches were studied in [12] to estimate the propagation delay on the links from round-trip delay measurements. In some network tomography applications, a network administrator is interested in identifying a few links with higher propagation delay from source-destination delay measurements. This problem was studied in [13] where the vector of propagation link delays was assumed sparse, and compressed sensing principle was used to estimate that vector.

### 2.1.2 Estimation of link delay density

The study of link delay density from source-destination measurements was performed in, for example, [14], [15], [16], [17], [18], [19], [20], [21] and [11]. Some authors have focused on tree-structured networks in which packets may enter only at the root node and transverse to one of the terminal nodes, see, e.g, [21], [15], [16], [19] and [11]. Both random routing as well as deterministic routing are possible in tree-structured networks. In a binary tree, for example, one of two branches is possible from each node, and a particular branch is chosen according to a binary distribution. In a tree-structured network, some researches rely on

active network tomography in which test probes are transmitted across the network using different probing schemes, see, e.g, [14], [15], [16], [17], [18] [20] and [21]. Next, we address some of the active probing schemes including unicast and multicast probing. Consider the tree-structured network depicted in Fig. 2.2. The unicast probing scheme is based on sending a probe packet from a source node, say node 0, to a destination node, say node 2. In multicast probing, a probe packet is sent from a source node, say node 0, to a group of destination nodes, say node 2 and node 3, as follows. The probe packet is sent on the link follows from node 0 to node 1. At node 1, the packet is replicated and sent on the link from node 1 to node 2 and on the link from node 1 to 3, and the source-destination delays from node 0 to node 2 and from node 0 to node 3 are measured. The advantageous of using multicast probing is that multicast probe packets observed at multiple destinations experience the same delays on shared links among their path from root to the destination, while independent unicast probing does not generate correlated measurements. On the other hand, due to security reasons, the multicast scheme is not enabled in many networks. Hence, a packet pair probing scheme was suggested in [22] and [23] in which a pair of unicast probe packets are sent back-to-back from the root node along distinct paths. In [24], a new probing scheme based on sending unicast packets to a group of receivers in tandem was developed. A “flexicast” scheme was developed in [17] in which groups of destination nodes used for probing have different sizes.

The study of queueing delay on links requires choosing an statistical model that best describes the behaviour of source-destination delay measurements. Due to the lack of knowledge about link delay density, [14], [16], [17], [20] and [21] used non-parametric discrete models for the link delay density where the packet was assumed lost if the delay is infinity. Conditions for identifiability of the discrete models were given in [17]. In [14], multicast probes were utilized, and link delay variances were estimated from the covariances of source-destination delay measurements. In [20], a mixture model was studied for the link delay density, and multicast probes were utilized. The model described in [20] was comprised of three components: a point mass for zero delay, a discrete uniform distribution, and an

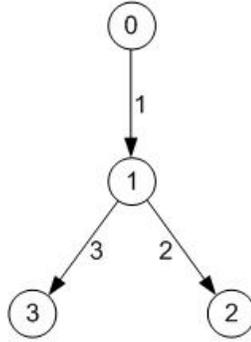


Figure 2.2: A tree network topology

exponential density. The proposed approach in [20] for link delay density estimation was based on matching the Fourier transform of the source-destination delay density. In [19], a discrete/continuous mixture model was studied for link delay density where the discrete component was a point mass for zero delay and the continuous component was a Gaussian mixture density. An EM algorithm was developed in [19] to estimate the link delay density. In [21], a Markov model was studied to capture the dependencies among delays on different links, and the discrete link delays were estimated.

Non-parametric models are not always a good approximation of true parametric ones. Therefore, parametric models for link delay density have been proposed in some researches. For example, Gaussian link delay densities were studied in [17] and exponential models were developed in [11]. The necessary and sufficient conditions for identifiability of the exponential link delay density estimation problem was established in [11]. However, the Gaussian and exponential models are not suitable to describe the heavy tail of link delay density [25]. This motivates the introduction of other models as more realistic ones. Gamma models were developed in [18] where flexicast probes were utilized, and a moment matching approach was studied.

In [11], a tree-structure network was studied, and the link delay density was modeled by a mixture of exponential densities. The delays corresponding to different links were

assumed independent, and an EM algorithm was used to estimate the parameter of the mixture model in [11]. The moment generating function of exponential density was used to implement the EM algorithm in [11]. Our approach in link delay density estimation from source-destination delay measurements in Chapter 4 can be regarded as a generalization of the work in [11].

## 2.2 Traffic network tomography

Traffic network tomography aims at estimating source-destination traffic rates in a computer network from measurements taken from that network. We next detail some of the existing models and estimation approaches developed in the literature for estimation of the source-destination traffic rates. Throughout this thesis, capital letters and lower case letters are used to denote random variable and their realizations, respectively.

### 2.2.1 Vardi’s model

Vardi, who coined the term “network tomography”, studied estimation of traffic rate over source-destination pairs from link counts in [2], for networks with deterministic routing regime and networks with random routing regime.

#### Deterministic routing regime

We demonstrate the principle of Vardi’s model for deterministic routing through a simple example from [2]. Consider the network with  $N = 4$  nodes,  $c = 12$  source-destination pairs and  $q = 7$  links, as depicted in Fig. 2.3 [2, Example 1]. In deterministic routing, packets with a given source-destination travel on a pre-determined route. The  $j$ th source-destination pair may be referred to as the  $(j_1, j_2)$  pair where  $j_1$  denotes the source node and  $j_2$  denotes the destination node. For  $j \in \{1, \dots, c\}$ , let  $U_j$  denote the number of packets originated from  $j_1$  and destined to  $j_2$ . The random variable  $U_j$  was in [2] assumed to be a Poisson random variable with rate  $\lambda_j$ . Let  $\mathbf{U} = (U_1, \dots, U_c)'$  denote a column vector where

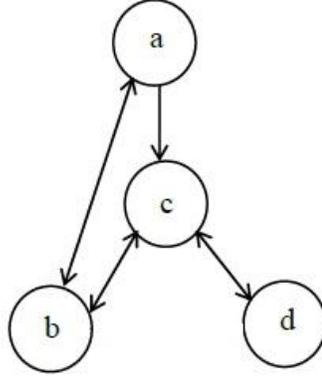


Figure 2.3: An example of a 4-node directed network [2, Example 1].

Table 2.1: Routing matrix  $A$  for network in Fig. 2.3.

	$(a, b)$	$(a, c)$	$(a, d)$	$(b, a)$	$(b, c)$	$(b, d)$	$(c, a)$	$(c, b)$	$(c, d)$	$(d, a)$	$(d, b)$	$(d, c)$
$a \rightarrow b$	1	0	0	0	0	0	0	0	0	0	0	0
$a \rightarrow c$	0	1	0	0	0	1	0	0	0	0	0	0
$b \rightarrow a$	0	0	1	1	0	1	1	0	0	1	0	0
$b \rightarrow c$	0	0	0	0	1	0	0	0	0	0	0	0
$c \rightarrow b$	0	0	0	0	0	0	1	1	0	1	1	0
$c \rightarrow d$	0	0	1	0	0	1	0	0	1	0	0	0
$d \rightarrow c$	0	0	0	0	0	0	0	0	0	1	1	1

' is the matrix transpose. The random variables  $\{U_1, \dots, U_c\}$  were assumed statistically independent in [2]. Let  $\lambda = (\lambda_1, \dots, \lambda_c)'$ .

Let  $A = \{a_{ij}, i = 1, \dots, q; j = 1, \dots, c\}$  denote a  $q \times c$  routing matrix where  $a_{ij} = 1$  if the path from  $j_1$  to  $j_2$  goes through link  $i$ , and  $a_{ij} = 0$  otherwise. The routing matrix for the network of Fig. 2.3 is given in Table 2.1. In this table,  $a \rightarrow b$  represents the link between node  $a$  and node  $b$ , and  $(a, b)$  represents the source-destination pair corresponding to these two nodes.

For  $i \in \{1, \dots, q\}$ , let  $V_i$  denote the number of packets travelled on link  $i$ . Let  $\mathbf{V} =$

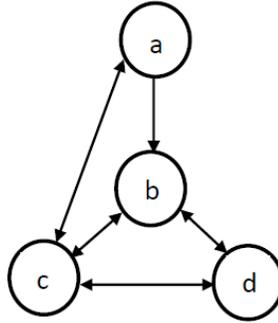


Figure 2.4: A sketch of 4-node directed network with random routing regime.

$(V_1, \dots, V_q)'$ . We have that  $\mathbf{V} = \mathbf{A}\mathbf{U}$ . Clearly, for  $i \in \{1, \dots, q\}$ ,  $V_i$  is a Poisson random variable, but the random variables  $\{V_1, \dots, V_q\}$  are not statistically independent. This presents a major difficulty in using this model.

### Random routing regime

In a random routing network, packets with source-destination  $j$  may travel on different routes selected according to some probability distribution. Deterministic routing regime can be regarded as a particular case of random routing regime. In the random routing model as described by Vardi [2], Markov chains are attributed to each source-destination pair. The states of each Markov chain are a subset of the nodes of the network. For the source-destination pair  $j$ , the transition probability from one node, say  $i_1$ , to another, say  $i_2$ , is the probability that upon leaving node  $i_1$ , the packet travels on the link  $i = (i_1, i_2)$ . This probability is simply  $a_{ij}$ . As an example, consider the network from [2, Example 5.1] depicted in Fig. 2.4 with  $N = 4$ ,  $q = 9$  links and  $c = 12$  source-destination pairs. The routing matrix for this example is shown in Table 2.2. The Markov chain for the source-destination pair  $(a, d)$ , for example, is shown in Fig. 2.5.

In the random routing network, let  $V_i^j$  denote the number of packets that pass through link  $i$  with source-destination address  $j$ , and  $P_i^j$  denotes the probability that link  $i$  will be

Table 2.2: Example of a random routing matrix  $A$  for the network of Fig. 2.4.

	$(a, b)$	$(a, c)$	$(a, d)$	$(b, a)$	$(b, c)$	$(b, d)$	$(c, a)$	$(c, b)$	$(c, d)$	$(d, a)$	$(d, b)$	$(d, c)$
$a \rightarrow b$	.8	.2	.2	0	0	0	0	0	0	0	0	0
$a \rightarrow c$	.2	.8	.8	0	1	1	0	0	0	0	0	0
$b \rightarrow a$	0	0	0	1	.2	.1	1	0	0	1	0	0
$b \rightarrow c$	0	.8	0	0	.8	.1	0	0	0	0	0	1
$b \rightarrow d$	0	.2	1	0	0	.8	0	0	1	0	0	0
$c \rightarrow b$	.8	0	.2	0	0	0	.8	.8	.2	1	1	0
$c \rightarrow d$	.2	0	.8	0	0	1	.2	.2	.8	0	0	0
$d \rightarrow b$	1	0	0	0	0	0	1	1	0	.8	.8	.2
$d \rightarrow c$	0	1	0	0	0	0	0	0	0	.2	.2	.8

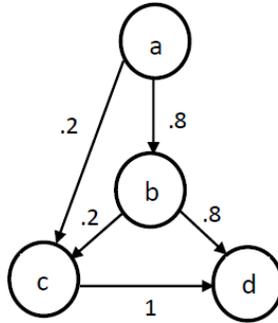


Figure 2.5: Markov chain for SD  $(a, d)$ .

Table 2.3: Link usage probabilities  $\{P_i^j\}$  for source-destination  $(a, d)$  in Fig. 2.5.

Link:	$a \rightarrow b$	$a \rightarrow c$	$c \rightarrow b$	$c \rightarrow d$	$d \rightarrow b$
Probability:	.2	.8	.36	.16	.64

used by the source-destination  $j$ . By the thinning property of Poisson random variables for  $i \in \{1, \dots, q\}$ ,  $\{V_i^1, \dots, V_i^c\}$  are mutually independent Poisson random variables with corresponding rates  $\{\lambda_1 P_i^1, \dots, \lambda_c P_i^c\}$ . We have

$$V_i = \sum_{j=1}^c V_i^j. \quad (2.1)$$

Hence, for  $i \in \{1, \dots, q\}$ ,  $V_i$  is a Poisson random variable with rate  $\sum_{j=1}^c \lambda_j P_i^j$ . Let  $P_{ii'}^j$  denote the probability that a packet with source-destination  $j$  passes through both link  $i$  and link  $i'$ , then

$$\text{cov}(V_i, V_{i'}) = \sum_j \lambda_j P_{ii'}^j. \quad (2.2)$$

The probabilities  $\{P_i^j\}$  could be evaluated by considering all paths that are used by packets with source-destination address  $j$  and travel through link  $i$ . These probabilities are shown in Table 2.4 for the Markov chain corresponding to source-destination pair  $(a, d)$  in Fig. 2.5. The scheme proposed by Vardi to evaluate  $\{P_i^j\}$  as well as  $\{P_{ii'}^j\}$  is described in [2].

Tebaldi and West developed an alternative approach for modeling the traffic over a random routing network as follows [1]. Consider, as before, a network with  $c$  source-destination pairs and  $q$  links. As the routing is Markovian, packets leaving the source may take different paths to their destination. Let  $k_j$  denote the number of possible paths from  $j_1$  to  $j_2$  which are numbered as  $t = 1, \dots, k_j$ . Suppose that each packet with source-destination address

$j$  is routed through path  $t$  with probability  $\pi_{jt}$  such that  $\sum_{t=1}^{k_j} \pi_{jt} = 1$ .  $U_{jt}$  denotes the number of packets out of the  $U_j$  packets that are routed through path  $t$  such that

$$\sum_{t=1}^{k_j} U_{jt} = U_j. \quad (2.3)$$

Given  $U_j$ , it follows from the thinning property of Poisson random variables, that  $\{U_{j1}, \dots, U_{jk_j}\}$  are mutually independent Poisson random variables with the corresponding rates  $\{\lambda_j \pi_{j1}, \dots, \lambda_j \pi_{jk_j}\}$ . Let  $\tilde{A}$  denote a  $q \times \sum_j k_j$  zero-one matrix which is established by substituting each  $j$ th column of  $A$  with  $k_j$  columns of zero-one entries, where a one entry indicates that the link belongs to the  $j$ th path. Also,  $\tilde{\mathbf{U}} = (U_{11}, \dots, U_{1k_1}, \dots, U_{c1}, \dots, U_{ck_c})'$  can be obtained by replacing each  $j$ th entry of  $\mathbf{U}$  by  $(U_{j1}, \dots, U_{jk_j})'$ . Hence, a set of equations which is similar to the set  $\mathbf{V} = A\mathbf{U}$  in the deterministic routing regime can be obtained. Here we have  $\mathbf{V} = \tilde{A}\tilde{\mathbf{U}}$ .

Thus, the random routing matrix is embedded in a larger zero-one deterministic routing matrix, with a new parameter  $\{\lambda_j \pi_{jt}, j = 1, \dots, c; t = 1, \dots, k_j\}$ . The embedding of the random routing matrix from Table 2.2 in a zero-one matrix is shown in Table 2.4. Denoting the estimate of  $\lambda_j \pi_{jt}$  as  $\widehat{\lambda_j \pi_{jt}}$ , and using (2.3), the estimate of  $\lambda_j$ , is given by

$$\widehat{\lambda_j} = \sum_{t=1}^{k_j} \widehat{\lambda_j \pi_{jt}}. \quad (2.4)$$

In addition, the estimates of the routing probabilities of paths are given by

$$\widehat{\pi_{jt}} = \frac{\widehat{\lambda_j \pi_{jt}}}{\sum_{s=1}^{k_j} \widehat{\lambda_j \pi_{js}}}, \quad t = 1, \dots, k_j. \quad (2.5)$$

Alternatively, given the routing matrix  $A$ , the probability  $\pi_{jt}$  is determined by the

Table 2.4: The embedded zero-one matrix from the random routing matrix of Table 2.2 ,  $\tilde{A}$ , [1].

	(a, b)			(a, c)			(a, d)			(b, a)	(b, c)		(b, d)			(c, a)		(c, b)		(c, d)		(d, a)		(d, b)		(d, c)	
	.8	.16	.04	.8	.16	.04	.64	.2	.16	1	.8	.2	.8	.1	.1	.8	.2	.8	.2	.2	.8	.2	.8	.8	.2	.2	.8
$a \rightarrow b$	1	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$a \rightarrow c$	0	1	1	1	0	0	1	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
$b \rightarrow a$	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	1	1	0	0	0	0	1	1	0	0	0	0
$b \rightarrow c$	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
$b \rightarrow d$	0	0	0	0	0	1	0	1	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
$c \rightarrow b$	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	0
$c \rightarrow d$	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0	0	0	0	0	0
$d \rightarrow b$	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	0	1
$d \rightarrow c$	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0

product of the appropriate transition probabilities in the  $j$ th column of  $A$ . For example, when  $j = (a, d)$  in Table 2.2, there are  $k_j = 3$  possible paths,  $\{abd, acbd, acd\}$ , which can be chosen with probabilities  $\{.2, .16, .64\}$ , respectively. Hence, the elements of  $A$ ,  $\{a_{ij}\}$ , can also be estimated from  $\{\hat{\pi}_{jt}, t = 1, \dots, k_j\}$ , vice versa. As a result, drawing Markov chains for the various source-destination pairs, the random routing problem could have been treated in a similar way to deterministic case by utilizing an embedded fat zero-one matrix derived from introduction of different paths.

### 2.2.2 Vanderbei and Iannone's model

In [4], aggregated traffic counts at input and output nodes were measured, and the traffic over a network with  $N$  nodes and  $c = N(N - 1)$  source-destination pairs was modeled as follows. For  $i \in \{1, \dots, N\}$ , let  $V_i$  denote the number of packets originated from node  $i$ . For  $i \in \{N + 1, \dots, q = 2N - 1\}$ , let  $V_i$  denote the number of packets destined at node  $i - N$ . In the zero-one matrix  $A = \{a_{ij}\}$ ,  $a_{ij} = 1$  if  $U_j$  contributes to  $V_i$ , and  $a_{ij} = 0$  otherwise. Hence, we have  $\mathbf{V} = \mathbf{A}\mathbf{U}$ . An example of a network with  $N = 4$  nodes and  $c = 12$  source-destination pairs is depicted in Fig. 2.6. The corresponding  $A$  matrix for the network of Fig. 2.6 is given in Table 2.5. We discuss some of the interesting features of this model in Chapter 5.

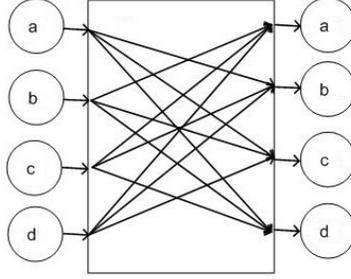


Figure 2.6: A network of 4 nodes.

Table 2.5: Matrix  $A$  of the network in Fig. 2.6

$(a, b)$	$(a, c)$	$(a, d)$	$(b, a)$	$(b, c)$	$(b, d)$	$(c, a)$	$(c, b)$	$(c, d)$	$(d, a)$	$(d, b)$	$(d, c)$
1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	1	0	0	1	0	0	1	0	0
1	0	0	0	0	0	0	1	0	0	1	0
0	1	0	0	1	0	0	0	0	0	0	1

### 2.2.3 Maximum likelihood parameter estimation

Recall that both Vardi's, and Vanderbei and Iannone's model led to similar sets of under-determined linear equations of the form  $\mathbf{V} = \mathbf{A}\mathbf{U}$ . In this section, we study the maximum likelihood parameter estimation approach for estimating  $\lambda$  from  $\mathbf{V} = \mathbf{A}\mathbf{U}$  given several independent realizations of  $\mathbf{V}$ . The approach was implemented using the EM algorithm in [4]. We first address the identifiability issue and then discuss the details of the EM algorithm.

#### Identifiability

A parametric model is said to be identifiable if distinct parameter values imply distinct probability density function values for almost each observation. Vardi [2] showed that the unknown source-destination traffic rates  $\lambda$  are identifiable provided that  $A$  does not contain

duplicate columns or zero columns. Duplicate columns correspond to identical source-destination pairs which need not be distinguished. Also, a zero column in  $A$  corresponds to a useless source-destination pair which can be simply deleted. Clearly, the two conditions are satisfied in any practical scenarios, and the models developed in [2] and [4] are both identifiable.

### The EM algorithm

The EM approach was developed by Vanderbei and Iannone in [4] for estimating  $\lambda$  from  $K$  independent realizations of  $\mathbf{V}$ . Let  $\{v_i^{(1)}, \dots, v_i^{(K)}\}$  and  $\{u_j^{(1)}, \dots, u_j^{(K)}\}$  denote  $K$  statistically independent realizations of  $V_i$  and  $U_j$ , respectively. For  $k = 1, \dots, K$ , let  $\mathbf{v}^{(k)} = (v_1^{(k)}, \dots, v_q^{(k)})'$  and  $\mathbf{u}^{(k)} = (u_1^{(k)}, \dots, u_c^{(k)})'$ . Thus, for  $k = 1, \dots, K$ ,  $\mathbf{v}^{(k)} = A\mathbf{u}^{(k)}$ . Let  $\mathbf{v}_1^K = \{\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(K)}\}$ . We denote the associated density by  $p_\lambda(\cdot)$  and an expected value with respect to  $p_\lambda(\cdot)$  by  $E_\lambda\{\cdot\}$ . The EM algorithm requires evaluation of the log-likelihood of the complete statistics  $\mathbf{U}$ . Let the  $c \times 1$  column vector  $\lambda^{(\iota)} = (\lambda_1^{(\iota)}, \dots, \lambda_c^{(\iota)})'$  denote the parameter estimate at the end of the  $\iota$ -th iteration. The algorithm starts from some initial estimate of  $\lambda$ , say  $\lambda^{(0)}$ , and is terminated when  $\sum_j (\lambda_j^{(\iota)} - \lambda_j^{(\iota+1)})^2$  falls below a threshold. Another criterion for EM is that the relative difference of the log-likelihoods from two consecutive iterations falls below a threshold. The EM auxiliary function is given by,

$$\begin{aligned} Q(\lambda, \lambda^{(\iota)}) &= E_{\lambda^{(\iota)}}\{\log p_\lambda(\mathbf{u}) \mid \mathbf{v}_1^K\} \\ &= \sum_{j=1}^c [-\lambda_j + E_{\lambda^{(\iota)}}\{U_j \mid \mathbf{v}_1^K\} \log \lambda_j - E_{\lambda^{(\iota)}}\{\log U_j! \mid \mathbf{v}_1^K\}] \end{aligned} \quad (2.6)$$

A new parameter estimate at the end of the  $(\iota + 1)$ -th iteration,  $\lambda^{(\iota+1)}$ , is obtained by maximizing  $Q(\lambda, \lambda^{(\iota)})$  with respect to  $\lambda$ . Hence,

$$\lambda^{(\iota+1)} = E_{\lambda^{(\iota)}}\{\mathbf{U} \mid \mathbf{v}_1^K\} = \frac{1}{K} \sum_{k=1}^K E_{\lambda^{(\iota)}}\{\mathbf{U} \mid \mathbf{v}^{(k)}\}, \quad (2.7)$$

which follows from independence of  $\{\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(K)}\}$ . The conditional mean estimate  $E_{\lambda^{(\iota)}}\{\mathbf{U} \mid \mathbf{v}^{(k)}\}$  in (2.7) is given by,

$$E_{\lambda^{(\iota)}}\{\mathbf{U} \mid \mathbf{v}^{(k)}\} = \sum_{\mathbf{u}: \mathbf{v}^{(k)} = A\mathbf{u}} \mathbf{u} \frac{p_{\lambda^{(\iota)}}(\mathbf{u})}{\sum_{\mathbf{u}: \mathbf{v}^{(k)} = A\mathbf{u}} p_{\lambda^{(\iota)}}(\mathbf{u})}. \quad (2.8)$$

Evaluation of (2.8) requires finding all possible solutions of under-determined set of equations  $\mathbf{v}^{(k)} = A\mathbf{u}$  which is not a feasible task. To diminish the computational difficulty of evaluation of (2.8), [4] proposed to evaluate (2.8) by assuming that  $\{\mathbf{U}(t), t = 1, 2, \dots\}$  evolves as a stationary ergodic Markov chain with uniform stationary distribution. A realization of that Markov chain, which satisfies  $\mathbf{v}^{(k)} = A\mathbf{u}(t)$  for each  $t$ , was generated recursively. By using this realization, the conditional expectation in (2.8) is evaluated by,

$$E_{\lambda^{(\iota)}}\{\mathbf{U} \mid \mathbf{v}^{(k)}\} = \lim_{T \rightarrow \infty} \frac{\frac{1}{T} \sum_{t=1}^T \mathbf{u}(t) P_{\lambda^{(\iota)}}(\mathbf{u}(t))}{\frac{1}{T} \sum_{t=1}^T P_{\lambda^{(\iota)}}(\mathbf{u}(t))}. \quad (2.9)$$

Exact evaluation of (2.9) is not computationally feasible. Hence, a large enough value of  $T$  is chosen in (2.9). A recursive approach was proposed to find  $\{\mathbf{u}(t); t = 1, \dots, T\}$  in [4]. We first require to find  $\mathbf{u}(1) = (u_1(1), \dots, u_c(1))'$ . The scheme developed by Vanderbei and Iannone to find  $\mathbf{u}(1)$  was described in [4]. An alternative approach developed in [1] may be also utilized to find  $\mathbf{u}(1)$ . The procedure is based on the following argument. The routing matrix  $A$  is full rank in any practical network tomography problem, since otherwise some measurements (rows) are redundant and can be excluded. Hence, in the network model  $\mathbf{V} = A\mathbf{U}$ , the columns of  $A$  can be reordered such that  $A = [A_c, A_d]$  where  $A_c$  is a  $q \times q$

non-singular matrix. Similarly, from reordering the entries of  $\mathbf{U}$ , we have  $\mathbf{U}' = (\mathbf{U}'_c, \mathbf{U}'_d)$ .

Hence,

$$\mathbf{U}_c = A_c^{-1}(\mathbf{V} - A_d \mathbf{U}_d). \quad (2.10)$$

Then,  $\mathbf{u}(1)$  may be found as follows. First, the entries of  $\mathbf{u}_d$  are chosen randomly on  $[0, \max_j v_j]$ . Next,  $\mathbf{u}_c$  is obtained following (2.10). If  $\mathbf{u}_c$  contains any negative entries, a new  $\mathbf{u}_d$  is chosen randomly and the process will continue, otherwise the resulting  $\mathbf{u}$  will be chosen as  $\mathbf{u}(1)$ . Let  $\mathbf{u}(t) = (u_1(t), \dots, u_c(t))'$  for  $t = 2, \dots, T$ . Next, given  $\mathbf{u}(t)$ , a random walk was defined to find  $\mathbf{u}(t+1)$  in [4], recursively as follows. Choose four distinct nodes  $i_1, i_2, j_1, j_2 \in \{1, \dots, N\}$  at random. Let  $l_1 = (i_1, j_1), l_2 = (i_1, j_2), l_3 = (i_2, j_2), l_4 = (i_2, j_1)$ . Select a random integer  $\zeta$  between  $-\min\{u_{l_1}(t), u_{l_3}(t)\}$  and  $\min\{u_{l_2}(t), u_{l_4}(t)\}$ . For  $t = 1, \dots, T-1$ ,  $\mathbf{u}(t+1)$  is given by

$$\mathbf{u}(t+1) = \mathbf{u}(t) + \zeta \mathbf{1}_{l_1} + \zeta \mathbf{1}_{l_3} - \zeta \mathbf{1}_{l_2} - \zeta \mathbf{1}_{l_4} \quad (2.11)$$

where  $\mathbf{1}_i$  is a column vector of an appropriate dimension with a one in its  $i$ -th component and zero elsewhere.

#### 2.2.4 Vardi's moment matching approach

Vardi [2] proposed a moment matching approach for estimating  $\lambda$  from  $\mathbf{v}_1^K$  to bypass the computational difficulty of implementing (2.8), as follows. The EM algorithm (2.7) was also considered by Vardi [2, Eq. (9)]. The gradient equation of the likelihood function of  $\mathbf{V}$  is given by,

$$\lambda = \frac{1}{K} \sum_{k=1}^K E_{\lambda}\{\mathbf{U} \mid \mathbf{v}^{(k)}\}, \quad (2.12)$$

where  $\lambda$  is a fixed point of the EM algorithm and a stationary point of the likelihood of the observable statistics  $\mathbf{V}$ . Let  $\eta = (\eta_1, \dots, \eta_q)'$  denote a  $q \times 1$  column vector where  $\eta_i = E_{\lambda}\{V_i\} = \sum_{j=1}^c a_{ij} \lambda_j$ . Hence,  $\eta = A\lambda$ . Let the  $q \times 1$  column vector  $\hat{\eta} = (\hat{\eta}_1, \dots, \hat{\eta}_q)'$  denote the sample average of  $\mathbf{V}$  given by

$$\hat{\eta} = \frac{1}{K} \sum_{k=1}^K \mathbf{v}^{(k)}. \quad (2.13)$$

Multiplying both sides of (2.12) by  $A$  shows that if  $\lambda$  is a fix point of the EM algorithm, then

$$A\lambda = \frac{1}{K} \sum_{k=1}^K E_{\lambda}\{A\mathbf{U} \mid \mathbf{v}^{(k)}\} = \frac{1}{K} \sum_{k=1}^K \mathbf{v}^{(k)} = \hat{\eta}. \quad (2.14)$$

Thus, the EM algorithm (2.7) attempts to solve  $\hat{\eta} = A\lambda$ . This under-determined system of linear equations with positivity constraints  $\{\lambda_j \geq 0; j = 1, \dots, c\}$  is referred to as a LININPOS problem [2]. Vardi proposed a moment matching approach for estimating  $\lambda$  which relies on a so-called canonical EM iteration for solving LININPOS problems. The iteration was first introduced and discussed by Shepp and Vardi [5] for the positron emission tomography (PET) problem under independence assumptions of  $\{v_1^{(k)}, \dots, v_q^{(k)}\}$  for  $k = 1, \dots, K$ . The independence assumptions hold for PET problem. The EM iterate used by Vardi [2, Eq. (26)] for solving  $A\lambda = \hat{\eta}$  is given by, for  $l = 1, \dots, c$ ,

$$\lambda_l^{(\iota+1)} = \lambda_l^{(\iota)} \frac{1}{\sum_{i=1}^q a_{il}} \sum_{i=1}^q \frac{a_{il} \hat{\eta}_i}{\sum_{h=1}^c a_{ih} \lambda_h^{(\iota)}}. \quad (2.15)$$

Another justification for iteration (2.15) can be found in [26]. This iteration has been widely used in other applications such as image deblurring, see e.g., [26] and [27], and emission tomography, see e.g., [5], [28] and [29].

Vardi [2] then proposed to estimate  $\lambda$  by matching both the first and second moments of  $\mathbf{V}$  to their sample values. Let  $\psi = \{\psi_{il}; i = 1, \dots, q; l = 1, \dots, q\}$  denote a  $q \times q$  covariance matrix where  $\psi_{il} = E_\lambda\{V_i V_l\} - E_\lambda\{V_i\}E_\lambda\{V_l\}$ . For  $i \neq l$ ,  $\psi_{il}$  is given by,

$$\psi_{il} = \sum_{j=1}^c a_{ij} a_{lj} \lambda_j, \quad (2.16)$$

For  $i = 1, \dots, q$ ,  $\psi_{ii}$  is given by

$$\psi_{ii} = \sum_{j=1}^c a_{ij} \lambda_j. \quad (2.17)$$

Let  $\hat{\psi} = \{\hat{\psi}_{il}; i = 1, \dots, q; l = 1, \dots, q\}$  denote the sample covariance matrix where,

$$\hat{\psi}_{il} = \frac{1}{K} \sum_{k=1}^K v_i^{(k)} v_l^{(k)} - \frac{1}{K^2} \sum_{k=1}^K v_i^{(k)} \sum_{k=1}^K v_l^{(k)}. \quad (2.18)$$

For  $i = 1, \dots, q; j = i, \dots, q$ , elements of the sample covariance matrix,  $\hat{\psi}$ , are ordered lexicographically, and arranged into an  $(1+q)q/2 \times 1$  vector denoted by  $\hat{\phi} = \{\hat{\phi}_j\}$ . Let  $B = \{b_{ij}; i = 1, \dots, q(q+1)/2; j = 1, \dots, c\}$  denote an  $(1+q)q/2 \times c$  matrix with rows

given by element-wise product of rows of  $A$  ordered to match the indexing of  $\hat{\phi}$ . Thus, matching the sample mean and covariance of  $\mathbf{V}$  to their theoretical values, the following matrix equation should be solved,

$$\begin{pmatrix} \hat{\eta} \\ \hat{\phi} \end{pmatrix} = \begin{pmatrix} A \\ B \end{pmatrix} \lambda. \quad (2.19)$$

Some equations in (2.19) are infeasible. For example when  $\sum_l a_{il}a_{jl} > 0$  while  $\hat{\psi}_{ij} < 0$  or when  $\hat{\psi}_{ij} \neq 0$  while  $\sum_l a_{il}a_{jl} = 0$ . Such moment equations are removed from (2.19). Some equations in (2.19) are inconsistent. For example  $\hat{\eta}_i$  and  $\hat{\psi}_{ii}$  express the mean and variance, respectively, which are the same under Poisson model. Vardi let the “geometry of Kullback-Leibler distance” determine the best trade-off between all these inconsistent equations. An EM iterative solution to (2.19) could be attempted similarly to the solution

of (2.14) with the appropriate substitution of  $A$  by  $\begin{pmatrix} A \\ B \end{pmatrix}$  and  $\hat{\eta}$  by  $\begin{pmatrix} \hat{\eta} \\ \hat{\psi} \end{pmatrix}$ .

The EM iterate used by Vardi [2, Eq. (28)] to solve (2.19) for  $l = 1, \dots, c$  is given by,

$$\lambda_l^{(\iota+1)} = \lambda_l^{(\iota)} \frac{1}{\sum_{i=1}^q a_{il}} \sum_{i=1}^q \frac{a_{il} \hat{\eta}_i}{\sum_{h=1}^c a_{ih} \lambda_h^{(\iota)}} + \lambda_l^{(\iota)} \frac{1}{\sum_{i=1}^{(q+1)q/2} b_{il}} \sum_{i=1}^{(q+1)q/2} \frac{b_{il} \hat{\phi}_i}{\sum_{h=1}^c b_{ih} \lambda_h^{(\iota)}}. \quad (2.20)$$

### 2.2.5 Tebaldi and West Bayesian approach

Tebaldi and West [1] studied the rate estimation problem of [2], and developed a Bayesian solution for estimation of the source-destination traffic rates from a single realization of  $\mathbf{V}$ . In this approach, the source-destination rates were assumed to be statistically independent random variables drawn from specified marginal prior distributions. Let  $\mathbf{\Lambda} = (\Lambda_1, \dots, \Lambda_c)'$ .

The goal of [1] was to compute the posterior distributions which are the conditional probability of  $\mathbf{U} = \mathbf{u}$  given  $\mathbf{V} = \mathbf{v}$  and the conditional probability of  $\mathbf{\Lambda} = \lambda$  given  $\mathbf{V} = \mathbf{v}$ . The posterior distributions are used for inference on  $\mathbf{U}$  and the associated source-destination traffic rates. Since posterior computations are analytically difficult, iterative Markov Chain Monte Carlo (MCMC) simulation methods were developed and implemented in [1]. It is not clear that if the measurements are taken at multiple periods, i.e.,  $K > 1$ , how the sufficient statistics can be brought into the analysis. Thus, whereas the Bayesian approach is sensitive to the prior assumption on the distribution of  $\mathbf{\Lambda}$ , it seems more applicable to networks where prior knowledge of  $\mathbf{\Lambda}$  is available and repeated measurements are hard to achieve. The work of [1] was mainly concerned with the tomography problem arising in road networks.

### 2.2.6 Other related works

In this section, we briefly address some other related researches to traffic network tomography. In [30], the traffic over each source-destination pair was modeled by a normal random variable with the variance be the mean raised to a constant power, and the time-varying source-destination traffic rates were estimated from link counts in a network with deterministic routing regime. In [30], a moving window was used to measure the aggregated link counts within a certain time period, and an EM algorithm was developed to compute the maximum likelihood estimation (MLE) of the source-destination rates.

In [31], traffic over the  $j = (j_1, j_2)$ -th pair was assumed constant and proportional to the product of the traffic count originated from  $j_1$  and the traffic count destined at  $j_2$ . This model was referred to as the *gravity model* [31]. An information-theoretic approach was then developed in [31] for estimation of the source-destination traffic counts. The rate estimation problem of [2] was also studied in [32] where a pseudo maximum likelihood solution was developed. The main idea in [32] was to construct independent sub-problems by choosing pairs of rows in  $A$ .

## 2.3 Other aspects of network tomography

Other aspects of network tomography include link loss rate estimation from source-destination loss measurements and topology identification from source destination measurements. The study of link loss rate from source-destination loss measurements was performed in, for example, [24], [33], [34] and [35]. In [24], source-destination loss measurements were used to identify the links with large loss rates. In [35], a striped unicast probing experiment was described to imitate multicast probing. Link losses for packets of the same striped were assumed correlated, and link loss rates were estimated in [35]. In [33] and [34], link losses are described by Bernoulli processes, and multicast probes were utilized. A maximum likelihood estimator was developed in [33] to estimate link loss rates. An EM-based solution to the link loss rate estimation problem of [33] was developed in [34] where the source-destination loss measurements were missing at some of the destinations. Another aspect of network tomography, namely, topology identification from source-destination measurements was studied in [36], [37], [38], [39] and [23]. In [36], a semi-randomized probing technique was developed to identify the topology of a multiple source-multiple destination network. In [37], the source-destination loss measurements were utilized to infer the underlying topology. In [23], source-destination delay measurements were used, and the underlying topology of the network was then identified using an MCMC procedure. The topology of the network is assumed known throughout this thesis.

## Chapter 3: Background on bivariate Markov chains

In this chapter, we review some aspects of the theory of bivariate Markov chains relevant to our study. In [7], a detailed review of bivariate Markov processes and their estimation techniques was provided.

### 3.1 Continuous-time Bivariate Markov chain

Let  $Z = \{Z(t), t \geq 0\}$  denote a continuous-time homogeneous bivariate Markov chain. The bivariate Markov chain comprises a pair of random processes, say  $Z = (X, S)$ . The two random processes  $X = \{X(t), t \geq 0\}$  and  $S = \{S(t), t \geq 0\}$  are jointly Markov while neither of them is necessarily Markov. The  $S$  process is an underlying process that controls the statistical properties such as the sojourn time in each state of the  $X$ -chain. The bivariate Markov chain  $Z$  jumps whenever either  $S$  or  $X$  jumps, or  $X$  and  $S$  jump simultaneously. We denote the state space of  $X$  by  $\mathbb{X} = \{1, \dots, d\}$  and the state space of  $S$  by  $\mathbb{S} = \{1, 2, \dots, r\}$ , for some finite  $d$  and  $r$ . The bivariate Markov chain  $Z$  takes values in  $\mathbb{Z} = \mathbb{X} \times \mathbb{S} = \{(a, i); a \in \mathbb{X}, i \in \mathbb{S}\}$ . The joint states  $\{(a, i)\}$  are assumed to be ordered lexicographically. We use the notation  $P$  to denote a probability measure. For  $(b, j) \neq (a, i)$ , let

$$g_{ab}(ij) = \lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} P(Z(t + \epsilon) = (b, j) \mid Z(t) = (a, i)), \quad (3.1)$$

and let  $g_{aa}(ii) = -\sum_{(b,j) \neq (a,i)} g_{ab}(ij)$ . The generator of  $Z$  is given by the matrix  $G = \{g_{ab}(ij); a, b \in \mathbb{X}, i, j \in \mathbb{S}\}$ . The generator may be expressed as a block matrix  $G = \{G_{ab}; a, b \in \mathbb{X}\}$  where  $G_{ab} = \{g_{ab}(ij); i, j \in \mathbb{S}\}$  is an  $r \times r$  matrix. The generator matrix  $G$  and the sub-matrices  $\{G_{aa}, a \in \mathbb{X}\}$  are assumed to be irreducible. The off-diagonal

elements of  $G$  are non-negative while the diagonal elements of  $G$  are non-positive. The entries of  $G$  are assumed finite throughout this thesis. For a sufficiently small time,  $t$ , we have [40]

$$P(Z(t) = (b, j) \mid Z(0) = (a, i)) = \begin{cases} g_{ab}(ij)t + o(t), & (a, i) \neq (b, j) \\ 1 + g_{aa}(ii)t + o(t), & (a, i) = (b, j). \end{cases} \quad (3.2)$$

Let  $P_t$  denote the transition probability matrix of  $Z$ , i.e.,  $P_t = \{P(Z(t) = (b, j) \mid Z(0) = (a, i)), (a, i) \neq (b, j)\}$  for some  $t > 0$ . The transition matrix  $P_t$  and the generator of the bivariate Markov chain satisfy the following set of equations,

$$P_t = e^{Gt}. \quad (3.3)$$

The matrix  $P_t$  is continuous at  $t = 0$  and

$$\lim_{t \rightarrow 0} P_t = I, \quad (3.4)$$

where  $I$  is the identity matrix. The sojourn time of  $Z$  in state  $(a, i) \in \mathbb{Z}$  is denoted by  $\Delta T_{ai}$  and has exponential density with mean of  $-1/g_{aa}(ii)$ , i.e.,

$$P(\Delta T_{ai} > \tau \mid Z(0) = (a, i)) = e^{g_{aa}(ii)\tau}, \quad \tau \geq 0. \quad (3.5)$$

When the bivariate Markov chain jumps from state  $(a, i)$ , the probability that it will jump to state  $(b, j)$  is given by

$$-g_{ab}(ij)/g_{aa}(ii). \quad (3.6)$$

The generator  $G$  can be partitioned as follows. For each  $a \in \mathbb{X}$ , let the  $S$ -chain take values in  $\mathbb{S}_a = \{1, \dots, r_a\}$  where  $r_a$  is known. Then  $\{G_{ab}; a, b \in \mathbb{X}\}$ , where  $G_{ab} = \{g_{ab}(ij); i \in \mathbb{S}_a, j \in \mathbb{S}_b\}$  is an  $r_a \times r_b$  matrix. The state space of the  $Z$ -chain is given by

$$\mathbb{Z} = \bigcup_{a=1}^d \{(a, i), i \in \mathbb{S}_a\}. \quad (3.7)$$

Recall that neither  $X$  nor  $S$  is necessarily Markov. The  $S$ -chain is Markov if and only if  $r_a$  is independent of  $a$  for  $a \in \mathbb{X}$ , and [41]

$$Q = \sum_{b=1}^d G_{ab} \quad (3.8)$$

Suppose that the process  $Z$  is sampled at the jump points of  $X$ . Let  $T^k$  denote the time of  $(k+1)$ st jump of the  $X$  process for  $k = 0, 1, \dots$ . Let  $X_k = X(T^k)$ ,  $S_k = S(T^k)$  and  $Z_k = (X_k, S_k)$ . The sojourn time of  $X$  in state  $X_{k-1}$  is denoted by  $T_k$ , and  $T_k = T^k - T^{k-1}$  for  $k = 1, \dots$ . We assume that the first jump of  $X$  occurs at  $t = 0$ . Hence,  $T_0 = 0$ . The realizations of  $T^k$ ,  $T_k$  and  $Z_k$  are denoted by  $t^k$ ,  $t_k$  and  $z_k$ , respectively. We refer to the sequence  $\{Z_k\}$  as the *sampled bivariate Markov chain*. Let  $\nu_{ai}(\phi) = P_\phi(Z_0 = (a, i))$  for  $a \in \mathbb{X}, i \in \mathbb{S}$ . Define the row vector  $\nu_a(\phi) = (\nu_{a1}(\phi), \nu_{a2}(\phi), \dots, \nu_{ar}(\phi))$  for  $a \in \mathbb{X}$ , and let  $\nu(\phi) = (\nu_1(\phi), \dots, \nu_d(\phi))$  denote the initial distribution of  $Z$ . The parameter of the bivariate Markov chain is denoted by  $\phi$  and comprises the generator matrix  $G$  and the initial distribution  $\nu(\phi)$ . Let  $P_\phi$  denote a probability measure of the process  $Z$ . The Markov renewal property of the bivariate Markov chain applies to the jumps of  $X$ . This

property implies that

$$P_\phi(T_k \leq t, Z_k = z_k \mid T_{k-1} = t_{k-1}, Z_{k-1} = z_{k-1}, \dots, T_1 = t_1, Z_1 = z_1, Z_0 = z_0) = P_\phi(T_k \leq t, Z_k = z_k \mid Z_{k-1} = z_{k-1}). \quad (3.9)$$

Define  $F_{ij}^{ab}(t; \phi) = P_\phi(T_k \leq t, Z_k = (b, j) \mid Z_{k-1} = (a, i))$  for  $a \neq b$ . By homogeneity of bivariate Markov chain  $F_{ij}^{ab}(t; \phi)$  is independent of  $k$ . Let  $f_{ij}^{ab}(t; \phi)$  denote the transition density which is obtained from differentiation of  $F_{ij}^{ab}(t; \phi)$  with respect to  $t$ , i.e.,

$$f_{ij}^{ab}(t; \phi) = \frac{\partial}{\partial t} P_\phi(T_1 \leq t, Z_1 = (b, j) \mid Z_0 = (a, i)), \quad (3.10)$$

This density is given by [42]

$$f_{ij}^{ab}(t; \phi) = [e^{G_{aa}t} G_{ab}]_{ij}, \quad a \neq b, \quad t \geq 0. \quad (3.11)$$

Let  $f^{ab}(t) = \{f_{ij}^{ab}(t); i, j \in \mathbb{S}\}$  denote the  $r \times r$  transition density matrix. For  $a \neq b$ ,

$$f^{ab}(t; \phi) = e^{G_{aa}t} G_{ab}, \quad a \neq b, \quad t \geq 0. \quad (3.12)$$

The likelihood function of the sample path of  $X$ -chain, i.e.,  $\{X(t), t \in [0, t^n]\}$ , is given by

$$\begin{aligned}
p_\phi(x(t), t \in [0, t^n]) &= p_\phi(x_0, t_1, x_1, \dots, t_n, x_n) \\
&= \nu_{x_0}(\phi) \left\{ \prod_{l=1}^n f^{x_{l-1}x_l}(t_l; \phi) \right\} \mathbf{1},
\end{aligned} \tag{3.13}$$

where  $\mathbf{1}$  denotes a column vector of all ones. We next introduce the forward and backward recursions which simplify the calculation of the likelihood function (3.13) significantly. For  $k = 1, \dots, n$ , define

$$L(k; \phi) = \nu_{x_0}(\phi) \prod_{l=1}^k f^{x_{l-1}x_l}(t_l; \phi), \tag{3.14}$$

to denote the forward density. Define

$$R(k; \phi) = \prod_{l=k}^n f^{x_{l-1}x_l}(t_l; \phi) \mathbf{1}, \tag{3.15}$$

to denote the backward density. For  $k = 0$ ,  $L(0; \phi) = \nu_{x_0}(\phi)$ , and for  $k = n+1$ ,  $R(n+1; \phi) =$

$\mathbf{1}$ . The forward density can be calculated recursively as in the following

$$L(k; \phi) = L(k-1; \phi) f^{x_{k-1}x_k}(t_k; \phi). \tag{3.16}$$

The recursion for the backward density is also given by

$$R(k; \phi) = f^{x_{k-1}x_k}(t_k; \phi) R(k+1; \phi). \tag{3.17}$$

Using (3.16) and (3.17) and for  $k \in \{1, \dots, n+1\}$ , the likelihood function (3.13) is given by

$$p_\phi(x(t), t \in [0, t^n]) = L(k-1; \phi)R(k; \phi) \quad (3.18)$$

The numerical stability of the forward and backward recursions is improved by recursive scaling as follows [7, Section 3.4]. Define  $c_0 = \nu_{x_0}(\phi)\mathbf{1}$  and  $\tilde{L}(0; \phi) = \nu_{x_0}(\phi)/c_0$ . For  $k = 1, \dots, n$ , the scaled forward recursion is defined as

$$\tilde{L}(k; \phi) = \frac{\tilde{L}(k-1; \phi)f^{x_{k-1}x_k}(t_k; \phi)}{c_k} \quad (3.19)$$

where  $c_k = \tilde{L}(k-1; \phi)f^{x_{k-1}x_k}(t_k; \phi)\mathbf{1}$ . Let  $\tilde{R}(n+1; \phi) = \mathbf{1}$ . For  $k = n, n-1, \dots, 1$ , the scaled backward recursion is defined as

$$\tilde{R}(k; \phi) = \frac{f^{x_{k-1}x_k}(t_k; \phi)\tilde{R}(k+1; \phi)}{c_k} \quad (3.20)$$

and the likelihood (3.18) is given by

$$p_\phi(x(t), t \in [0, t^n]) = c_0 \prod_{k=1}^n c_k. \quad (3.21)$$

We next calculate the transition probabilities of the sampled bivariate Markov chain. For  $a \neq b$ , the transition probabilities of  $\{Z_k\}$  is given by

$$\begin{aligned}
P_\phi(Z_k = (b, j) \mid Z_{k-1} = (a, i)) &= \left[ \int_0^\infty f^{ab}(t) dt \right]_{ij} \\
&= [-G_{aa}^{-1} G_{ab}]_{ij},
\end{aligned} \tag{3.22}$$

and the transition matrix of the sampled bivariate Markov chain is defined by the block matrix  $D = \{D_{ab}; a, b \in \mathbb{X}\}$  where,

$$D_{ab} = \begin{cases} -G_{aa}^{-1} G_{ab}, & a \neq b \\ \mathbf{0}, & a = b. \end{cases} \tag{3.23}$$

The state  $(b, j)$  is said to be recurrent if the probability of eventual return is one. The sampled bivariate Markov chain has one closed set of recurrent, possibly periodic, states, while the remaining states are transient [43, Lemma 3]. Hence, the transition matrix  $D$  has a unique stationary distribution with zero entries for the transient states. The stationary distribution of  $D$  coincides with the initial distribution  $\nu(\phi)$  when  $\nu(\phi) = \nu(\phi)D$ , and the sample bivariate Markov chain is said to be stationary.

Let  $\nu_{ai}^k(\phi) = P_\phi(Z_k = (a, i))$  for  $a \in \mathbb{X}, i \in \mathbb{S}, k = 0, 1, \dots$ . Define the row vector  $\nu_a^k(\phi) = (\nu_{a1}^k(\phi), \nu_{a2}^k(\phi), \dots, \nu_{ar}^k(\phi))$  for  $a \in \mathbb{X}$ , and let  $\nu^k(\phi) = (\nu_1^k(\phi), \dots, \nu_d^k(\phi))$ . Note that  $\nu_{ai}^0(\phi) = \nu_{ai}(\phi)$ . For  $a \in \mathbb{X}$ , we have

$$\nu_b^k(\phi) = \sum_{a=1}^d \nu_a^{k-1}(\phi) D_{ab}. \tag{3.24}$$

If the sample bivariate Markov chain is initialized with its stationary distribution,  $\nu^k(\phi)$  does not depend on  $k$  and satisfies  $\nu(\phi) = \nu(\phi)D$ . For simplicity throughout this thesis, we

suppress the dependence of  $\nu_{ai}^k(\phi)$ ,  $\nu_a^k(\phi)$ , and  $\nu^k(\phi)$  on  $\phi$ , and we refer to the parameter of the bivariate Markov chain as  $\phi = (G, \nu)$ .

## Chapter 4: Delay Network Tomography using a Partially Observable Bivariate Markov Chain

In this chapter, we study the problem of estimating link delay density from source-destination measurements. We model the total delay on each link which is caused by queuing, propagation, processing, and transmission delays. We propose a partially observable bivariate Markov chain (BMC) to model the traffic over an unstructured network with random routing regime as well as a tree-structured network with deterministic routing. The properties of the proposed model and the algorithm for estimating its parameter are discussed. Our presentation in this Chapter follows the recently submitted paper [44].

### 4.1 Partially observable bivariate Markov chain model

We propose to model the traffic over an unstructured network with random routing regime as a bivariate Markov chain  $Z = (X, S)$ . In this model, the states of the  $X$ -chain represent the nodes of the network. A subset of the state space  $\mathbb{X} = \{1, \dots, d\}$  denoted by  $\mathbb{X}_1 = \{1, \dots, d_1\}$  represents the source states. The set of the remaining states denoted by  $\mathbb{X}_2 = \{d_1 + 1, \dots, d\}$  represents the destinations. Clearly,  $\mathbb{X} = \mathbb{X}_1 \cup \mathbb{X}_2$ . A packet travels from a randomly selected source node towards its destination through random routing regime, and the associated source-destination delay is measured. Upon leaving the destination node, the packet will enter another randomly selected source node and propagate through random routing regime towards its destination. We repeat this process to obtain a set of independent source-destination delays. The underlying  $S$ -chain controls the statistical properties of the  $X$ -chain. In particular, the  $X$ -chain is not set to be Markov, and the sojourn time in each of its states has a phase-type density [7]. Phase-type distributions are rather general and may be used to approximate any desirable sojourn time distribution.

For example, mixture of exponential densities and the sums of exponential densities are particular phase-type distributions. In the proposed model, the two chains  $X$  and  $S$  are hidden, and we only measure source-destination delays. Therefore, we refer to the model as a partially observable bivariate Markov chain.

We partition the generator of the bivariate Markov chain,  $G$ , into the class of source nodes and the class of destination nodes,

$$G = \left( \begin{array}{ccc|ccc} G_{11} & \cdots & G_{1,d_1} & & & \\ \vdots & & \vdots & h_1 & \cdots & h_{d_2} \\ G_{d_1,1} & \cdots & G_{d_1,d_1} & & & \\ \hline & e_1 & & -\xi_1 & & \\ & \vdots & & & \ddots & \\ & e_{d_2} & & & & -\xi_{d_2} \end{array} \right) \quad (4.1)$$

where  $\{G_{ij}, i, j = 1, \dots, d_1\}$  are  $r \times r$  matrices,  $\{h_l, l = 1, \dots, d_2\}$  are  $rd_1 \times 1$  column vectors with  $d_2 = d - d_1$ ,  $\{e_l, l = 1, \dots, d_2\}$  are  $1 \times rd_1$  row vectors, and  $\{\xi_l\}$  are positive scalars. We can also incorporate any knowledge provided by network topology in the structure of the generator. In particular, if there does not exist any physical link connecting node  $a \in \mathbb{X}_1$  to node  $b \in \mathbb{X}$ , then the corresponding sub-matrix  $G_{ab} = 0$ . The generator,  $G$ , can be written as,

$$G = \left( \begin{array}{c|c} H_{cc} & H_{cd} \\ \hline H_{dc} & H_{dd} \end{array} \right) \quad (4.2)$$

where the subscripts  $c$  and  $d$  are associated with source and destination states, respectively. We do not allow the traffic to flow from one destination node to another destination node. Therefore, the sub-matrix  $H_{dd}$  is diagonal. Also, the bivariate Markov chain never starts

from a destination node, and the initial distribution of the sampled bivariate,  $\nu$ , is given by

$$\nu = (\nu_1, \dots, \nu_{d_1}, \mathbf{0}, \dots, \mathbf{0}). \quad (4.3)$$

The sojourn time of the process in the  $i$ -th destination node is an exponential random variable with the mean of  $\frac{1}{\xi_i}$ . We choose sufficiently large  $\{\xi_i\}$ . Hence, the sojourn time in each destination node is negligible, and the process jumps from a destination node to a source node instantaneously. The sojourn time of the process in each state of the  $X$ -chain is phase-type [42]. Let  $\bar{\nu}_{x_0} = \frac{\nu_{x_0}}{\nu_{x_0} \mathbf{1}}$  denote the initial conditional distribution of the  $S$ -chain associated with  $x_0$ , given that  $X_0 = x_0$ . For  $X_0 = x_0$ , by using ((3.12)) and ((3.23)), the phase-type density is given by:

$$\begin{aligned} p_\phi(t | x_0) &= \frac{\sum_{s_0, s_1, x_1} p_\phi(z_1, t | z_0) p_\phi(z_0)}{\int_0^\infty \sum_{s_0, s_1, x_1} p_\phi(z_1, t | z_0) p_\phi(z_0) dt} \\ &= \frac{\sum_{x_1} \nu_{x_0} e^{G_{x_0 x_0} t} G_{x_0 x_1} \mathbf{1}}{\sum_{x_1} \nu_{x_0} D_{x_0 x_1} \mathbf{1}} = -\bar{\nu}_{x_0} e^{G_{x_0 x_0} t} G_{x_0 x_0} \mathbf{1}, \end{aligned} \quad (4.4)$$

where we have used  $\sum_{x_1} D_{x_0 x_1} \mathbf{1} = \mathbf{1}$ . Since  $X$ -chain is not Markov, for  $X_{k+1} = x_{k+1}$ ,  $X_k = x_k$  and some  $k > 0$ , the density of (4.4) can be re-written as,

$$\begin{aligned} p_\phi(t | x_k) &= \frac{\sum_{s_k, s_{k+1}, x_{k+1}} p_\phi(z_{k+1}, t | z_k) p_\phi(z_k)}{\int_0^\infty \sum_{s_k, s_{k+1}, x_{k+1}} p_\phi(z_{k+1}, t | z_k) p_\phi(z_k) dt} \\ &= \frac{\sum_{x_{k+1}} \nu_{x_k}^k e^{G_{x_k x_k} t} G_{x_k x_{k+1}} \mathbf{1}}{\sum_{x_{k+1}} \nu_{x_k}^k D_{x_k x_{k+1}} \mathbf{1}} = -\bar{\nu}_{x_k}^k e^{G_{x_k x_k} t} G_{x_k x_k} \mathbf{1}, \end{aligned} \quad (4.5)$$

where  $\nu_{x_k}^k$  is given by (3.24), and  $\bar{\nu}_{x_k}^k = \frac{\nu_{x_k}^k}{\nu_{x_k}^k \mathbf{1}}$ . According to [45], every sojourn time density

is either phase-type or can be approximated well enough by a phase-type density. The number of phases of the phase-type density in (4.4) and (4.5) equals the number of states in  $\mathbb{S}_{x_0}$ . The estimation of the true phase-type density can be improved by increasing the number of phases of the phase-type density.

Next, we use the parameter of a bivariate Markov chain to derive the source-destination delay density, density of each link delay, and the random routing probabilities. An EM algorithm is then used for estimating the parameter  $\phi$  of the bivariate Markov chain from  $K$  independent source-destination delay measurements. This EM algorithm was motivated by the algorithm derived in [9] for estimation of the parameter of a univariate Markov chain with one absorbing state from  $K$  independent absorbing times. We discuss the details of the EM algorithm in Section 4.2.

Let  $Y = \{Y_1, Y_2, \dots, Y_K\}$  denote the  $K$  independent source-destination measurements, and define  $\mu = (\nu_1, \dots, \nu_{d_1})$ . We denote an infinitesimal interval of  $Y_k$  by  $dy$ . By using (3.12), for  $a \in \mathbb{X}_1$  and  $b \in \mathbb{X}_2$ , we have,

$$P_\phi(Y_k \in dy, Z(y_k) = (b, j) | Z(0) = (a, i)) = \mathbf{1}'_{ai} e^{H_{cc}y_k} H_{cd} \mathbf{1}_{bj}, \quad (4.6)$$

where  $\mathbf{1}_{ai}$  denotes a column vector of an appropriate dimension with a one in its  $(a, i)$  component and zero elsewhere. Hence,

$$P_\phi(Y_k \in dy, Z(0) = (a, i)) = \nu_{ai} \mathbf{1}'_{ai} e^{H_{cc}y_k} H_{cd} \mathbf{1}, \quad (4.7)$$

and the source-destination delay density is given by,

$$P_\phi(Y_k \in dy) = \mu e^{H_{cc}y_k} H_{cd} \mathbf{1}. \quad (4.8)$$

The likelihood of the source-destination measurements is given by

$$P_\phi(Y_1 \in dy, \dots, Y_K \in dy) = \prod_{k=1}^K \mu e^{H_{cc} y_k} H_{cd} \mathbf{1}. \quad (4.9)$$

The delay on the link connecting node  $a$  to node  $b$  is the transit time of the process  $X$  from state  $a$  to state  $b$  where  $a \neq b$ . For  $X_1 = x_1$  and  $X_0 = x_0$ , the density of delay on the link connecting these two nodes is given by,

$$\begin{aligned} p_\phi(t \mid x_0, x_1) &= \frac{\sum_{s_0, s_1} p_\phi(z_1, t \mid z_0) p_\phi(z_0)}{\int_0^\infty \sum_{s_0, s_1} p_\phi(z_1, t \mid z_0) p_\phi(z_0) dt} \\ &= \frac{\nu_{x_0} e^{G_{x_0 x_0} t} G_{x_0 x_1} \mathbf{1}}{\nu_{x_0} D_{x_0 x_1} \mathbf{1}}, \end{aligned} \quad (4.10)$$

where we have used (3.12) and (3.23). The link delay density (given in (4.10)) is a matrix exponential density. The family of matrix exponential densities is very rich and includes phase-type densities [8]. The random routing probabilities follow from (3.12), (3.23) and (3.24). For  $z_0 = (x_0, s_0)$  and  $z_1 = (x_1, s_1)$ , we have,

$$\begin{aligned} p_\phi(x_1 \mid x_0) &= \frac{\int_0^\infty \sum_{s_0, s_1} p_\phi(z_1, t \mid z_0) p_\phi(z_0) dt}{\sum_{x_1} \int_0^\infty \sum_{s_0, s_1} p_\phi(z_1, t \mid z_0) p_\phi(z_0) dt} \\ &= \frac{\nu_{x_0} D_{x_0 x_1} \mathbf{1}}{\sum_{x_1} \nu_{x_0} D_{x_0 x_1} \mathbf{1}} = \frac{\nu_{x_0} D_{x_0 x_1} \mathbf{1}}{\nu_{x_0} \mathbf{1}}, \end{aligned} \quad (4.11)$$

which is the probability that a packet reaches node  $x_1$  upon leaving node  $x_0$ . Clearly,  $D_{x_0 x_0} = \mathbf{0}$ . Therefore, we do not restrict the summation over  $x_1$  in (4.11) to  $x_1 \neq x_0$ . The probability of each path in the network is given by (3.18) which follows from the forward backward recursions.

## 4.2 Maximum likelihood parameter estimation

In this section, we develop the maximum likelihood parameter estimation approach for estimating the parameter of the continuous-time bivariate Markov chain  $\phi$  from  $K$  independent source-destination delay measurements. The approach is implemented using the EM algorithm. This EM algorithm was first developed in [9] for estimating the parameter of the phase-type density arising in a univariate Markov chain with all but one transient states and one absorbing state. Given  $K$  independent replications of the absorption time, [9] implemented the EM algorithm for estimating the parameter of the Markov chain from which the estimate of the phase-type density follows. The generator of the univariate Markov chain discussed in [9] may be derived by setting  $d_2 = 1$  and  $\xi_1 = 0$  in (4.1) as,

$$G = \left( \begin{array}{ccc|c} G_{11} & \dots & G_{1,d_1} & h_1 \\ \vdots & & \vdots & \\ G_{d_1,1} & \dots & G_{d_1,d_1} & \\ \hline \mathbf{0} & & & 0 \end{array} \right) = \left( \begin{array}{c|c} H_{cc} & H_{cd} \\ \hline \mathbf{0} & 0 \end{array} \right). \quad (4.12)$$

We next address the identifiability issue and discuss the details of the EM algorithm.

### 4.2.1 Identifiability

The proposed bivariate Markov chain model is said to be identifiable if distinct parameter values imply distinct values for the density of almost all source-destination measurements. If this condition does not hold, then the chain is said to be unidentifiable. In general, the density of (4.8) is not identifiable [9]. For example, consider the generator given by [9],

$$G = \left( \begin{array}{c|c} H_{cc} & H_{cd} \\ \hline \mathbf{0} & 0 \end{array} \right) = \left( \begin{array}{cc|c} -\lambda & 0 & \lambda \\ 0 & -\lambda & \lambda \\ \hline 0 & 0 & 0 \end{array} \right). \quad (4.13)$$

By substituting (4.13) in (4.8), the source-destination delay density is given by  $P_\phi(y_k \in dy) = \lambda e^{-\lambda y_k}$  regardless of the choice of the initial distribution of the Markov chain  $\nu$ . However, the approach of fitting the bivariate Markov chain model to  $K$  independent measurements is useful even if the measurements were not generated by the model.

#### 4.2.2 EM algorithm

The EM approach is an iterative algorithm for estimating the parameter of a bivariate Markov chain which aims at achieving the highest value of log-likelihood function of the measurements. The algorithm starts from some initial estimate of the parameter and is terminated when the relative change in consecutive likelihood values is negligible. The parameter of the bivariate Markov chain model comprises the off-diagonal elements of  $H_{cc}$ , the elements of  $H_{cd}$  and the initial distribution of the chain,  $\mu$ . The elements of the diagonal matrix  $H_{dd}$  are chosen to be positive large scalars  $\{\xi_i\}$ . The elements of the sub-matrix  $H_{dc}$  are chosen as follows. By setting  $\{e_i = \xi_i \mu, i = 1, \dots, d_2\}$ , the initial distribution of  $Z$  would remain  $\mu$  after bouncing back from any destination node. The initial parameter estimate, and the estimate of the parameter at the end of the  $\iota$ -th iteration, are denoted by  $\phi_0$  and  $\phi_\iota$ , respectively. Let  $E_{\phi_\iota}$  denote expectation under  $P_{\phi_\iota}$ . We denote the bivariate Markov chain in the interval  $[0, Y_k]$  by  $\tilde{Z}_k = \{Z(t), 0 \leq t \leq Y_k\}$ . The EM procedure requires the likelihood function of the complete statistics of the bivariate Markov chain, i.e.,  $\{(\tilde{Z}_k, Y_k), k = 1, \dots, K\}$ . The estimate of the parameter  $\phi$  at the end of the  $\iota + 1$ -th iteration is given by,

$$\phi_{\iota+1} = \operatorname{argmax}_{\phi} \sum_{k=1}^K E_{\phi_{\iota}} \{ \log p_{\phi}(Y_k, \tilde{Z}_k) \mid Y_k \}, \quad (4.14)$$

where we have used the independence of the measurements. Next, we calculate the likelihood of the complete statistics,  $p_{\phi}(Y_k, \tilde{Z}_k)$ , as given in [46, Theorem 3.1]. For a given  $k$ , let  $Y_k = y_k$ . For  $\{a, b\} \in \mathbb{X}_1$ , let  $M_{ij}^{ab}(y_k)$  denote the number of jumps from state  $(a, i)$  to state  $(b, j)$  in  $[0, y_k]$ , and let  $D_i^a(y_k)$  denote the total sojourn time of the bivariate Markov chain,  $Z$ , spent in state  $(a, i)$  in  $[0, y_k]$ . For  $a \in \mathbb{X}_1$  and  $\bar{l} \in \mathbb{X}_2$ , let  $M_{i\bar{l}}^{a\bar{l}}(y_k)$  denote the number of jumps from state  $(a, i)$  to state  $(\bar{l}, 1)$  in  $[0, y_k]$ . Clearly,  $M_{i\bar{l}}^{a\bar{l}}(y_k)$  is either zero or one. As described earlier, the sojourn time in state  $(\bar{l}, 1)$  is assumed negligible. Define the indicator function

$$\varphi_{ai}(t) = \begin{cases} 1, & Z(t) = (a, i) \\ 0, & \text{otherwise.} \end{cases} \quad (4.15)$$

Hence, the log-likelihood of the complete statistics in  $[0, y_k]$  is given by [46, Theorem 3.1],

$$\begin{aligned} \log p_{\phi}(Y_k, \tilde{Z}_k) &= \sum_{(a,i)} \varphi_{ai}(0) \log \nu_{ai} + \sum_{(a,i)} \sum_{(b,j) \neq (a,i)} M_{ij}^{ab}(y_k) \log g_{ab}(ij) \\ &\quad - \sum_{(a,i)} D_i^a(y_k) \sum_{(b,j) \neq (a,i)} g_{ab}(ij). \end{aligned} \quad (4.16)$$

Given the parameter estimate at the end of the  $\iota$ -th iteration,  $\phi_{\iota}$ , the EM auxiliary function in  $[0, y_k]$  is given by,

$$\begin{aligned}
Q(\phi, \phi_\iota) &= E_{\phi_\iota} \{ \log p_\phi(Y_k, \tilde{Z}_k) \} \\
&= \sum_{(a,i)} E_{\phi_\iota} \{ \varphi_{ai}(0) \mid Y_k = y_k \} \log \nu_{ai} + \sum_{(a,i)} \sum_{(b,j) \neq (a,i)} E_{\phi_\iota} \{ M_{ij}^{ab}(y_k) \mid Y_k = y_k \} \log g_{ab}(ij) \\
&\quad - \sum_{(a,i)} E_{\phi_\iota} \{ D_i^a(y_k) \mid Y_k = y_k \} \sum_{(b,j) \neq (a,i)} g_{ab}(ij). \tag{4.17}
\end{aligned}$$

A new parameter estimate at the end of  $(\iota + 1)$ th iteration including a new estimate of the initial distribution of the bivariate Markov chain, of the off-diagonal elements of  $H_{cc}$ , and of the elements of  $H_{cd}$ , as obtained from (4.14), is given by

$$\hat{\nu}_{ai} = \frac{1}{K} \sum_{k=1}^K E_{\phi_\iota} \{ \varphi_{ai}(0) \mid Y_k = y_k \}, \tag{4.18}$$

$$\hat{g}_{ab}(ij) = \frac{\sum_{k=1}^K E_{\phi_\iota} \{ M_{ij}^{ab}(Y_k) \mid Y_k = y_k \}}{\sum_{k=1}^K E_{\phi_\iota} \{ D_i^a(Y_k) \mid Y_k = y_k \}}, \quad (b, j) \neq (a, i). \tag{4.19}$$

We define the conditional mean estimate of  $M_{ij}^{ab}(y_k)$  and  $D_i^a(y_k)$  as,

$$\begin{aligned}
\hat{M}_{ij}^{ab}(y_k) &= E_{\phi_\iota} \{ M_{ij}^{ab}(Y_k) \mid Y_k = y_k \}, \\
\hat{D}_i^a(y_k) &= E_{\phi_\iota} \{ D_i^a(Y_k) \mid Y_k = y_k \}. \tag{4.20}
\end{aligned}$$

We next use the approach of [9] to evaluate the conditional mean estimates in (4.18) and (4.19). The key to this approach is (3.12).

Using (4.7) and (4.8), the conditional mean estimate of  $\varphi_{ai}(0)$  given  $Y_k = y_k$  can be evaluated as,

$$E_{\phi_\iota}\{\varphi_{ai}(0) \mid Y_k = y_k\} = P_{\phi_\iota}(Z(0) = (a, i) \mid Y_k = y_k) \quad (4.21)$$

$$= \frac{P_{\phi_\iota}(Y_k \in dy, Z(0) = (a, i))}{P_{\phi_\iota}(Y_k \in dy)} = \frac{\nu_{ai} \mathbf{1}'_{ai} e^{H_{cc} y_k} H_{cd} \mathbf{1}}{\mu e^{H_{cc} y_k} H_{cd} \mathbf{1}}. \quad (4.22)$$

The total sojourn time of the bivariate Markov chain in state  $(a, i)$  in  $[0, y_k]$  for  $a \in \mathbb{X}_1$  is given by,

$$D_i^a(y_k) = \int_0^\infty \varphi_{ai}(t) dt = \int_0^{y_k} \varphi_{ai}(t) dt, \quad (4.23)$$

where we have used the fact that for  $t > y_k$ ,  $\varphi_{ai}(t) = 0$ . Hence, the conditional mean estimate of  $D_i^a(y_k)$  given  $Y_k = y_k$  is the integral of the conditional probability of  $Z(t) = (a, i)$  given  $Y_k = y_k$ . This conditional probability is given by,

$$P_{\phi_\iota}(Z(t) = (a, i) \mid Y_k = y_k) = \frac{P_{\phi_\iota}(Y_k \in dy \mid Z(t) = (a, i)) P_{\phi_\iota}(Z(t) = (a, i))}{P_{\phi_\iota}(Y_k \in dy)}, \quad (4.24)$$

where the conditional probability of  $Y_k \in dy$  given  $Z(t) = (a, i)$  follows from (4.7) and is given by  $\mathbf{1}'_{ai} e^{H_{cc}(y_k - t)} H_{cd} \mathbf{1}$ . The probability density of  $Z(t) = (a, i)$  in (4.24) is also a phase-type and is given by

$$P_{\phi_\iota}(Z(t) = (a, i)) = \nu e^{Gt} \mathbf{1}_{ai}. \quad (4.25)$$

Hence,

$$\begin{aligned}
\hat{D}_i^a(y_k) &= \frac{\int_0^{y_k} [\nu e^{Gt} \mathbf{1}_{ai}] [\mathbf{1}'_{ai} e^{H_{cc}(y_k-t)} H_{cd} \mathbf{1}] dt}{\mu e^{H_{cc} y_k} H_{cd} \mathbf{1}} \\
&= \frac{\mathbf{1}'_{ai} [\int_0^{y_k} e^{H_{cc}(y_k-t)} H_{cd} \mathbf{1} \nu e^{Gt} dt] \mathbf{1}_{ai}}{\mu e^{H_{cc} y_k} H_{cd} \mathbf{1}}.
\end{aligned} \tag{4.26}$$

The number of jumps from state  $(a, i)$  to state  $(b, j)$  for  $\{a, b\} \in \mathbb{X}_1$  and  $(a, i) \neq (b, j)$  in  $[0, y_k]$  is given by [9]

$$M_{ij}^{ab}(y_k) = \lim_{\epsilon \rightarrow 0} \sum_{l=0}^{\lfloor y_k/\epsilon \rfloor - 1} \varphi_{ai}(l\epsilon) \varphi_{bj}((l+1)\epsilon), \tag{4.27}$$

where  $\lfloor \cdot \rfloor$  denotes the floor function, and we have used the fact that  $\varphi_{bj}((l+1)\epsilon) = 0$  for  $(l+1)\epsilon \geq y_k$ . From Lebesgue's monotone convergence theorem,  $\hat{M}_{ij}^{ab}(y_k)$  for  $\{a, b\} \in \mathbb{X}_1$  is given by,

$$\begin{aligned}
\hat{M}_{ij}^{ab}(y_k) &= \lim_{\epsilon \rightarrow 0} \sum_{l=0}^{\lfloor y_k/\epsilon \rfloor - 1} P_{\phi_\epsilon}(Z(l\epsilon) = (a, i), Z((l+1)\epsilon) = (b, j) | Y_k \in dy) \\
&= \frac{1}{P_{\phi_\epsilon}(Y_k \in dy)} \lim_{\epsilon \rightarrow 0} \sum_{l=0}^{\lfloor y_k/\epsilon \rfloor - 1} \epsilon [P_{\phi_\epsilon}(Z(l\epsilon) = (a, i)) \\
&\quad \cdot \frac{1}{\epsilon} P_{\phi_\epsilon}(Z((l+1)\epsilon) = (b, j) | Z(l\epsilon) = (a, i)) \cdot P_{\phi_\epsilon}(Y_k \in dy | Z((l+1)\epsilon) = (b, j))] .
\end{aligned} \tag{4.28}$$

The probability of  $Z(l\epsilon) = (a, i)$  follows from (4.25) and is given by  $\nu e^{G(l\epsilon)} \mathbf{1}_{ai}$ . The conditional probability of  $Z((l+1)\epsilon) = (b, j)$  given  $Z(l\epsilon) = (a, i)$  in (4.28) is given by  $\mathbf{1}'_{ai} e^{G\epsilon} \mathbf{1}_{bj}$ . Furthermore, the conditional probability of  $Y_k \in dy$  given  $Z((l+1)\epsilon) = (b, j)$

follows from (4.7) and (4.8) and is given by  $\mathbf{1}'_{bj} e^{H_{cc}(y_k - (l+1)\epsilon)} H_{cd} \mathbf{1}$ .

Using the change of variable  $t = (l+1)\epsilon$  and

$$\lim_{\epsilon \downarrow 0} \frac{e^{G\epsilon} - I}{\epsilon} = G, \quad (4.29)$$

in (4.28), for  $(a, i) \neq (b, j)$  and  $\{a, b\} \in \mathbb{X}_2$ , we have

$$\begin{aligned} \hat{M}_{ij}^{ab}(y_k) &= \frac{\int_0^{y_k} [\nu e^{Gt} \mathbf{1}_{ai}] g_{ab}(ij) [\mathbf{1}'_{bj} e^{H_{cc}(y_k-t)} H_{cd} \mathbf{1}] dt}{\mu e^{H_{cc}y_k} H_{cd} \mathbf{1}} \\ &= \frac{g_{ab}(ij) \mathbf{1}'_{bj} [\int_0^{y_k} e^{H_{cc}(y_k-t)} H_{cd} \mathbf{1} \nu e^{Gt} dt] \mathbf{1}_{ai}}{\mu e^{H_{cc}y_k} H_{cd} \mathbf{1}}. \end{aligned} \quad (4.30)$$

The number of jumps from state  $(a, i)$  to state  $(\bar{l}, 1)$  where  $a \in \mathbb{X}_1$  and  $\bar{l} \in \mathbb{X}_2$  in  $[0, y_k]$  is either 0 or 1 and is given by,

$$M_{i\bar{l}}^{a\bar{l}}(y_k) = \lim_{\epsilon \rightarrow 0} \varphi_{ai}(y_k - \epsilon) \varphi_{\bar{l}1}(y_k). \quad (4.31)$$

The conditional mean estimate of  $M_{i\bar{l}}^{a\bar{l}}$  is given by

$$\begin{aligned} \hat{M}_{i\bar{l}}^{a\bar{l}} &= \lim_{\epsilon \rightarrow 0} P_{\phi_\epsilon}(Z(Y_k - \epsilon) = (a, i), Z(Y_k) = (\bar{l}, 1) \mid Y_k \in dy) \\ &= \frac{1}{P_{\phi_\epsilon}(Y_k \in dy)} \lim_{\epsilon \rightarrow 0} [P_{\phi_\epsilon}(Z(y_k - \epsilon) = (a, i)) \\ &\quad \cdot P_{\phi_\epsilon}(Z(y_k) = (\bar{l}, 1), Y_k \in dy \mid Z(y_k - \epsilon) = (a, i))] . \end{aligned} \quad (4.32)$$

The probability of  $Z(y_k - \epsilon) = (a, i)$  in (4.32) follows from (4.25) and is given by  $\nu e^{G(y_k - \epsilon)} \mathbf{1}_{ai}$ . The conditional probability of  $P_{\phi_\epsilon}(Z(y_k) = (\bar{l}, 1), Y_k \in dy \mid Z(y_k - \epsilon) = (a, i))$

in (4.32) is given by  $\mathbf{1}'_{ai}e^{H_{cc}\epsilon}h_l$ . Note that for  $a \in \mathbb{X}_1$ ,  $i = 1, \dots, r$  and  $\bar{l} \in \mathbb{X}_2$  where  $l = \bar{l} - d_1$ ,  $g_{a\bar{l}}(i\mathbf{1}) = h_l((a-1)r+i)$ . Hence,

$$\begin{aligned}\hat{M}_{i\bar{l}}^{a\bar{l}} &= \lim_{\epsilon \rightarrow 0} \frac{[\nu e^{G(y_k - \epsilon)} \mathbf{1}_{ai}] [\mathbf{1}'_{ai} e^{H_{cc}\epsilon} h_l]}{\mu e^{H_{cc}y_k} H_{cd} \mathbf{1}} \\ &= \frac{[\nu e^{Gy_k} \mathbf{1}_{ai}] [\mathbf{1}'_{ai} h_l]}{\mu e^{H_{cc}y_k} H_{cd} \mathbf{1}}.\end{aligned}\tag{4.33}$$

We next discuss an efficient approach for evaluating the integrals in (4.26) and (4.30). This approach involves matrix exponential without numerical integration. In particular, we are interested in evaluating

$$J(y) = \int_0^y e^{H_{cc}(y-t)} \cdot H_{cd} \mathbf{1} \nu \cdot e^{Gt} dt.\tag{4.34}$$

Let

$$C = \begin{pmatrix} H_{cc} & H_{cd} \mathbf{1} \nu \\ \mathbf{0} & G \end{pmatrix}.\tag{4.35}$$

According to [10, Theorem 1], the upper right block of the matrix exponential  $e^{Cy}$  is  $J(y)$ .

Other approaches for evaluating the matrix exponential includes Padé approximation [47] and Runge-Kutta numerical integration [48]. The Padé approximation approach [47] requires an order of  $r^3$  operations for a matrix of order  $r$  while this approach is much faster than the approach of [48] which was used in [9] to evaluate (4.34).

### 4.3 Tree-Structured Networks

In this section, we study the problem of link delay density estimation from source-destination delay measurements in a tree-structured network. Some researchers have focused on networks whose topology can be represented by a tree. Some of these studies have been addressed in Section 2.1.2. In a tree-structured network, packets enter at a root node and are transmitted towards the terminal nodes. The partially observable bivariate Markov model is applicable to the tree-structured networks as well. In such networks, both random and deterministic routing regimes are possible. We next study a pre-determined route in a tree-structured network. We number the nodes consecutively, i.e., the root node is numbered by 1, and the destination node is numbered by  $d_1 + 1$ . Hence,  $\mathbb{X}_1 = \{1, \dots, d_1\}$  and  $\mathbb{X}_2 = \{d_1 + 1\}$ . The generator of such a bivariate Markov chain model is given by

$$G = \left( \begin{array}{cccc|c} G_{11} & G_{12} & & & \mathbf{0} \\ & \ddots & \ddots & & \\ & & G_{d_1-1,d_1-1} & G_{d_1-1,d_1} & \\ & & & G_{d_1,d_1} & h_1 \\ \hline \xi_1 \nu_1 & & \mathbf{0} & & -\xi_1 \end{array} \right), \quad (4.36)$$

where clearly the  $X$ -chain always jumps from node 1 to node 2, then from 2 to 3 and so on. For  $i = 1, \dots, d_1 - 1$ , all the  $G_{i,i+1}$  matrices and  $G_{d_1 d_1}$  matrix are assumed to be  $r \times r$ . Also,  $h_1$  and  $\nu_1$  are  $r \times 1$  and  $1 \times r$  vectors, respectively. The bivariate Markov process described by (4.36) never starts from any state  $(a, i)$  where  $a \neq 1$ . Hence, the initial distribution of such a chain is given by

$$\nu = (\nu_1, \mathbf{0}, \dots, \mathbf{0}, 0). \quad (4.37)$$

Recall that  $\bar{\nu}_a^k = \frac{\nu_a^k}{\nu_a^k \mathbf{1}}$ . Let  $\bar{\nu}_a^k(i)$  denote the  $i$ -th component of  $\bar{\nu}_a^k$ . A pre-determined route in a tree-structured network with diagonal  $\{G_{aa}\}$  was also studied in [11]. In such a network, the sojourn time of the process in state  $a$  given that  $X_k = a$  following (4.4) is given by

$$p_\phi(t | a) = \sum_{i=1}^r -\bar{\nu}_a^k(i) g_{aa}(ii) e^{g_{aa}(ii)t}. \quad (4.38)$$

The density of (4.38) is a mixture of  $r$  exponential densities. For  $X_k = a$  and  $X_{k+1} = a + 1$ , the density of delay on the link connecting  $x_k$  to  $x_{k+1}$  follows from (4.10) and using  $G_{aa}\mathbf{1} = G_{a,a+1}\mathbf{1}$ , and is given by

$$\begin{aligned} p_\phi(t | a, a + 1) &= \frac{\nu_a^k e^{G_{aa}t} G_{a,a+1} \mathbf{1}}{\nu_a^k D_{a,a+1} \mathbf{1}} = \frac{\nu_a^k}{\nu_a^k \mathbf{1}} e^{G_{aa}t} G_{a,a+1} \mathbf{1} \\ &= \sum_{i=1}^r -\bar{\nu}_a^k(i) g_{aa}(ii) e^{g_{aa}(ii)t}. \end{aligned} \quad (4.39)$$

Hence, the sojourn time in node  $a$  is the same as delay on the link connecting  $a$  to  $a + 1$ . The source-destination delay is the sum of sojourn times in the states of the  $X$ -chain, i.e.,

$$Y = \sum_{l=1}^{d_1} T_l. \quad (4.40)$$

In [11],  $\{g_{aa}(ii)\}$  were assumed distinct and  $\{T_l\}$  were assumed independent. An EM algorithm was developed in [11] for estimating the parameter of the mixture model of (4.39), i.e.,  $\{g_{aa}(ii)\}$  and  $\{\bar{\nu}_a^k\}$ , from  $K$  independent realizations of  $Y$ . The key to the EM algorithm as described earlier is the evaluation of the density of  $Y$ . We next summarize the approach

of [11] where the moment generating functions were used to derive the density of  $Y$ . Assume that for a given  $k$ ,  $Y_k = y_k$ . Let  $\varphi_a^k = (\varphi_{a1}^k, \dots, \varphi_{ar}^k)$  where  $\varphi_{ai}^k = 0$  if the indicator function given in (4.15)  $\varphi_{ai}(t) = 0$  for all  $t \in [0, y_k]$ , and  $\varphi_{ai}^k = 1$  otherwise. For  $a \in \mathbb{X}_1$ ,  $\varphi_a^k$  has one and only one non zero entry. Hence, given  $\varphi_a^k$  is one,  $T_a$  has an exponential density with parameter  $-g_{aa}(ii)$ . Therefore, the bivariate Markov chain as observed in  $[0, y_k]$  interval can be expressed by  $\tilde{Z}_k = \{(T_a, \varphi_a^k); a \in \mathbb{X}_1, \}$ . The complete statistics for the EM algorithm is given by  $\{(\tilde{Z}_k, Y_k), k = 1, \dots, K\}$ . The density of  $Y_k$  as evaluated in [11] is given by

$$p_\phi(Y_k \in dy) = - \sum_{a=1}^{d_1} \sum_{i=1}^r \beta_a^k(i) g_{aa}(ii) e^{g_{aa}(ii)y_k}, \quad (4.41)$$

where  $\beta_a^k(i)$  is given by [11],

$$\beta_a^k(i) = \bar{\nu}_a^k(i) \prod_{b=1, b \neq a}^{d_1} \sum_{j=1}^r \frac{-\bar{\nu}_b^k(j) g_{bb}(jj)}{g_{aa}(ii) - g_{bb}(jj)}. \quad (4.42)$$

The density in (4.41) was then used to derive the conditional expectations required to estimate  $\{\bar{\nu}_a^k\}$  and  $\{g_{aa}(ii)\}$  in [11]. The density of  $Y_k$  as evaluated in (4.8) does not require  $\{T_l\}$  to be independent.

## 4.4 Numerical Results

In this section, a bivariate Markov chain was attributed to an unstructured network with random routing regime as well as to a tree-structured network with deterministic routing regime. We have compared our results for the tree-structured network with the mixture modeling approach of [11].

#### 4.4.1 Unstructured Network

The generator of an unstructured network with random routing regime is given in Eq. (4.1). We first demonstrate the accuracy of the estimation approach as follows. We have used a bivariate Markov chain with  $d_1 = 8$  source states,  $d_2 = 8$  destination states, and  $r = 4$  for each source state. The elements of the initial distribution vector  $\mu$  were first chosen randomly on  $[0, 1]$  interval, and were then normalized so that  $\mu\mathbf{1} = 1$ . The entries of the generator  $G$  were selected as follows. The off-diagonal entries of  $H_{cc}$  as well as the elements of  $H_{cd}$  were chosen randomly according to a uniform distribution on  $[0, 100]$ . For  $i = 1, \dots, d_2$ ,  $e_i$  was set to  $\xi_i\mu$  where  $x_i = 10^5$ . By choosing large values for  $\{\xi_i\}$ , the sojourn time in each destination node would be negligible. The source-destination delay was next generated as follows. The bivariate Markov chain was first initialized according to  $\mu$ . The chain spent some exponential time in each pair of states, until it reaches the  $(a, 1)$  state for some  $a \in \mathbb{X}_2$ . The total sojourn time was then recorded, and the chain was bounced back to one of the source nodes chosen according to  $\mu$ . The  $K = 20,000$  source-destination delay measurements  $\{Y_k\}$  were generated in this manner. We next used a bivariate Markov chain with  $d_1 = 8$  source states,  $d_2 = 8$  destination states, and  $r = 4$  for each source state to model the traffic over the network. The EM algorithm developed in Section 4.2.2 was used to find the maximum likelihood estimation of the model parameter given the generated data  $\{Y_k\}$ . The EM algorithm was initialized as follows. The entries of the initial distribution  $\mu$  were chosen randomly according to a uniform distribution on  $[0, 1]$ , and were then normalized. The off-diagonal entries of  $H_{cc}$  as well as the elements of  $H_{cd}$  were chosen randomly according to a uniform distribution on  $[0, 200]$ . The EM algorithm was allowed to run for 1000 iterations to estimate the parameter of the model as the likelihood of the observation increases very slowly. Due to large dimensions of the generator  $G$ , we only present the true and estimated values for the diagonal entries of  $H_{cc}$ . Fig. 4.1 demonstrates the true and estimated values for the 32 diagonal elements of  $H_{cc}$ .

Next, we used a bivariate Markov chain with  $d_1 = 8$  source states,  $d_2 = 8$  destination

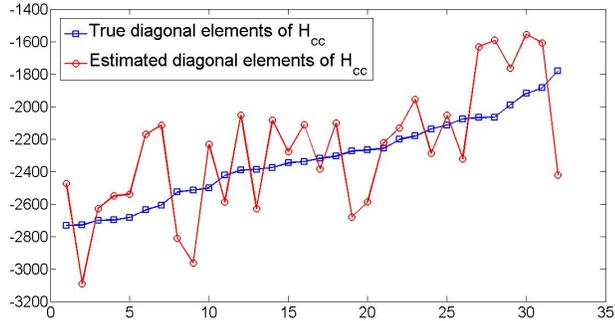


Figure 4.1: True diagonal elements of  $H_{cc}$  depicted in ascending order, and their estimates.

states, and  $r = 10$  for each source state to generate the source-destination delay measurements. The entries of  $G$  and  $\mu$  were chosen as described in the previous example. As the generator  $G$  contains a large number of entries, we only specify the entries of  $G_{11}$  and  $\nu$  as an instance in Appendix A. We have used a relatively low order bivariate Markov chain model with  $d_1 = d_2 = 8$  and  $r = 4$  to model the traffic over the network, and the EM algorithm with the same initialization procedure as described earlier was used to estimate the parameter of the model. The EM algorithm was terminated after 1000 iterations. The link delay density and the routing probabilities were then inferred given the estimated parameter. In this unstructured network, the bivariate Markov chain model contains  $d_1^2 - d_1 = 56$  links connecting source nodes to one another,  $d_1 d_2 = 64$  links connecting source nodes to the destination nodes, and  $d_2 d_1 = 64$  links connecting destination nodes to the source nodes. We have already assumed a negligible delay on the links within the third set. Hence, we are only interested in inferring the delay on the links of the first two sets which total 120 links.

Fig. 2.4 shows plots of the density of the overall source-destination delay, given by  $P_\phi(Y_k \in dy)$  in (4.8), when  $\phi$  is the true parameter and when  $\phi$  is the estimated parameter.

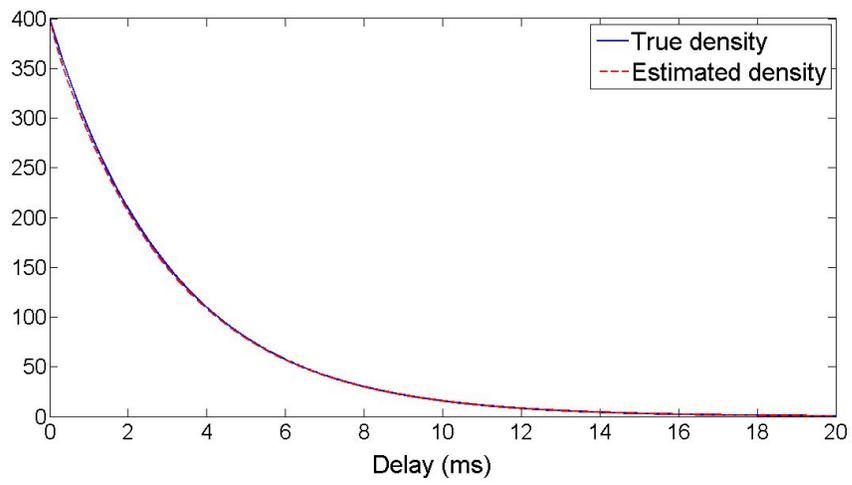


Figure 4.2: True and estimated overall source-destination delay densities.

To further demonstrate the accuracy of the EM algorithm in estimating the source-destination delay density, the divergence between the true and estimated densities has been evaluated. According to [49], the Kullback-Leibler divergence between two densities  $f$  and  $g$  is given by,

$$D_{KL}(f||g) = \int_{-\infty}^{\infty} f(v) \log \frac{f(v)}{g(v)} dv. \quad (4.43)$$

The Kullback-Leibler divergence for the densities with at least .99 probability of the more dispersed density concentrated in  $[0, u]$ , was approximated as

$$D_{KL}(f||g) \approx \sum_{i=0}^{\lfloor u/\Delta \rfloor} \Delta f(\Delta i) \log \frac{f(\Delta i)}{g(\Delta i)}, \quad (4.44)$$

where  $\Delta = 10^{-5}$  was chosen. In our study, we have assumed that  $f$  is the true density while  $g$  is the estimated density. The estimated divergence value for the two densities in Fig. 4.2 was found to be  $7.1819 \cdot 10^{-6}$ .

The true and estimated parameter of the bivariate Markov chain model were used to find the true and estimated delay density on each link of the unstructured network following (4.10). We have chosen two arbitrary links and plotted the true and estimated link delay densities for those links in Fig. 4.3. The corresponding divergence values for the link delay estimates in Fig. 4.3(a) and 4.3(b) are  $4.038 \cdot 10^{-5}$  and 0.012, respectively. Fig. 4.4 shows the divergence values for all 120 link delay estimates.

We have also evaluated the packet routing probabilities using (4.11). Fig. 4.5 depicts the mean squared error in estimating the packet routing probabilities.

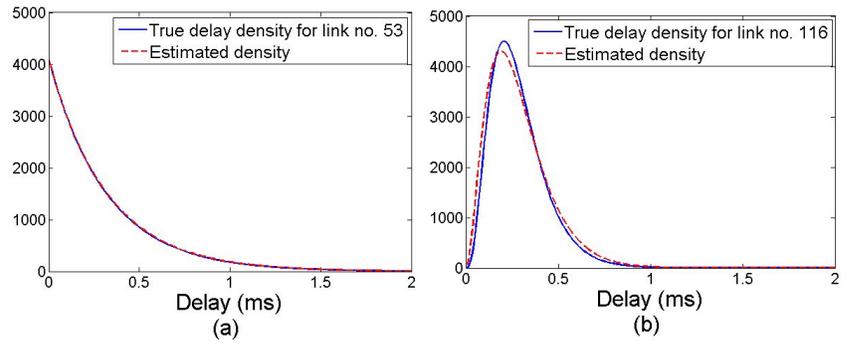


Figure 4.3: True and estimated link delay densities are shown in (a) and (b) for two of the links.

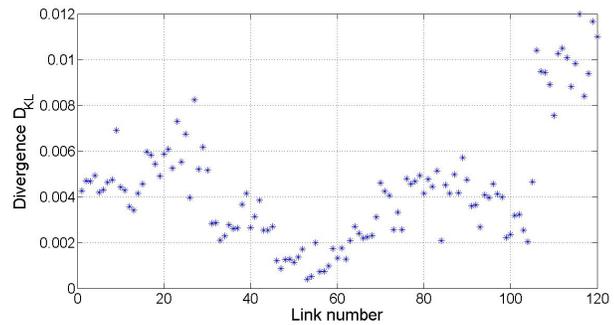


Figure 4.4: Divergence values for the estimated link delay densities.

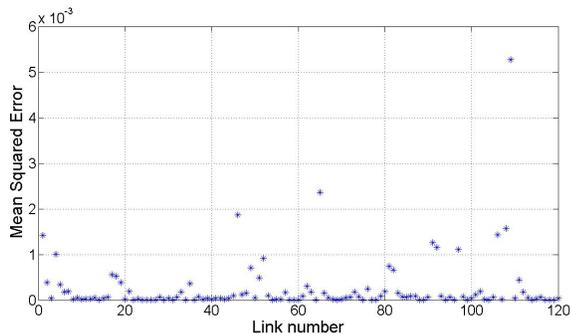


Figure 4.5: Mean squared error for estimated packet routing probabilities on the various links.

#### 4.4.2 Tree-Structured Networks

In this section, we have implemented a tree-structured network, and applied a bivariate Markov chain to model a pre-determined route of that network. The source-destination delay measurements were generated using a bivariate Markov chain with  $d_1 = 5$ ,  $d_2 = 1$  and  $r = 10$ . The off-diagonal entries of the generator as given in (4.36), and the entries of the initial distribution  $\nu_1$  were generated randomly as described in the unstructured network. The value of  $\xi_1$  was set to  $10^5$  as before.

We have then used the EM algorithm to estimate the parameter of a bivariate Markov chain with  $d_1 = 5$ ,  $d_2 = 1$  and  $r = 4$  given the generated source-destination delays. The same generated data was used to implement the mixture-based modeling approach of [11]. The delay density on each link was modeled by a mixture of four exponential densities. The proposed EM algorithm for estimation of the parameter of the partially observable bivariate Markov chain was initialized using two different procedures denoted by  $A$  and  $B$ . In experiment  $A$ , the off-diagonal entries of  $G$  and the non-zero elements of  $\nu$  were chosen randomly as described in the unstructured network. In experiment  $B$ , the matrices  $\{G_{aa}\}$  were assumed to be diagonal, and the non-zero and off-diagonal entries of  $G$  as well

Table 4.1: Divergence between true and estimated densities in the tree-structured network.

	$D_{KL} - A$	$D_{KL} - B$	$D_{KL} - C$	$D_{KL} - D$	$D_{KL} - E$
Source-dest.	$4.8082 \cdot 10^{-5}$	$7.6032 \cdot 10^{-4}$	$1.9929 \cdot 10^{-5}$	$2.123 \cdot 10^{-4}$	$9.8786 \cdot 10^{-5}$
Link no. 1	0.0004	0.0016	0.0068	0.0035	0.2497
Link no. 2	0.001	0.1015	0.008	0.1047	0.2878
Link no. 3	0.0103	0.0166	0.021	0.0126	0.1181
Link no. 4	0.0079	0.0706	0.0521	0.0885	0.2623
Link no. 5	0.0017	0.0339	0.0096	0.0303	3.3571

as the entries of  $\nu_1$  were chosen randomly as described in experiment *A*. The estimated link delay density obtained from experiment *B* is the mixture density given by (4.39), and this model does not rely on the assumption that the link delays are independent. The EM algorithm of [11] was also initialized using several different procedures denoted by *C-E*. In experiment *C*, the initial value of the parameter of the  $i$ th mixture component for the  $a$ th link, where  $i \in \{1, 2, 3, 4\}$  and  $a \in \{1, 2, 3, 4, 5\}$ , was identical to the initial value of  $g_{aa}(ii)$  in experiment *A*. We have also used  $\{\bar{\nu}_a^k(i)\}$  from *A* to initialize the mixing probabilities in [11]. In experiment *D*, the initialization was the same as *B*. The initialization in experiment *E* was drawn randomly as in *A*. Fig. 4.6 depicts the true and estimated source-destination delay densities obtained from experiments *A-E*. The true and estimated link delay densities for the five links on the pre-determined route are plotted in Fig. 4.7. The divergence values between the true and estimated link delay densities are provided in Table. 4.1.

Clearly, experiment *A* provides better or similar results to *C* where the two EM algorithms in bivariate Markov chain model and the mixture-based modeling of [11] are initialized similarly. The results provided by experiment *B* and experiment *D* are also comparable. We are not interested in the results obtained from experiment *E* as it performs the worst.

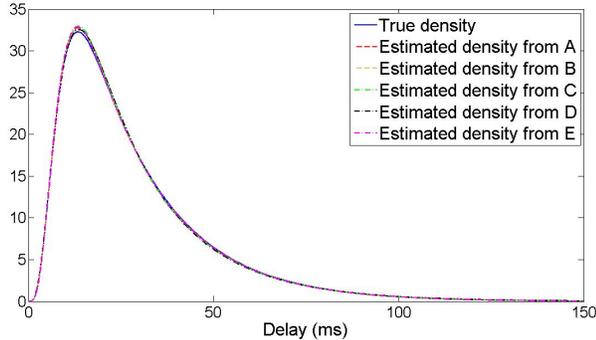


Figure 4.6: True and estimated source-destination delay densities in the tree-structured network.

#### 4.4.3 Recursive implementation of the EM algorithm

In this section, we discuss the recursive implementation of Eq. (4.18) and Eq. (4.19). In recursive implementation of the EM algorithm, a new estimate of the parameter of the model is obtained using the previous estimate and a block of the observation sequence. We refer to the recursive implementation of the EM algorithm as the online algorithm. In the online algorithm, the estimates are updating as the data becomes available, and there is no need to store the entire sequence of observations. Online estimation of hidden Markov models was previously studied in [50]. Suppose that we receive the observation sequence  $\{Y_1, \dots, Y_K\}$  in blocks of the same size, i.e.,  $\{Y_1, \dots, Y_L\}$ ,  $\{Y_{L+1}, \dots, Y_{2L}\}$ ,  $\dots$ ,  $\{Y_{K-L+1}, \dots, Y_K\}$  where  $L$  is the block size. In the online algorithm, the recursions (4.18) and (4.19) at the end of the  $(\iota + 1)$  iteration are updated using  $\{Y_{\iota L+1}, \dots, Y_{(\iota+1)L}\}$ , and we use the previous estimate of the parameter of the model,  $\phi_\iota$ , to derive the conditional expectations in (4.18) and (4.19). Hence, new estimates of the off-diagonal elements of  $H_{cc}$ , the elements of  $H_{cd}$ , and the initial distribution of the chain at the end of the  $(\iota + 1)$ -th iteration, are given by

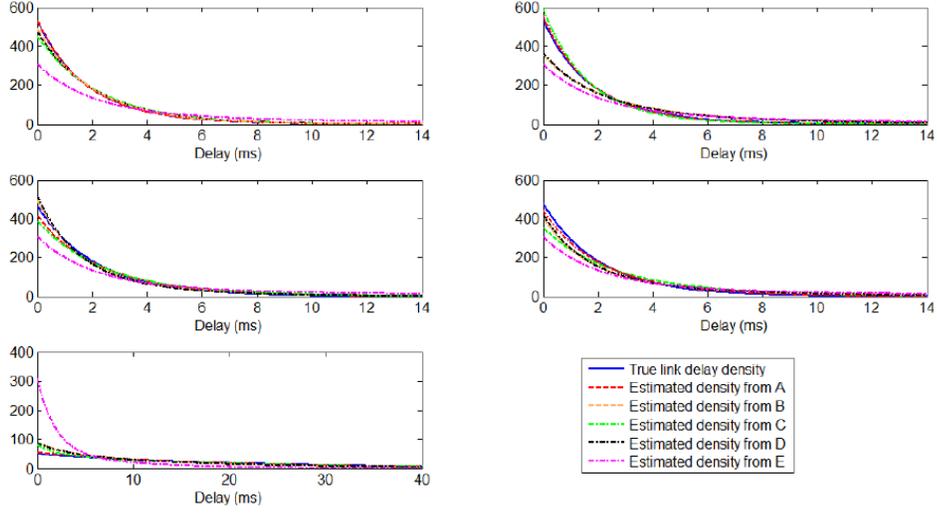


Figure 4.7: True and estimated link delay densities for the five different links in the tree-structured network.

$$\hat{\nu}_{ai} = \frac{1}{L} \sum_{k=\iota L+1}^{(\iota+1)L} E_{\phi_\iota} \{ \varphi_{ai}(0) \mid Y_k = y_k \}, \quad (4.45)$$

$$\hat{g}_{ab}(ij) = \frac{\sum_{k=\iota L+1}^{(\iota+1)L} E_{\phi_\iota} \{ M_{ij}^{ab}(Y_k) \mid Y_k = y_k \}}{\sum_{k=\iota L+1}^{(\iota+1)L} E_{\phi_\iota} \{ D_i^a(Y_k) \mid Y_k = y_k \}}, \quad (b, j) \neq (a, i). \quad (4.46)$$

We have studied the same unstructured network with the random routing regime as in Section 4.4.1. We have used the same observation sequence generated in Section 4.4.1 from the bivariate Markov model with  $d_1 = d_2 = 8$  and  $r = 10$ . A partially observable bivariate Markov chain model with  $d_1 = d_2 = 8$  and  $r = 4$  was then applied to the observation sequence, and the initial values for the parameter of that model was chosen to

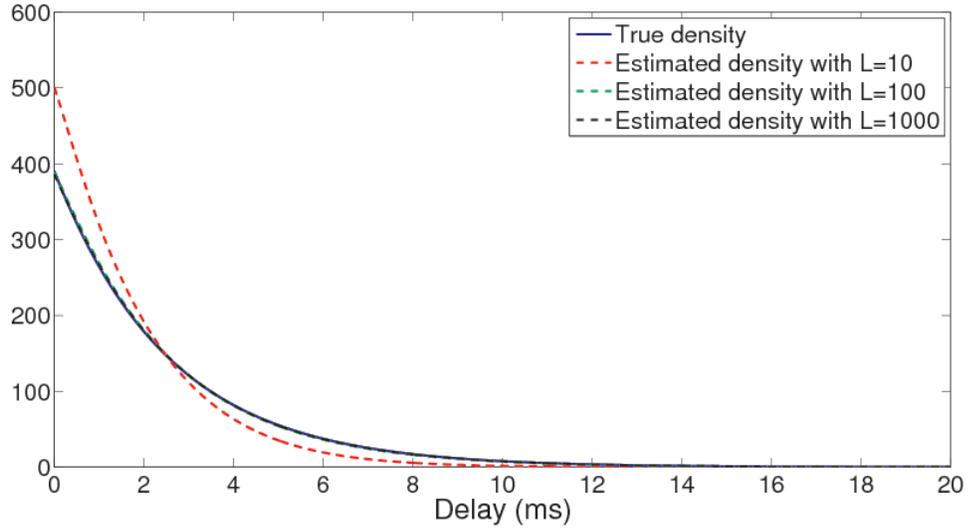


Figure 4.8: True and estimated overall source-destination delay densities obtained from online algorithm.

be identical to the initial values used in Section 4.4.1. The value of  $\xi_i$  for  $i = 1, \dots, d_2$  was set to  $10^5$  as before. We next implemented recursions (4.45) and (4.46) using  $L = 10, 100$  and  $1000$ . Clearly, the online algorithm was run for  $K/L$  iterations in each scenario. Fig. 4.8 demonstrates the true and estimated source-destination delay densities obtained from the online algorithm using  $L = 10, 100$  and  $1000$ . Table 4.2 shows the corresponding divergence values between the true and estimated source-destination delay densities. The divergence values between the true and estimated link delay densities for the 120 links are demonstrated in Fig. 4.9. Fig. 4.10 shows the mean squared error for estimated packet routing probabilities on the 120 links. The online algorithm provides better results in estimating source-destination and link delay densities as well as packet routing probabilities with increasing the size of the block,  $L$ .

Table 4.2: Divergence between true and estimated densities in the unstructured network obtained from the online algorithm.

	$D_{KL}; L = 10$	$D_{KL}; L = 100$	$D_{KL}; L = 1000$
Source-dest.	0.0777	$2.4144 \cdot 10^{-4}$	$6.4544 \cdot 10^{-5}$

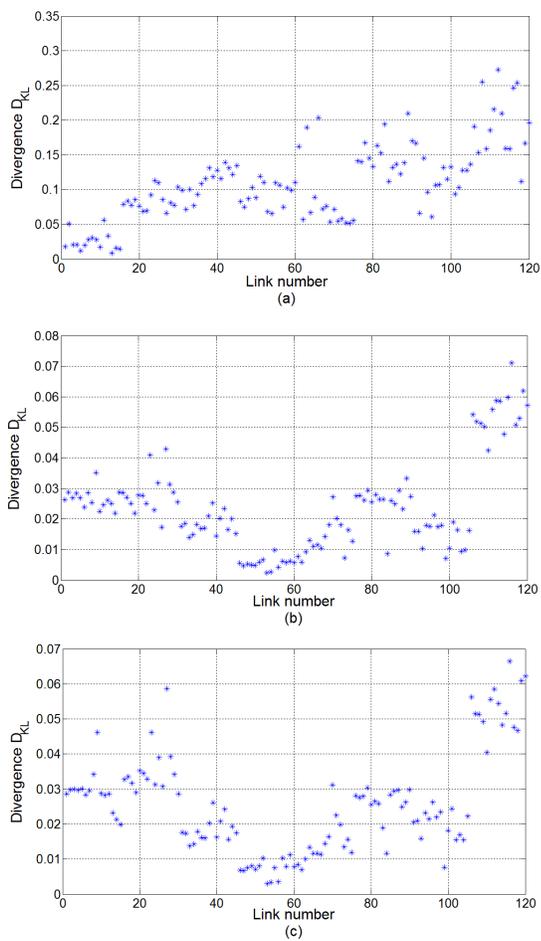


Figure 4.9: Divergence values for the estimated link delay densities obtained from online algorithm with (a)  $L = 10$  (b)  $L = 100$  and (c)  $L = 1000$ .

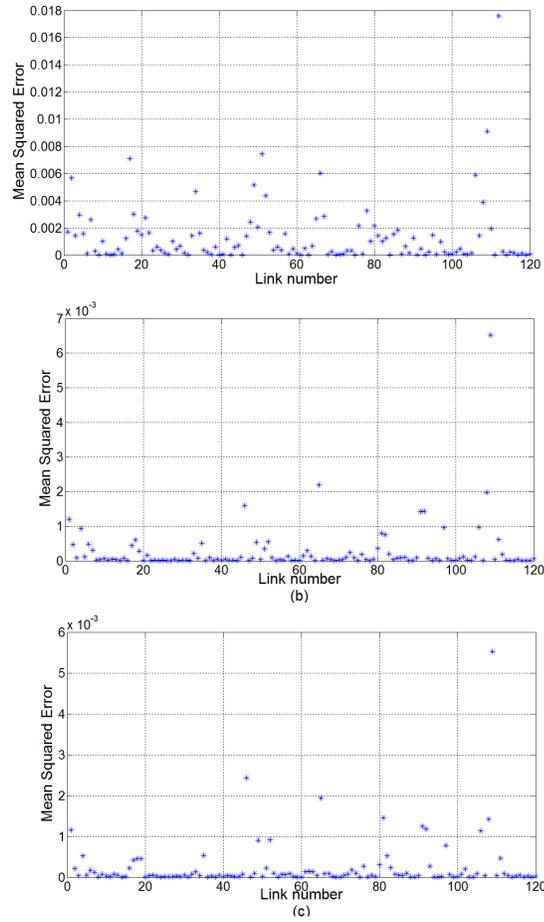


Figure 4.10: Mean squared error for estimated packet routing probabilities on the various links obtained from online algorithm with (a)  $L = 10$  (b)  $L = 100$  and (c)  $L = 1000$ .

## Chapter 5: Source-destination traffic rates estimation

In this chapter, we study the problem of source-destination traffic rates estimation from aggregated traffic flow at input and output nodes in the network. We detail a covariance-based approach we developed for estimating the rates in the model of [4]. We also detail a second rate estimation approach for the model in [4] which is based on the maximum entropy principle. We then provide numerical results to compare the performance of the proposed approaches with that of the approaches discussed in [2] and [4].

### 5.1 Covariance-based rate estimation

Consider the linear model,  $\mathbf{V} = \mathbf{A}\mathbf{U}$ , described in Section 2.2.2 . Recall that  $a_{ij} = 1$  if  $U_j$  contributes to  $V_i$ , and  $a_{ij} = 0$  otherwise. Next, we show that the source-destination rates,  $\lambda$ , can be estimated uniquely from matching the covariance of the model to empirical covariance measurements.

Consider a network with  $N$  nodes and  $c$  source-destination pairs. For source-destination  $j = (j_1, j_2)$  where  $j_2 \neq N$ ,  $U_j$  only contributes to the traffic flow originated from  $j_1$ ,  $V_{j_1}$ , and the traffic flow destined at  $j_2$ ,  $V_{N+j_2}$ . In this case,  $a_{j_1 j} = 1, \sum_{l \neq j_1} a_{lj} = 1$ . For source-destination  $j = (j_1, N)$ ,  $U_j$  only contributes to  $V_{j_1}$ . In this case,  $a_{j_1 j} = 1, \sum_{l \neq j_1} a_{lj} = 0$ . The covariance matrix of  $\mathbf{V}$ ,  $\psi$ , is given by Eq. (2.16) where  $a_{ij}a_{lj} = 1$  if and only if  $U_j$  contributes to both  $V_i$  and  $V_l$ , and  $a_{ij}a_{lj} = 0$  otherwise. Clearly, for  $i \neq l$ ,  $\psi_{il} = 0$  if both  $V_i$  and  $V_l$  represent the incoming traffic counts to some nodes in the network. The same argument can be made if both  $V_i$  and  $V_l$  represent the outgoing traffic counts from some nodes in the network. However, if  $V_i$  represents the outgoing traffic counts from some node  $j_1$  and  $V_l$  represents the incoming traffic counts to some other node  $j_2$ ,  $\psi_{il} = \lambda_j$  where  $j = (j_1, j_2)$ .

Hence, for  $j = (j_1, j_2), j_2 \neq N$ ,  $\lambda_j$  is given by

$$\lambda_j = \psi_{j_1 l}, \quad (5.1)$$

where  $l = j_2 + N$ , and for  $j = (j_1, N)$ ,  $\lambda_j$  is given by

$$\lambda_j = \sum_h a_{j_1 h} \lambda_h (1 - \sum_{l \neq j_1} a_{lh}) = \psi_{j_1 j_1} - \sum_{l \neq j_1} \psi_{j_1 l}. \quad (5.2)$$

It follows from Eq. 5.1 and Eq. (5.2) that the source-destination traffic rates can be determined by  $\{\psi_{ij}\}$ . We demonstrate this feature through a simple example. Consider the network of Fig. 2.6 and the corresponding  $A$  matrix given in Table. 2.5. The covariance matrix,  $\psi$ , for this network is given by,

$$\psi = \begin{bmatrix} \psi_{11} & 0 & 0 & 0 & 0 & \lambda_1 & \lambda_2 \\ 0 & \psi_{22} & 0 & 0 & \lambda_4 & 0 & \lambda_5 \\ 0 & 0 & \psi_{33} & 0 & \lambda_7 & \lambda_8 & 0 \\ 0 & 0 & 0 & \psi_{44} & \lambda_{10} & \lambda_{11} & \lambda_{12} \\ 0 & \lambda_4 & \lambda_7 & \lambda_{10} & \psi_{55} & 0 & 0 \\ \lambda_1 & 0 & \lambda_8 & \lambda_{11} & 0 & \psi_{66} & 0 \\ \lambda_2 & \lambda_5 & 0 & \lambda_{12} & 0 & 0 & \psi_{77} \end{bmatrix}, \quad (5.3)$$

where

$$\psi_{11} = \lambda_1 + \lambda_2 + \lambda_3,$$

$$\psi_{22} = \lambda_4 + \lambda_5 + \lambda_6,$$

$$\psi_{33} = \lambda_7 + \lambda_8 + \lambda_9,$$

$$\psi_{44} = \lambda_{10} + \lambda_{11} + \lambda_{12},$$

$$\psi_{55} = \lambda_4 + \lambda_7 + \lambda_{10},$$

$$\psi_{66} = \lambda_1 + \lambda_8 + \lambda_{11},$$

$$\psi_{77} = \lambda_2 + \lambda_5 + \lambda_{12}.$$

Next, we recursively estimate  $\psi$  by its empirical value. The sample covariance matrix,  $\widehat{\psi}$ , is given by (2.18). Let  $\widehat{\psi}^{(\iota)} = \{\widehat{\psi}_{il}^{(\iota)}\}$  and  $\widehat{\eta}^{(\iota)} = \{\widehat{\eta}_i^{(\iota)}\}$  denote the sample covariance and mean of  $\mathbf{V}$  at the end of the  $\iota$ -th iteration, respectively. We set the initial values  $\{\widehat{\eta}_i^{(0)}\}$  and  $\{\widehat{\psi}_{il}^{(0)}\}$  to zeros. Hence, for  $\iota = 0, 1, \dots, K-1; i = 1, \dots, q; l = 1, \dots, q$ ,

$$\begin{aligned} \widehat{\eta}_i^{(\iota+1)} &= \frac{1}{\iota+1} [\iota \widehat{\eta}_i^{(\iota)} + v_i^{(\iota+1)}], \\ \widehat{\psi}_{il}^{(\iota+1)} &= \frac{\widehat{\psi}_{il}^{(\iota)} \iota}{\iota+1} + \frac{\iota}{(\iota+1)^2} [\widehat{\eta}_i^{(\iota)} \widehat{\eta}_l^{(\iota)} + v_i^{(\iota+1)} v_l^{(\iota+1)} - \widehat{\eta}_i^{(\iota)} v_l^{(\iota+1)} - \widehat{\eta}_l^{(\iota)} v_i^{(\iota+1)}]. \end{aligned} \quad (5.4)$$

The parameter  $\lambda$  can be estimated recursively by matching the covariance matrix,  $\psi$ , to its empirical value, as follows. Let  $\lambda^{(\iota)} = (\lambda_1^{(\iota)}, \dots, \lambda_c^{(\iota)})'$  denote the estimate of  $\lambda$  at the end of the  $\iota$ -th iteration. For  $j = (j_1, j_2), j_2 \neq N$ ,

$$\lambda_j^{(\ell)} = \max\{0, \widehat{\psi}_{j_1 l}^{(\ell)}\}, \quad (5.5)$$

where  $l = N + j_2$ . For  $j = (j_1, N)$ ,

$$\lambda_j^{(\ell)} = \max\{0, \widehat{\psi}_{j_1 j_1}^{(\ell)} - \sum_{l \neq j_1} \widehat{\psi}_{j_1 l}^{(\ell)}\}. \quad (5.6)$$

## 5.2 Maximum-entropy approach

In this section, we use the maximum entropy principle and propose a new approach for estimating  $\lambda$  from  $\mathbf{v}_1^K$ . The approach was adopted from [12] where it was applied for estimation of propagation link delay as explained in Section 2.1.1.

### 5.2.1 Shannon's Entropy

Entropy is a measure of uncertainty in a random variable. Let  $W$  denote a discrete random variable with probability mass function  $p_W(\cdot)$ . The alphabet of  $W$  is denoted by  $\mathbb{W}$ . An expected value with respect to  $p_W(\cdot)$  is denoted by  $E_W\{\cdot\}$ . The entropy of  $W$  is defined by [51],

$$H(W) = E_W\{-\log p_W(W)\} = - \sum_{w \in \mathbb{W}} p_W(w) \log p_W(w). \quad (5.7)$$

### 5.2.2 Underlying framework

In this section, we develop the underlying framework for maximum entropy estimation of traffic rates.

Let the network be observed in a sufficiently large time interval. Consider packets originated in various nodes of the network during that interval. Label each of these packets

with its source-destination address. Now, suppose that we put all the labelled packets into a jar and randomly draw a packet from that jar. Let  $W$  denote the label of the virtual packet. Let  $p_j$  denote the probability of  $W = j$  for  $j = 1, \dots, c$ , and let  $\mathbf{p} = (p_1, \dots, p_c)'$ . Our goal is estimating  $\{p_j\}$ . Once  $\mathbf{p}$  is estimated, it can be used to estimate  $\lambda$  as we shall discuss shortly. Clearly, for  $j = 1, \dots, c$ ,

$$p_j \geq 0,$$

$$\sum_{j=1}^c p_j = 1. \tag{5.8}$$

The maximum-entropy approach for estimating  $\mathbf{p}$  aims at achieving the least informative solution that is consistent with the given constraints specified in (5.8). Entropy can be regarded as a suitable function to measure the lack of knowledge about the proposed conceptual experiment, and thus it is a suitable function to maximize in order to find  $\{p_j\}$ . Hence, we propose to estimate the desired  $\mathbf{p}$  that maximizes the entropy of  $W$  subject to the constraints of (5.8). The entropy of  $W$  is given by,

$$H(W) = - \sum_{j=1}^c p_j \log p_j. \tag{5.9}$$

It is well known that  $H(W)$  is maximized by a uniform distribution, i.e., for  $j = 1, \dots, c$ ,

$$p_j = \frac{1}{c}. \tag{5.10}$$

Next, we derive the maximum entropy estimate of  $\mathbf{p}$  under a new set of constraints. Clearly, the larger  $\lambda_j$  is, the more packets of label  $j$  are present in the jar and the higher

$p_j$  is. Hence, a reasonable choice for  $p_j$  is given by

$$p_j = \frac{\lambda_j}{C}, \quad (5.11)$$

where the constant  $C = \sum_{j=1}^c \lambda_j$ .

As described earlier, for  $\lambda$  being a stationary point of the likelihood function of  $\mathbf{V}$ , the first moment equations are satisfied, i.e.,  $\hat{\eta} = A\lambda$  where  $\hat{\eta}$  is given by (5.1). The constant  $C$  in (5.11) is given by

$$C = \sum_j \lambda_j = \sum_j \lambda_j \sum_{i=1}^N a_{ij} = \sum_{i=1}^N \sum_j a_{ij} \lambda_j = \sum_{i=1}^N \hat{\eta}_i, \quad (5.12)$$

which follows from the model feature described in Section 5.1. Let  $\beta = (\beta_1, \dots, \beta_q)'$  where  $\beta_i = \frac{\hat{\eta}_i}{C}$ . By dividing both sides of  $\hat{\eta} = A\lambda$  by the constant  $C$ , the following set of equations is obtained,

$$\beta = A\mathbf{p}. \quad (5.13)$$

Thus, our goal is to estimate  $\mathbf{p}$  which maximizes (5.9) under the constrains in (5.8) and (5.13). Let  $\alpha = (\alpha_0, \dots, \alpha_q)'$  denote the vector of Lagrange multipliers. The Lagrangian is given by,

$$\mathcal{L}(\mathbf{p}, \alpha) = - \sum_{j=1}^c p_j \log p_j - \alpha_0 \left( \sum_{j=1}^c p_j - 1 \right) - \sum_{i=1}^q \alpha_i \left( \sum_{j=1}^c a_{ij} p_j - \beta_i \right). \quad (5.14)$$

Setting the derivative of (5.14) with respect to  $\{p_j\}$ ,  $\{\alpha_i\}$ , and  $\alpha_0$  to zeros, we have,

$$-\log p_j - \alpha_0 - \sum_i \alpha_i a_{ij} = 0, \quad j = 1, \dots, c, \quad (5.15)$$

$$\sum_j a_{ij} p_j - \beta_i = 0, \quad i = 1, \dots, q, \quad (5.16)$$

$$\sum_j p_j = 1. \quad (5.17)$$

Thus, the desired probabilities  $\{p_j\}$  and the Lagrange multiplier  $\alpha_0$  are given by,

$$p_j = e^{-(\alpha_0 + \sum_i \alpha_i a_{ij})}, \quad j = 1, \dots, c, \quad (5.18)$$

$$\alpha_0 = \log \sum_j e^{-(\sum_i \alpha_i a_{ij})}. \quad (5.19)$$

By substitution of (5.19) in (5.18) and the result in (5.16), the rest of Lagrange multipliers,  $\{\alpha_i\}$ , satisfy the following equations,

$$\frac{\sum_j a_{ij} e^{-\sum_i \alpha_i a_{ij}}}{\sum_j e^{-\sum_i \alpha_i a_{ij}}} = \beta_i, \quad i = 1, \dots, q. \quad (5.20)$$

Define  $f(\alpha) = \sum_j e^{-\sum_i \alpha_i a_{ij}}$ . The probabilities  $\{p_j\}$  and the Lagrange multipliers are determined by solving the following set of equations for  $\alpha$ ,

$$-\frac{\partial \log f(\alpha)}{\partial \alpha_i} = \beta_i, \quad i = 1, \dots, q. \quad (5.21)$$

Hence,

$$p_j = \frac{e^{-\sum_i \alpha_i a_{ij}}}{f(\alpha)}, \quad j = 1, \dots, c, \quad (5.22)$$

$$\alpha_0 = \log f(\alpha). \quad (5.23)$$

Iterative methods such as the CVX package were used in [12] to solve (5.21). The CVX package was discussed in [52] and [53]. Once the probabilities  $\{p_j\}$  are obtained, the source-destination traffic rates can be estimated using (5.11). Let  $\hat{\lambda} = (\hat{\lambda}_1, \dots, \hat{\lambda}_c)'$  denote the estimation of  $\lambda$  given by,

$$\hat{\lambda}_j = Cp_j, \quad j = 1, \dots, c, \quad (5.24)$$

where  $C$  is given by (5.12).

## 5.3 Numerical results

In this section, we studied two networks with  $N = 4$  and  $N = 10$  nodes. The performance of the covariance-based and maximum entropy approaches are evaluated. We compared our results with the approaches of [4] and [2]. The simulations were implemented in Matlab. We tested the approaches on data generated by  $\lambda = (1, 2, \dots, c)'$ , and used  $K = 100$  independent realizations of  $\mathbf{V}$ . The simulation set-up for each approach is described separately.

### 5.3.1 Simulation set-up

We implemented the approach of [4] and [2] given by recursions (2.7) and (2.20), respectively. The initial values for the entries of  $\lambda$  were drawn uniformly from the integers in  $[0, 100]$ . The recursions (2.7) and (2.20) were terminated when  $\sum_j (\lambda_j^{(\ell+1)} - \lambda_j^{(\ell)})^2$  was less than  $10^{-8}$ . The conditional mean estimate in (2.7) was implemented using (2.9) where we choose  $T = 400$ .

The covariance-based approach was implemented, as follows. First, the covariance matrix  $\psi$  and the mean vector  $\eta$  were estimated recursively from (5.4). The initial entries of  $\psi$  and  $\eta$  are set to zeros. The recursive estimation of source-destination traffic rates were obtained using Eq. (5.5) and Eq. (5.6).

In the maximum entropy approach, we used the CVX package developed in [52] to solve (5.21). The source-destination traffic rates were then estimated from (5.24).

### 5.3.2 Results

We evaluated the quality of the estimates obtained from each of the implemented approaches, as follows. First, a set of generated data,  $\mathbf{v}_1^{100}$ , was used for estimating  $\lambda$  denoted by  $\widehat{\lambda}^1 = (\widehat{\lambda}_1^1, \dots, \widehat{\lambda}_c^1)'$ . Next, we generate a new set of  $\mathbf{v}_1^{100}$ , and denote the estimated source-destination rates by  $\widehat{\lambda}^2 = (\widehat{\lambda}_1^2, \dots, \widehat{\lambda}_c^2)'$ . The process was repeated 50 times and the estimated source-destination rates from the  $h$ -th set of data was denoted by  $\widehat{\lambda}^h = (\widehat{\lambda}_1^h, \dots, \widehat{\lambda}_c^h)'$ . Let  $\bar{\lambda} = (\bar{\lambda}_1, \dots, \bar{\lambda}_c)'$  and  $\sigma^2 = (\sigma_1^2, \dots, \sigma_c^2)'$  denote the mean and the mean squared error of  $\{\widehat{\lambda}^1, \dots, \widehat{\lambda}^{50}\}$  given by

$$\bar{\lambda} = \frac{1}{50} \sum_{\nu=1}^{50} \widehat{\lambda}^\nu, \quad (5.25)$$

$$\sigma_j^2 = \frac{1}{50} \sum_{\nu=1}^{50} (\widehat{\lambda}_j^\nu - \lambda_j)^2, \quad j = 1, \dots, c. \quad (5.26)$$

We evaluated and compared  $\bar{\lambda}$  and  $\sigma^2$  for different approaches. First, consider the network in Fig. 2.6 with  $N = 4$  nodes and  $c = 12$  source-destination pairs. In Fig. 5.1, we have plotted the true values of  $\{\lambda_j\}$  and  $\{\bar{\lambda}_j\}$  as obtained in our study from different implemented approaches. In Fig. 5.2, the entries of  $\sigma^2$  were plotted. The maximum entropy approach provides the best quality of estimation where  $\lambda$  is well estimated by  $\bar{\lambda}$  and the smallest values for  $\sigma_j^2$  were obtained for  $j = 1, \dots, c$ .

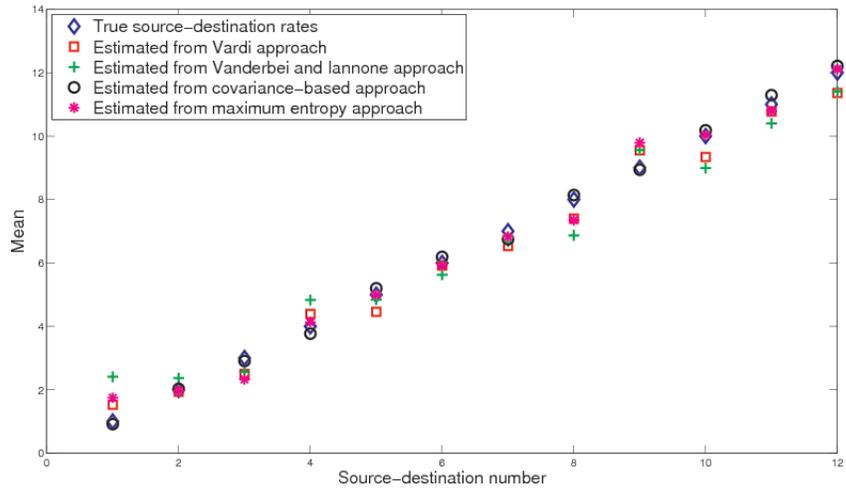


Figure 5.1: The mean of estimated source-destination rates in a network with  $N = 4$  nodes and  $c = 12$  source-destination pairs.

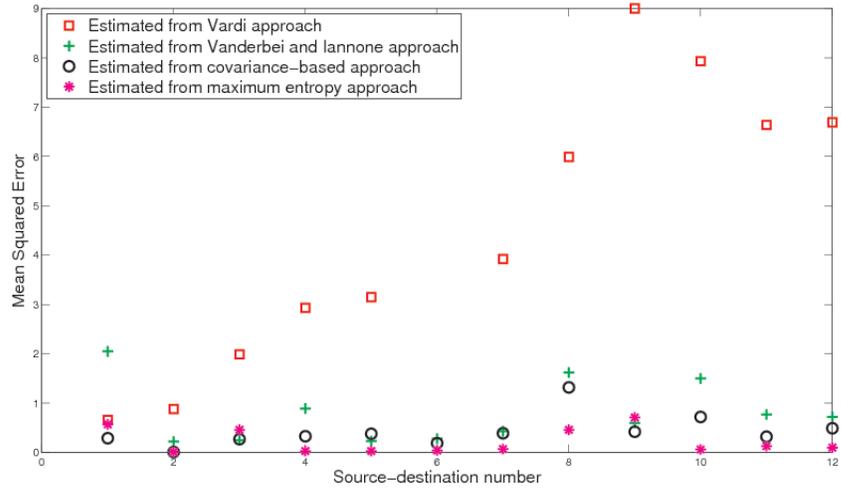


Figure 5.2: The mean squared error of estimated source-destination rates in a network with  $N = 4$  nodes and  $c = 12$  source-destination pairs.

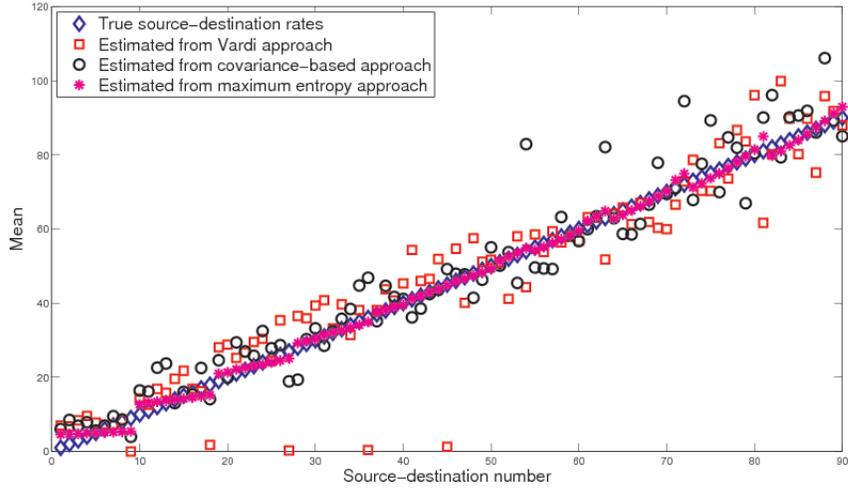


Figure 5.3: The mean of estimated source-destination rates in a network with  $N = 10$  nodes and  $c = 90$  source-destination pairs.

Next, we studied a network with  $N = 10$  nodes and  $c = 90$  source-destination pairs. In Fig. 5.3 and 5.4, we illustrated  $\bar{\lambda}$  and  $\sigma^2$  as obtained from different approaches. It is numerically difficult to implement the recursion developed in [4] as the number of nodes in the network grows. Hence, this recursion was not implemented for the network with  $N = 10$  nodes. Fig. 5.3 and Fig. 5.4 show that the source-destination rates are very well estimated by  $\bar{\lambda}$  as obtained from the maximum entropy approach.

Clearly, there is a trade-off between computational complexity and quality of estimates. The performance of any of the discussed schemes is improved by increasing the number of independent realizations of  $\mathbf{V}$ , i.e.,  $K$ . However, the implementation becomes computationally ineffective for large values of  $K$  except for the covariance-based approach. The covariance matrix,  $\psi$ , can be updated as the independent realizations of  $\mathbf{V}$  become available using (5.4). Next, we studied the network with  $N = 10$  nodes and  $c = 90$  source-destination pairs. The source-destination rates were estimated using  $K = 100$  and  $K = 100,000$  realizations of  $\mathbf{V}$ . Fig. 5.5 shows that the quality of the estimates were improved significantly by increasing  $K$ .

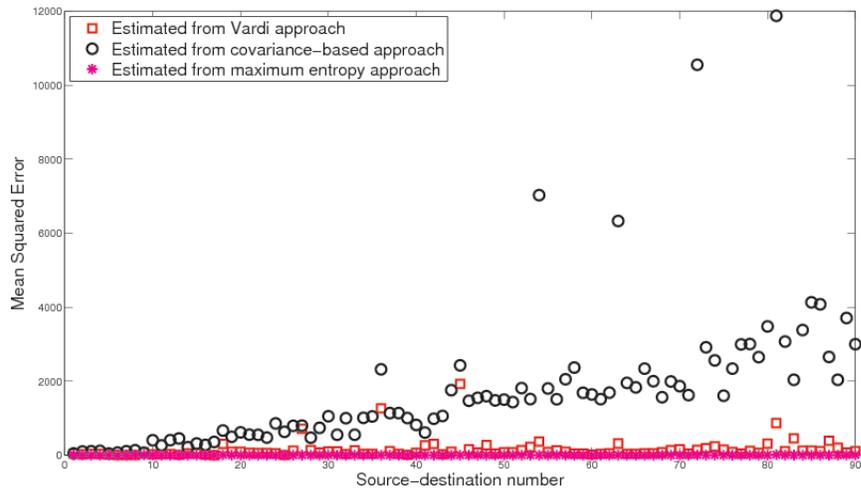


Figure 5.4: The mean squared error of estimated source-destination rates in a network with  $N = 10$  nodes and  $c = 90$  source-destination pairs

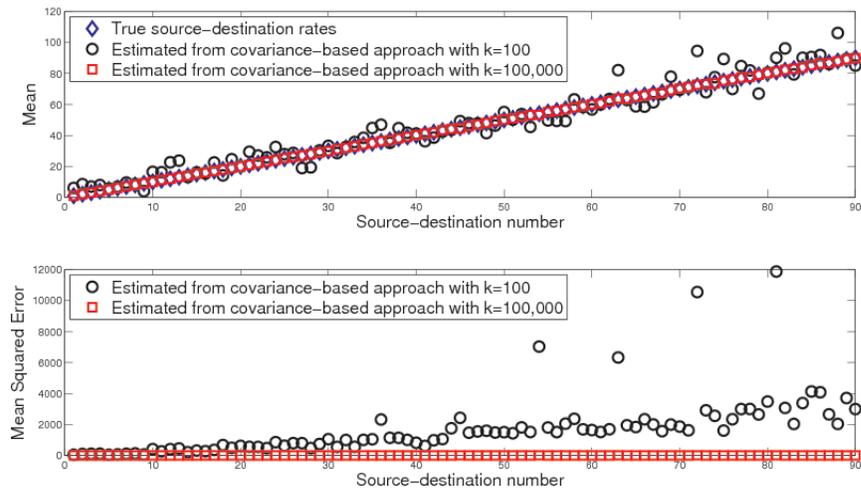


Figure 5.5: The mean and the mean squared error of the estimated rates as obtained from the covariance-based scheme using  $K = 100$  and  $K = 100,000$ .

## Chapter 6: Conclusions and Future directions

In this chapter, we summarize the main contributions of this thesis, and address some of the future directions.

### 6.1 Conclusion

We have studied two main aspects of network tomography, namely, the link delay density estimation from source-destination delay measurements and source-destination traffic rates estimation from aggregated input and output traffic counts.

We have studied link delay density estimation from source-destination delay measurements in both an unstructured network with random routing regime and a tree-structured network with deterministic routing regime. The traffic over the network was modeled as a partially observable bivariate Markov chain. This model implies that the delay on various links has matrix exponential density. The family of matrix exponential densities is rich and includes the family of phase type densities. The approach also provides the packet routing probabilities. We have adopted the EM algorithm of [9] for estimating the parameter of the bivariate Markov chain model. Our approach in estimation of link delay density is general and does not require an independence assumption for the delays on various links as is commonly done in the literature. We have evaluated the performance of the proposed approach in a numerical study, and compared our results with the mixture fitting approach of [11] for a single route of a tree-structure network. We have also developed an online algorithm for implementation of the EM algorithm. The online algorithm applies to blocks of data in a sequential manner.

We have formulated the problem of source-destination traffic rates estimation using the model of [4]. We have developed a new covariance-based approach. This simple and

practical approach relies on the relation between the covariances of the measurements and the rates. We have also developed a maximum entropy approach for the rate estimation problem. We have evaluated the performance of the two proposed approaches in a numerical study, and compared our results with the EM algorithm of [4] and the method of moments of [2].

## 6.2 Future directions

In this section, we address some possible extensions of the work presented in this thesis. The delay network tomography problem can also be extended in the following directions. In real scenarios, the delay on each link has finite support. Hence, we can consider using Markov models with finite support phase-type distributions. The phase type distributions with finite support was studied in [54].

The source-destination traffic rates estimation problem can be extended in the following directions. In this work, we have assumed that the density of traffic count over various source-destination pairs is Poisson. A possible extension of this work is to model the traffic over source-destination pairs using Markov modulated Poisson process which is more general than the Poisson process. Also, the source-destination traffic counts were assumed independent. However, in some real networks, this independence assumption is impractical as the traffic flow over various source-destination pairs may not be independent.

## Appendix A: An Appendix

Since the generator matrix  $G$  is a large matrix of size  $88 \times 88$ , we express only the first  $10 \times 10$  block,  $G_{11}$ , as follows:

$$G_{11} = \begin{bmatrix} -4099.7 & 83.1 & 58.5 & 54.9 & 91.7 & 28.6 & 75.7 & 75.4 & 38.05 & 56.8 \\ 57.5 & -4179.3 & 23.5 & 35.3 & 82.1 & 1.5 & 4.3 & 16.9 & 64.9 & 73.2 \\ 67.9 & 39.5 & -4394.6 & 98.8 & 3.8 & 88.5 & 91.3 & 79.6 & 9.9 & 26.2 \\ 26.9 & 42.3 & 54.8 & -4190.3 & 41.8 & 98.3 & 30.2 & 70.1 & 66.6 & 53.9 \\ 63.8 & 95.8 & 24.08 & 67.6 & -4339.7 & 67.2 & 69.52 & 6.8 & 25.5 & 22.4 \\ 40.4 & 44.8 & 36.6 & 76.4 & 62.8 & -4576.5 & 93.3 & 97.3 & 19.2 & 13.9 \\ 0.06 & 86.6 & 61.3 & 99 & 52.8 & 47.96 & -4273.9 & 22.8 & 49.8 & 90.09 \\ 46.7 & 64.8 & 2.5 & 84.2 & 55.9 & 85.4 & 34.8 & -3891.2 & 5.4 & 17.7 \\ 78.02 & 33.8 & 60.8 & 74.1 & 10.5 & 12.8 & 54.96 & 48.5 & -3936.8 & 79.9 \\ 42.3 & 65.6 & 72.3 & 53.1 & 10.9 & 63.2 & 12.7 & 13.4 & 9.9 & -4030.2 \end{bmatrix}$$

Also, the initial distribution  $\nu$  is given in blocks of  $1 \times 10$  vectors  $\nu_a, a \in \mathbb{X}$  as follows:

$$\nu_1 = [0.019 \quad 0.022 \quad 0.003 \quad 0.022 \quad 0.015 \quad 0.003 \quad 0.007 \quad 0.013 \quad 0.023 \quad 0.023]$$

$$\nu_2 = [0.004 \quad 0.023 \quad 0.023 \quad 0.012 \quad 0.019 \quad 0.004 \quad 0.01 \quad 0.022 \quad 0.019 \quad 0.023]$$

$$\nu_3 = [0.016 \quad 0.001 \quad 0.02 \quad 0.022 \quad 0.016 \quad 0.018 \quad 0.018 \quad 0.01 \quad 0.016 \quad 0.004]$$

$$\nu_4 = [0.017 \quad 0.001 \quad 0.007 \quad 0.002 \quad 0.003 \quad 0.02 \quad 0.017 \quad 0.008 \quad 0.023 \quad 0.001]$$

$$\nu_5 = [0.011 \quad 0.009 \quad 0.018 \quad 0.019 \quad 0.005 \quad 0.012 \quad 0.011 \quad 0.016 \quad 0.017 \quad 0.018]$$

$$\nu_6 = [0.007 \quad 0.016 \quad 0.016 \quad 0.004 \quad 0.003 \quad 0.012 \quad 0.023 \quad 0.008 \quad 0.014 \quad 0.006]$$

$$\nu_7 = [0.018 \quad 0.006 \quad 0.012 \quad 0.017 \quad 0.021 \quad 0.023 \quad 0.013 \quad 0.004 \quad 0.004 \quad 0.006]$$

$$\nu_8 = [0.02 \quad 0.006 \quad 0.019 \quad 0.006 \quad 0.022 \quad 0.009 \quad 0.005 \quad 0.006 \quad 0.015 \quad 0.012]$$

## Bibliography

## Bibliography

- [1] C. Tebaldi and M. West, “Bayesian inference on network traffic using link count data (with discussion),” *J. Amer. Stat. Assoc.*, vol. 93, no. 442, pp. 557–576, June 1998.
- [2] Y. Vardi, “Network tomography: Estimating SD traffic intensities from link data,” *J. Amer. Stat. Assoc.*, vol. 91, pp. 365–377, 1996.
- [3] D. P. Bertsekas and R. G. Gallager, *Data networks*. Prentice-hall, New Jersey, 2nd edition, 1992.
- [4] R. J. Vanderbei and J. J. Iannone, “An EM approach to OD matrix estimation,” *Technical Report SOR 94-04, Princeton University*, 1994.
- [5] L. Shepp and Y. Vardi, “Maximum likelihood reconstruction for emission tomography,” *IEEE Trans. Med. Imag.*, vol. 1, no. 2, pp. 113–122, 1982.
- [6] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, “Network tomography: recent developments,” *Statistical science*, vol. 19, no. 3, pp. 499–517, Aug. 2004.
- [7] Y. Ephraim and B. L. Mark, “Bivariate Markov processes and their estimation,” *Foundations and Trends in Signal Processing*, vol. 6, no. 1, pp. 1–95, 2013.
- [8] Q. M. He and H. Zhang, “On matrix exponential distributions,” *Advances in Applied Probability*, pp. 271–292, Mar. 2007.
- [9] S. Asmussen, O. Nerman, and M. Olsson, “Fitting phase-type distributions via the EM algorithm,” *Scand. J. Stat.*, vol. 23, no. 4, pp. 419–441, Dec. 1996.
- [10] C. F. V. Loan, “Computing integrals involving the matrix exponential,” *IEEE Trans. Autom. Cont.*, vol. 23, no. 3, pp. 395–404, June 1978.
- [11] Y. Xia and D. Tse, “Inference of link delay in communication networks,” *IEEE Trans. Selected Areas in Communications*, vol. 24, no. 12, pp. 2235–2248, Dec. 2006.
- [12] O. Gurewitz, I. Cidon, and M. Sidi, “One-way delay estimation using network-wide measurements,” *IEEE Trans. Inform. Theory*, vol. 52, no. 6, June 2006.
- [13] M. H. Firooz and S. Roy, “Network tomography via compressed sensing,” in *Proc. IEEE Globecom*, pp. 1–5, Dec. 2010.
- [14] N. G. Duffield and F. L. Presti, “Multicast inference of packet delay variance at interior network links,” in *Proc. Infocom*, vol. 3, pp. 1351–1360, Mar. 2000.

- [15] —, “Network tomography from measured end-to-end delay covariance,” *IEEE/ACM Trans. Networking*, vol. 12, no. 6, pp. 978–992, Dec. 2004.
- [16] F. L. Presti, N. G. Duffield, J. Horowitz, and D. F. Towsley, “Multicast-based inference of network-internal delay distributions,” *IEEE/ACM Trans. Networking*, vol. 10, no. 6, pp. 761–775, Dec. 2002.
- [17] E. Lawrence, G. Michailidis, and V. N. Nair, “Network delay tomography using flexicast experiments,” *J. Royal Stat. Soc.: Series B*, vol. 68, no. 5, pp. 785–713, 2006.
- [18] —, “Statistical inverse problems in active network tomography,” *J. Amer. Stat. Assoc. : IMS Lecture Notes-Monograph Series Complex Datasets and Inverse Problems: Tomography, Networks and Beyond*, vol. 54, pp. 24–44, 2007.
- [19] M. F. Shih and A. O. Hero, “Unicast-based inference of network link delay distributions with finite mixture models,” *IEEE Trans. Signal Processing*, vol. 51, no. 8, pp. 2219–2228, Aug. 2003.
- [20] A. Chen, J. Cao, and T. Bu, “Network tomography: Identifiability and Fourier domain estimation,” *IEEE Trans. Signal Processing*, vol. 58, no. 12, pp. 6029–6039, Dec. 2010.
- [21] K. Deng, Y. Li, W. Zhu, Z. Geng, and J. S. Liu, “On delay tomography: Fast algorithms and spatially dependent models,” *IEEE Trans. Signal Processing*, vol. 60, no. 11, pp. 5685–5697, Nov. 2012.
- [22] N. G. Duffield, F. L. Presti, V. Paxson, and D. F. Towsley, “Inferring link loss using striped unicast probes,” vol. 2, pp. 915–923, 2001.
- [23] M. Coates, R. Castro, R. Nowak, M. Gadhiok, R. King, and Y. Tsang, “Maximum likelihood network topology identification from edge-based unicast measurements,” in *Proc. ACM SIGMETRICS Measurement and Modeling of Computer Systems*, vol. 30, no. 1, pp. 11–20, June 2002.
- [24] N. G. Duffield, F. L. Presti, V. Paxson, and D. F. Towsley, “Network loss tomography using striped unicast probes,” *IEEE/ACM Trans. Networking*, vol. 14, no. 4, pp. 697–710, Aug. 2006.
- [25] A. Mukherjee, “On the dynamics and significance of low frequency components of internet load,” *Internetworking: Research and Experience*, vol. 5, pp. 163–205, Dec. 1994.
- [26] D. L. Snyder, T. J. Schulz, and J. A. O’Sullivan, “Deblurring subject to nonnegativity constraints,” *IEEE Trans. Sig. Proc.*, vol. 40, no. 5, pp. 1143–1150, May 1992.
- [27] Y. Vardi and D. Lee, “From image deblurring to optimal investments: maximum likelihood solutions for positive linear inverse problems (with discussion),” *J. Royal Stat. Soc.*, vol. 55, no. 3, pp. 569–612, 1993.
- [28] L. Shepp, Y. Vardi, and L. Kaufman, “A statistical model for positron emission tomography,” *J. Amer. Stat. Assoc.*, vol. 80, no. 389, pp. 8–20, Mar. 1985.

- [29] Y. Vardi, S. E. Levinson, and L. Shepp, “Applications of the EM algorithm to linear inverse problems with positivity constraints,” *Image models (and their speech model cousins)*, vol. 80, pp. 183–198, 1996.
- [30] J. Cao, D. Davis, S. V. Wiel, and B. Yu, “Time-varying network tomography,” *J. Amer. Stat. Assoc.*, vol. 95, no. 452, pp. 1063–1075, Dec. 2000.
- [31] Y. Zhang, M. Roughan, C. Lund, and D. Donoho, “An information-theoretic approach to traffic matrix estimation,” in *Proc. ACM Sigcomm*, pp. 301–312, Aug. 2003.
- [32] G. Liang and B. Yu, “Maximum pseudo likelihood estimation in network tomography,” *IEEE Trans. Sig. Proc.*, vol. 51, no. 8, pp. 2043–2053, Aug. 2003.
- [33] R. Cáceres, N. G. Duffield, J. Horowitz, and D. F. Towsley, “Multicast-based inference of network-internal loss characteristics,” *IEEE Trans. Inform. Theory*, vol. 45, no. 7, pp. 2462–2480, Nov. 1990.
- [34] N. G. Duffield, J. Horowitz, D. F. Towsley, W. Wei, and T. Friedman, “Multicast-based loss inference with missing data,” *IEEE J. Selected Areas in Comm.*, vol. 20, no. 4, pp. 700–713, May 2002.
- [35] N. G. Duffield, “Network tomography of binary network performance characteristics,” *IEEE Trans. Inform. Theory*, vol. 52, no. 12, pp. 5373–5388, Dec. 2006.
- [36] M. Rabbat, R. Nowak, and M. Coates, “Multiple source, multiple destination network tomography,” vol. 3, pp. 1628–1639, Mar. 2004.
- [37] N. G. Duffield, J. Horowitz, F. L. Presti, and D. F. Towsley, “Multicast topology inference from measured end-to-end loss,” *IEEE Trans. Inform. Theory*, vol. 48, no. 1, pp. 26–45, Jan. 2002.
- [38] N. Duffield, J. Horowitz, and F. L. Presti, “Adaptive multicast topology inference,” *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. in Proc. Infocom*, vol. 3, pp. 1636–1645, Apr. 2001.
- [39] M. F. Shih and A. O. Hero, “Network topology discovery using finite mixture models,” *Proc. IEEE ICASSP*, vol. 2, pp. ii–433ii–436, May 2004.
- [40] L. Breiman, *Probability*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1992.
- [41] F. Ball and G. F. Yeo, “Lumpability and marginalisability for continuous-time Markov chains,” *J. Applied Prob.*, vol. 30, no. 3, pp. 518–528, Sep. 1993.
- [42] B. L. Mark and Y. Ephraim, “An EM algorithm for continuous-time bivariate Markov chains,” *Comput. Stat. Data Analysis*, vol. 57, no. 1, pp. 504–517, 2013.
- [43] Y. Ephraim and B. L. Mark, “Consistency of maximum likelihood parameter estimation for bivariate Markov chains,” *Stochastic Models*, vol. 29, no. 1, pp. 89–111, 2013.
- [44] N. Etemadi-Rad, Y. Ephraim, and B. L. Mark, “Delay network tomography using a partially observable bivariate Markov chain,” *submitted to IEEE Trans. Networking*, July 2015.

- [45] R. W. Wolff, “Stochastic modelling and the theory of queues,” *Prentice Hall*, 1989.
- [46] A. Albert, “Estimating the infinitesimal generator of a continuous time, finite state Markov process,” *Annals Math. Stat.*, no. 2, pp. 727–753, June 1962.
- [47] C. Moler and C. F. V. Loan, “Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later,” *SIAM review*, vol. 45, no. 1, pp. 3–49, 2003.
- [48] W. J. J. Roberts, Y. Ephraim, and e. Dieguez, “On rydén’s EM algorithm for estimating MMPPs,” *IEEE Sig. Proc. Let.*, vol. 13, no. 6, pp. 373–376, June 2006.
- [49] S. Kullback and R. A. Leibler, “On information and sufficiency,” *Annals Math. Stat.*, vol. 22, no. 1, pp. 79–86, 1951.
- [50] C. Stiller and G. Radons, “Online estimation of hidden Markov models,” *IEEE Sig. Proc. Let.*, vol. 6, no. 8, pp. 213–215, 2004.
- [51] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.
- [52] M. Grant and S. Boyd, “CVX: Matlab software for disciplined convex programming, version 2.0 beta,” Sep. 2012.
- [53] M. C. Grant and S. P. Boyd, “Graph implementations for nonsmooth convex programs,” *Recent advances in learning and control*, pp. 95–110, 2008.
- [54] V. Ramaswami and N. C. Viswanath, “Phase type distributions with finite support,” *Stochastic Models*, vol. 30, no. 4, pp. 576–597, Nov. 2014.

## Curriculum Vitae

Neshat Etemadi Rad was born in Mashhad, Iran. She received her MS and BS degrees in Electrical Engineering from Amirkabir University of Technology (Tehran Polytechnic) and Sharif University of Technology in Tehran, Iran in 2008 and 2010, respectively. She moved to U.S. in Aug. 2011, and continued her education and received her PhD in Electrical and Computer Engineering at George Mason University, in 2015. Her main areas of interest are statistical signal processing and wireless communications.