# Network Traffic Modelling and Analysis

*Wei Liu*

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

August 2004

A thesis submitted to the Faculty of Graduate Study and Research in partial fulfillment of the requirements for the degree of Master of Engineering.

*Your file Votre référence*
*ISBN: 0-494-12627-2*
*Our file Notre référence*
*ISBN: 0-494-12627-2*

# Abstract

In all-photonic networks, both transmission and switching is performed in the optical domain, without optoelectronic conversion for the data traversing the network. An accurate traffic model is critical in an agile all-photonic network (AAPN) which has the ability to dynamically allocate bandwidth to traffic flows as the demand varies.

This thesis focuses on traffic modelling and analysis. A novel traffic model is proposed which can capture the traffic behaviours in all-photonic networks. The new model is based on a study of existing traffic modelling literature. It combines the time-varying Poisson model, gravity model and fractional Gaussian noise. This model can be used for the short-range traffic prediction. We examine Long-Range Dependence and test the time constancy of scaling parameters using the tools designed by Abry and Veitch, to analyze empirical and synthesized traffic traces.

## Sommaire

Dans les réseaux tout photoniques, les commutations et la transmission de données sont réalisées dans le domaine optique, et ce, sans conversion optoélectronique des données traversant le réseau. Cette particularité rend essentiel l'élaboration d'un modèle de trafic précis pour les réseaux agiles tout photoniques (RATP) en permettant d'allouer dynamiquement les bandes fréquentielles au trafic selon les variations de la demande.

La contribution de cette thèse consiste en l'analyse et la modélisation du trafic. Un nouveau modèle de trafic est proposé, celui-ci permet la capture du comportement du trafic dans les réseaux tout photoniques. L'élaboration du nouveau modèle est basée sur les modèles de trafic déjà existant présent dans la littérature scientifique. Il combine le modèle à temps variant de Poisson, le modèle de gravité ainsi que le bruit gaussien fractionnel. Le modèle proposé peut être utilisé pour des prédictions de trafic sur courtes distances. Nous examinons les dépendances sur longues distances et testons la constance des paramètres d'échelles en utilisant les outils développés par Abry et Veitch, ceci afin de synthétiser et d'analyser empiriquement les tracés de trafic.

# Acknowledgments

I would like to acknowledge my supervisor, Prof. Mark Coates, for his indispensable guidance and invaluable advice throughout my graduate studies at McGill University. I am also grateful to Prof. Coates for providing financial assistance to complete this research. I further would like to express my gratitude to Prof. Lorne Mason for the suggestions in traffic modelling and prediction.

I wish also to thank Garrick Ing for reading through my thesis and providing comments. I appreciate Jean-François Marceau for his help with the translation of my abstract.

I would like to express my deepest gratitude to my husband, Wei Tang for his wonderful love and support. His patience and understanding during my graduate studies were greatly appreciated. Finally, I am sincerely indebted to my parents for their everlasting encouragement.

# Contents

1 Introduction 1

1.1 Problem Statement . . . 1

1.2 Thesis Organization . . . 3

2 Literature Review 4

2.1 Network Tomography . . . 4

2.1.1 Link-level Parameter Estimation . . . 5

2.1.2 Path-level Parameter Estimation . . . 7

2.2 Traffic Modelling . . . 10

2.2.1 Voice in Telephony Networks . . . 10

2.2.2 Data Networks . . . 11

2.3 Traffic Prediction . . . 18

2.3.1 Telephony Network Traffic Prediction . . . 18

2.3.2 Data Network Traffic Prediction . . . 19

2.3.3 Prediction Techniques . . . 20

3 Traffic Models 27

3.1 Introduction . . . 27

3.2 Time-Varying Poisson Process . . . 27

3.3 Gravity Model . . . 29

3.4 Fractional Brownian Motion and Fractional Gaussian Noise . . . 30

3.5 Combined Model . . . 31

3.6 Statistical Analysis . . . 33

3.7 Examples of Synthesized Traffic . . . 35

4 Data Analysis 42

4.1 Empirical Traffic Traces . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42

4.2 Normalized Correlation Function Analysis . . . . . . . . . . . . . . . . . . 43

4.3 Visual Interpretation of Empirical Traffic Traces . . . . . . . . . . . . . . . . 47

4.4 Wavelet-based Estimation of Long-Range Dependence . . . . . . . . . . . . 47

4.4.1 The Long-Range Dependence Phenomenon . . . . . . . . . . . . . . 47

4.4.2 Discrete Wavelet Transform . . . . . . . . . . . . . . . . . . . . . . . 49

4.4.3 A Wavelet Based Joint Estimator of the Parameters of Long-Range Dependence . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 50

4.4.4 Traffic Trace Analysis . . . . . . . . . . . . . . . . . . . . . . . . . 52

4.5 Testing Time Constancy of Scaling Parameters . . . . . . . . . . . . . . . . 57

4.5.1 Definition of the Test on Time Constancy of Scaling Parameters . . 57

4.5.2 Choosing the number of blocks $m$ . . . . . . . . . . . . . . . . . . . 59

4.5.3 Measured traffic trace analysis . . . . . . . . . . . . . . . . . . . . . 59

5 Conclusions and Future Work 67

5.1 Thesis Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 67

5.2 Future Work: Application of the model in Agile All-Photonic Networks . . 68

References 72

# List of Figures

1.1 The overlaid star architecture of the AAPN network. . . . . . . . . . . . . . . . . . . 2

2.1 Tree structured network topology. Packets are sent from one source to a number of destinations. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

2.2 Packet train traffic model. A packet train is a burst of packets between an Origin-Destination pair. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12

2.3 Markov Modulated Poisson Process, constructed by varying the arrival rate of a Poisson process according to an $m$-state Markov chain. . . . . . . . . . . . . . . . . . . . 13

2.4 Wavelet examples (a) Haar scaling function $\phi_{j,k}(t)$ and wavelet function $\psi_{j,k}(t)$. (b) Binary tree of scaling and wavelet coefficients. . . . . . . . . . . . . . . . . . . . . 17

3.1 Graphical representation of the dynamic parametric traffic model. . . . . . . . . . . 28

3.2 The key elements of the proposed dynamic parametric traffic model. . . . . . . . . . 32

3.3 The mean rate of the total traffic flowing into origin node 1, $\sigma_1^2 = 1 \times 10^4$. . . . . . . . 36

3.4 The effect of the parameter $a$ on the synthesized traffic, $\sigma_1^2 = 10^4$. . . . . . . . . . . 37

3.5 Examples of the incremental sequence of fractional Brownian motion, with Hurst parameter $H = 0.3$ and $H = 0.8$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38

3.6 The effect of varying the Hurst parameters $H_{ij}$. . . . . . . . . . . . . . . . 39

3.7 The effect of varying the parameters $A_{ij}$. . . . . . . . . . . . . . . . . . . . . 40

3.8 The effect of variation combination of all parameters. . . . . . . . . . . . . 41

4.1 Total incoming traffic measured at Columbia University. . . . . . . . . . . . . . . . . 44

4.2 Normalized Correlation Function (NCF) of Columbia traffic traces. . . . . . . . . . . 45

4.3 Normalized Correlation Function (NCF) of examples of the synthesized traffic generated by the proposed model. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 46

4.4 The empirical Columbia traffic traces, to different destination groups. . . . . . . . . . 48

4.5 Goodness of fit $Q(j_1)$ as a function of $j_1$, based on the empirical Columbia traffic traces. 53

4.6 Goodness of fit $Q(j_1)$ as a function of $j_1$, based on the synthesized traffic traces. . . . . 54

4.7 Logscale diagram, based on the empirical Columbia traffic traces. . . . . . . . . . . . 55

4.8 Logscale diagram, based on the synthesized traffic traces. . . . . . . . . . . . . . . . . 56

4.9 Goodness of fit $Q$ for each block, based on the empirical Columbia traffic traces. . . . . 61

4.10 Stationarity testing of Columbia1 traffic trace. . . . . . . . . . . . . . . . . . . . . . 62

4.11 Stationarity testing of Columbia2 traffic trace. . . . . . . . . . . . . . . . . . . . . . 63

4.12 Goodness of fit $Q$ for each block of the synthesized traffic traces. . . . . . . . . . . . 64

4.13 Stationarity testing of traffic traces generated by the proposed model with parameters $\sigma_i^2 = 1 \times 10^4$, $\gamma_{ij}^2 = 1 \times 10^3$, $a = 0.8$, $A_{ij} = 1$, $H_{ij} = 0.3$. . . . . . . . . . . . . . . . . 65

4.14 Stationarity testing of the traffic traces generated by the proposed model with parameters $\sigma_i^2 = 1 \times 10^5$, $\gamma_{ij}^2 = 1 \times 10^4$, $a = 0.2$, $A_{ij} = 100$, $H_{ij} = 0.8$. . . . . . . . . . . . . . . 66

# Chapter 1

# Introduction

## 1.1 Problem Statement

In current optical networks[1], the electronic processing in the switches is a bottleneck, in the sense that it restricts the switching capacity because the optoelectronic conversion is slow [1]. In an all-photonic (all-optical) network, the electronic switches are replaced by optical switches. There may still be electronic control, but the data traversing the photonic node do not experience optoelectronic conversion. Current designs of all-photonic networks permit routing and multiplexing based on wavelengths. However, the reconfiguration time of these networks, which can be as long as several seconds, restricts the incorporation of time-domain multiplexing [2]. The Agile All-Photonic Network (AAPN) will introduce both an all-photonic core and agility, which specifies the ability to perform dynamic bandwidth allocation as the amount of traffic varies over time, at a time scale of micro- or milliseconds [3].

Currently, many core networks have a mesh topology because it is robust and can distribute traffic load over switches. Due to the lack of optical memory, all-photonic networks cannot afford the frequent resource contention at switches. In order to avoid the contention, the AAPN uses the overlaid star topology, as shown in Fig. 1.1. The edge nodes are connected to several central photonic core nodes, which provides protection in case that some

[1] Optical networks are high-capacity telecommunications networks based on optical technologies and components that provide routing, grooming, and restoration at the wavelength level as well as wavelength-based services. The examples of current optical networks are synchronous optical network (SONET) and wavelength division multiplexing (WDM).

link or core node fails.

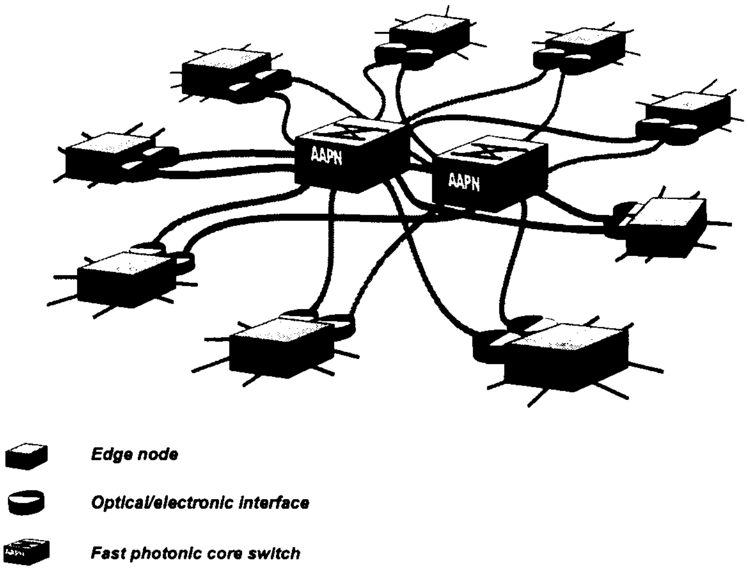


**Fig. 1.1** The overlaid star architecture of the AAPN network.

In the AAPN, control functionality is implemented primarily at the edge of the photonic core, which simplifies the scheduling of traffic flows within the core. The edge nodes will perform all complicated control functionalities, e.g. traffic aggregation, time-domain division of a single wavelength, and sharing the bandwidth of a single wavelength between several information flows. The edge nodes can execute these functionalities most efficiently if they have the most up-to-date information concerning the properties of the entire network, e.g. link loss rates, delay distributions and traffic demands. However, in a large scale network, the edge nodes are geographically distributed, potentially belong to different subnetworks, and thus have only limited and delayed network state information.

The distributed edge nodes cannot measure and observe the performance of the global network. In order to implement efficient control schemes, the edge nodes need to predict the future demands of the network based on the observable but incomplete and delayed measurements. The primary objective in this thesis is to propose a traffic modelling approach, which can capture the traffic characteristics in an all-photonic network, and identify the methods to predict the traffic demands based on the proposed traffic model and available network measurements. If network control decisions can be made based on reliable estimation of the current and future network traffic demands, then more effective switching strategies, efficient scheduling and bandwidth allocation strategies, and better traffic control protocols and dynamic routing algorithms can be designed. The methods of traffic modelling and prediction will be based on techniques and models proposed in network tomography, network traffic modelling and statistical signal processing.

## 1.2 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 reviews the recent literature in the fields of network tomography, network traffic modelling and traffic prediction. The chapter first provides a generalized formulation of the network tomography problem, and discusses link-level and path-level network performance parameter estimation. Then we survey network traffic models that have been presented in the literature. This chapter also covers some techniques in traffic prediction in both telephony and data networks. Chapter 3 proposes a new traffic modelling approach which can be used to capture the traffic behaviours in an AAPN. It describes the derivation of this model and presents some traffic examples generated by this model. Chapter 4 presents the comparison between the synthesized traffic traces generated by the proposed model and the empirical traffic traces measured in the actual network. We also test the long-range dependence and time constancy of scaling parameters of the synthesized traffic and actual traffic, using the tools proposed by Abry and Veitch [4]. Finally, Chapter 5 summarizes the conclusions of the thesis and presents the application of the proposed model in Agile All-Photonic Networks. Suggestions for future research directions are discussed.

# Chapter 2

# Literature Review

In this chapter, we explain the problem of network tomography, and present some solutions for link-level and path-level network parameter estimation. Then we review some traffic models appearing in literature, including both short-range dependent and long-range dependent. This chapter also describes some methods in traffic prediction in both telephony and data networks, and identifies some prediction techniques such as Kalman filtering, extended Kalman filtering and sequential Monte Carlo methods.

## 2.1 Network Tomography

In a large-scale AAPN, distributed edge nodes cannot directly acquire complete network information to guide control functions such as dynamic routing, scheduling and bandwidth sharing. It is also impossible to take detailed measurements at all nodes in the network due to the financial cost, administrative burden and the delay induced by processing, storing and transmitting the data. Edge nodes can, however, estimate network performance parameters based on available traffic measurements at a subset of the nodes in a large scale AAPN. The estimation task is called "network tomography" [5], due to the similarity between such network inference problems and medical tomography.

In general, network tomography problems can be divided into two groups [6,7]: link-level parameter estimation based on path-level traffic measurements, as described in [8–12], and path-level traffic intensity estimation based on link-level traffic measurements, as described in [5, 13–16].

### 2.1.1 Link-level Parameter Estimation

Link-level network tomography is the task of estimating the link-level network performance parameters (for example, loss rates and delay distributions) from path-level measurements, which typically consist of counts of packets or time delay between edge nodes. The measured path delay is the sum of the link delays in the path the traffic traverses, and includes the propagation delay and processing/queuing delays in the routers along the path.

Consider the network in Fig. 2.1. Packets are sent from one source to a number of destinations. By coordinating measurement scheme between the source and destinations, the path-level (end-to-end) behaviour can be measured. For instance, the source can measure the loss rates and transmission delay distribution by receiving the acknowledgment from the destinations after receiving packet successfully. In link-level network tomography, there are several key assumptions. The routing matrix is usually assumed to be known and constant during the measurement period.

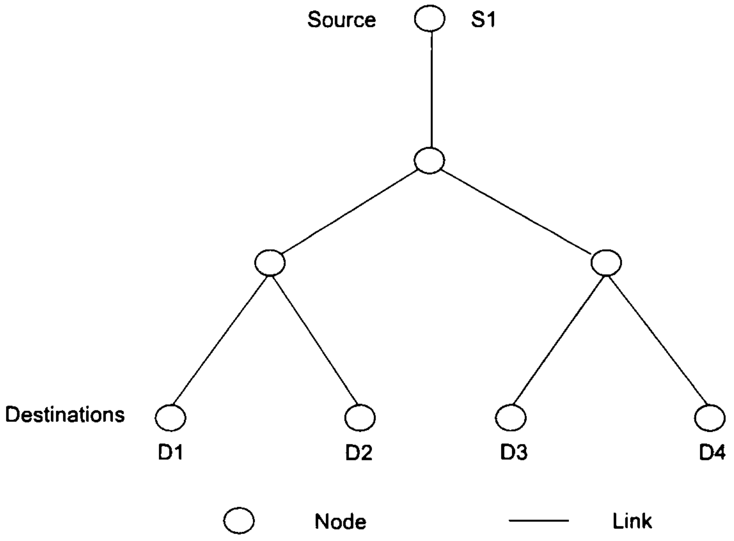


**Fig. 2.1** Tree structured network topology. Packets are sent from one source to a number of destinations.

Multicast and unicast are two common modes of communications in networks. Each packet is sent to only one destination in unicast communication, but to a group of destinations in multicast communication. Link-level network tomography have different approaches enabled by both multicast and unicast technologies.

Network tomography based on multicast probing was one of the first approaches to the problem. The Multicast-based Inference of Network-internal Characteristics (MINC[1]) Project at the University of Massachusetts pioneered the use of multicast probing for network tomography. Caceres *et al.* introduced a method to estimate link-level loss rates within a network through multicast end-to-end measurements [8]. The approach is based on the measurement and analysis of the loss behaviour of multicast probe traffic. A multicast packet is supposed to be received by a specified set of $N$ nodes. If some of these $N$ nodes failed to receive the data, then the loss can be localized to a subset of the links. Multicast traffic introduces correlation in the end-to-end losses, which can be measured by the end systems and be used to infer the loss behaviour of the links within the multicast routing tree between the origin and destination. Upon adoption of a parametric loss model, maximum likelihood estimators can be designed for estimating link loss rates within a multicast tree. The authors of [8] showed that these estimates are strongly consistent and converge almost surely to the true loss rates.

Multicast based network tomography is easy to implement, but since many networks do not support multicast, unicast based network tomography is more practical. However, unicast based tomography has the difficulty that although one can estimate end-to-end path loss rates and delay distributions from single unicast packet measurements, the mapping of these path-level parameters to the link-level parameters is not unique. To address this challenge, Coates *et al.* proposed a new theory and tool for gauging internal network loss characteristics solely from end-to-end measurements using unicast, back-to-back packet pairs [9]. The two packets in one packet pair are sent one after the other by the source possibly to different destinations, but sharing a common set of links in their paths. This method allows one to collect more information that can help to resolve the link-level loss rates and delay distributions. Coates *et al.* analyzed and compared some of the potential choices for conditional prior information models that can be used to resolve the link loss rates, and developed a factor graph framework capable of supporting efficient inference methods. Harfoush *et al.* independently developed a similar algorithm for link loss rate estimation [17].

[1] http://gaia.cs.umass.edu/minc

Duffield *et al.* also proposed a method that uses unicast probing to infer the internal loss characteristics [18]. This method relies on sending a cluster of unicast packets (a packet "stripe") to a set of destinations to mimic the effect of a multicast packet. The methods of multicast inference can then be applied to infer the link loss and delay characteristics of the network.

In [10], Coates *et al.* introduced a method for internal delay estimation based on the end-to-end delay measurements from source to destination. They developed an Expectation Maximization algorithm [19] for maximum likelihood estimation in stationary networks, and a Sequential Monte Carlo procedure [20] for estimation in nonstationary cases. Lo Presti *et al.* presented a recursive algorithm for estimating internal delay distributions by solving a set of convolution equations [21].

The above methods focus on estimation of network performance parameters, with the assumption that the network (routing) topology is known, but the routing matrix is not always available. In [12, 18, 22–25], the problem of topology identification is addressed. Ratnasamy and McCanne [22] presented schemes that allow a receiver to determine both the multicast tree structure and the bottle-neck bandwidth by analyzing correlations of loss patterns across the receiver set and by measuring how the network perturbs the fine-grained timing structure of the packets sent from the source. Duffield *et al.* [18] introduced an algorithm which combines different performance measures and reconstructs the multicast tree topology by adaptively choosing the one that ensures the best accuracy. Coates *et al.* introduced a probing scheme which only requires measurement of delay difference [12]. They developed a Markov Chain Monte Carlo (MCMC) procedure to efficiently identify the topology with the highest likelihood of generating the measurements.

### 2.1.2 Path-level Parameter Estimation

Path-level parameter estimation is the task of estimating path-level network parameters from measurements made on individual links. An example is the problem of estimating the amount of traffic between specified Origin-Destination (OD) pairs, based on measurements of the number of packets that pass through interior nodes in the network.

In path-level traffic intensity estimation, the relationship between Origin-Destination [6] flows and link counts can be described by the following linear relationship:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} \tag{2.1}$$

where $\mathbf{Y} = (y_1, ..., y_m)^T$ denotes the observed column vector of incoming/outgoing byte counts measured on each router link interface during a given time interval. $\mathbf{X} = (x_1, ..., x_n)^T$ denotes the unobserved vector of corresponding byte counts for all Origin Destination pairs during a give time interval in the network. One element of $\mathbf{X}$ corresponds to the number of bytes originating from a specified origin node to a specified destination node. One element of $\mathbf{Y}$ corresponds to all bytes sent from the origin node regardless of their destinations. $\mathbf{A}$ is the routing binary matrix with $m$ rows representing the links of the network and $n$ columns representing the OD pairs. The entries of matrix $\mathbf{A}$, $a_{ij}$'s, are "1" or "0" according to whether link $i$ does or does not belong to the directed path of the OD pair $j$.

The problem of OD (path-level) network tomography was first investigated by Vardi [5]. The OD traffic intensity is estimated from measurements of traffic on the links of a network. Vardi modelled the traffic between OD pairs as being generated from independent and identically distributed Poisson model. He discussed Maximum Likelihood Estimation and a related approximation, and analyzed numerical difficulties. He presented a method-of-moments approach, which can be implemented using an EM algorithm, to estimate the Poisson parameters in two different types of network routing schemes: fixed (deterministic) and random (Markovian).

Tebaldi *et al.* presented a Bayesian approach to path-level network tomography in [13]. The goal is to compute the posterior conditional probability distribution $p(\mathbf{X}|\mathbf{Y})$ of all OD demands given the link counts, using the Bayesian identity $p(\mathbf{X}|\mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{X})p(\mathbf{X})$. In [13], Tebaldi *et al.* modelled the prior distribution $p(\mathbf{X})$ for OD demands using a Poisson distribution, independently over all OD pairs. They then developed posterior distributions for inference on actual origin-destination counts and associated flow rates, using iterative MCMC simulation methods that combine Metropolis-Hastings steps within an overall Gibbs sampling framework.

Cao *et al.* [14] estimated the time-varying traffic matrix based on link byte counts measured at the router interfaces and under a fixed routing scheme, i.e., there is only one route between an Origin-Destination pair. They modelled the unobserved OD byte counts as an normal distribution and assumed that the OD counts are independent and identically distributed over successive measurement periods.

$$x_i \sim N(\lambda_i, \varphi\lambda_i^c), \tag{2.2}$$

where $c$ is a fixed power constant. This implies

$$y \sim N(A\lambda, A\Sigma A^T), \tag{2.3}$$

where $\lambda = (\lambda_1, ..., \lambda_n)^T$, and $\Sigma = \phi \cdot \text{diag}(\lambda_1^c, ..., \lambda_n^c)$. $\lambda > 0$ is the vector of OD mean rates, and $\phi > 0$ is the scale parameter. The authors assumed that the means and variances of the normal distribution are related through a power law, and proved the identifiability of the OD parameters from link data under this assumption. They performed maximum-likelihood estimation the parameters via a combination of the Expectation Maximization (EM) algorithm and a second-order global optimization routine. Cao *et al.* proposed a scalable method that relies on a divide-and-conquer strategy [15]. Traffic matrix estimation for a large network is divided into a number of smaller sub-problems and then each of these sub-problems are solved independently. The computational complexity of each sub-problem can be made independent of the size of the entire network, and the computation burden can be reduced without losing estimation accuracy.

Liang *et al.* proposed a Maximum Pseudo Likelihood Estimation (MPLE) approach in [16]. Combined with MPLE, the divide-and-conquer strategies described in [14, 15] can address large network problems, but the complexity for a network with $n$ nodes is still relatively high at $O(n^2)$. The main idea is to decompose the original problem into a series of simpler sub-problems by selecting pairs of rows from the routing matrix **A** and to form the pseudo-likelihood function by multiplying the marginal likelihoods of such sub-problems. The pseudo-likelihood approach achieves a good balance between the computational complexity and the statistical efficiency of the parameter estimation.

In [26], Zhang *et al.* introduced a method, termed tomogravity, for computing the traffic matrix from link data. The method consists of two basic steps: a gravity modeling step and a tomographic estimation step. A general formulation of a gravity model is given by the following equation:

$$X_{ij} = \frac{R_i A_j}{f_{ij}} \tag{2.4}$$

where $X_{ij}$ is the matrix element representing the force from $i$ to $j$; $R_i$ represents the repulsive factors that are associated with "leaving" from $i$; $A_j$ represents the attractive factors that are associated with "going" to $j$; and $f_{ij}$ is a friction factor from $i$ to $j$. The resulting gravity model simply states that the traffic exchanged between locations is proportional to the volumes entering and exiting at those locations. This model is appropriate only for high volume routers, for which the exchanged aggregated traffic is not sensitive to the detailed composition of the traffic. In the second step, the gravity model solution is refined by using a least-square solution that minimizes the Euclidean distance to the gravity model solution subject to the tomographic constraints.

## 2.2 Traffic Modelling

As new communication services evolve and the needs of users change, modern communication networks like the AAPN are required to support diverse and emerging communication traffic such as data, voice, and video. In order to efficiently accommodate the traffic and maintain acceptable quality of service, effective congestion control schemes are important. Accurate traffic models, which can capture the statistical characteristics of actual traffic, are needed to determine which control techniques, such as congestion control and bandwidth allocation, should be used in AAPN.

### 2.2.1 Voice in Telephony Networks

Voice traffic in telephony networks is relatively easy to model with reasonable accuracy. The temporal dynamics of voice traffic are well-described by the distributions of the inter-arrival times and durations of the calls. Poisson processes are widely used to represent the statistical characteristics of voice traffic [27]. A Poisson process can be characterized as a renewal process whose inter-arrival times of the calls are exponentially distributed with rate parameter $\lambda : P\{A_n \leq t\} = 1 - \exp(-\lambda t)$, where $A_n$ is the length of the time interval separating the $n$th arrival from the previous one. It can also be described as a counting process, satisfying $P\{N(t) = n\} = \exp(-\lambda t)(\lambda t)^n/n!$, where $N(t) = \max\{n : T_n \leq t\}$ is the number of (traffic) arrivals in the interval $(0, t]$. The number of arrivals in disjoint intervals is statistically independent.

### 2.2.2 Data Networks

In [28], Paxson and Floyd showed that for wide-area traffic, Poisson processes are valid only for modelling the arrival of user sessions (TELNET connections, FTP control connections). Other connection arrivals, such as SMTP (email) and NNTP (network news), do not display Poisson characteristics. An exponentially-distributed inter-arrivals model used to model packet arrivals generated by the user side of a TELNET connection seriously underestimates the burstiness of TCP traffic over a wide range of time scales. In a wide-area link, only one or two bursts may occur in an hour, but they will strongly dominate that hour's traffic. Paxson *et al.* suggested abandoning Poisson-based wide-area traffic for all but user session arrivals.

Traffic models can be stationary or nonstationary. The stationary traffic models in data networks can be classified into two classes: short-range and long-range dependent.

Let $\{X_t\}$, $t = 0, 1, 2, ...$ be a wide-sense stationary stochastic process. The process has a stationary mean $\mu = E[X_t]$, a stationary variance $\nu = E[(X_t - \mu)^2]$, and a stationary autocovariance function $\gamma_k = E[(X_t - \mu)(X_{t+k} - \mu)]$. The autocorrelation of the process $\{X_t\}$ at lag $k$ is defined as $\rho_k = \gamma_k/\gamma_0$. The process $\{X_t\}$ is short-range dependent if the autocorrelation decays to zero exponentially, i.e. $\sum_k \rho_k < \infty$. However, if the autocorrelation decays to zero at a slower rate than exponential so that it is not summable $\sum_k \rho_k \longrightarrow \infty$, the process $\{X_t\}$ is long-range dependent.

Short-range dependent models include Markov processes and Regression models, e.g. Markov Modulated Poisson Process (MMPP) and TES models [29,30]. These traffic models have a correlation structure that is significant only for relatively small lags. Long-range dependent traffic models, such as Fractional Autoregressive Integrated Moving Average (F-ARIMA) [31] and Alternating Fractal Renewal Process (AFRP) [32, 33], have significant correlations even for large lags.

#### Short-Range Dependent Traffic Models

The "packet train" traffic model is proposed by Jain *et al.* in [34]. A packet train is defined as a burst of packets between an OD pair as shown in Fig. 2.2. The inter-train gap is a user-defined parameter, dependent on the frequency with which applications use the network. Each train consists of a number of packets going in either direction. The inter-packet gap is a system parameter and depends on the network hardware and software. This model reflects the fact that much of the communication inside a network involves many packets spaced closely in time, exchanged between the same source-destination pair.

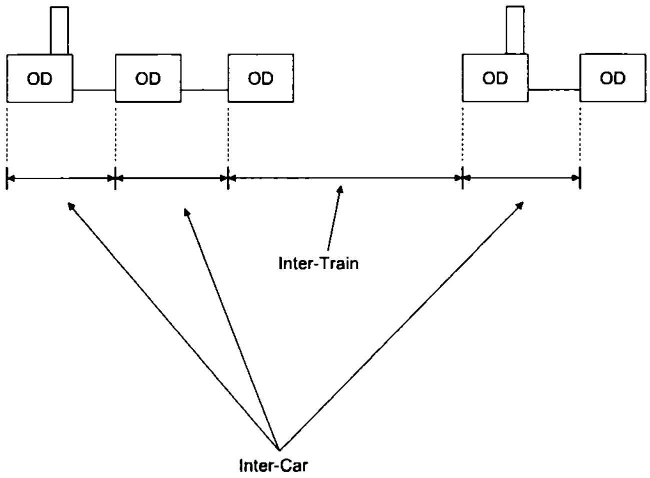


**Fig. 2.2** Packet train traffic model. A packet train is a burst of packets between an Origin-Destination pair.

A Markov-Modulated Poisson Process (MMPP) [29] can be constructed by varying the arrival rate of a Poisson process according to an $m$-state Markov chain which is independent of the arrival process. Let $\mathbf{M} = \{M(t)\}_{t=0}^{\infty}$ be a continuous-time Markov process, with state space $1, 2, ...m$. The modulation mechanism is that in state $k$ of $M$, arrivals occur according to a Poisson process at rate $\lambda_k$. As the state changes, so does the rate, as shown in Fig. 2.3. The MMPP parameters can be estimated from the empirical data, to quantize the arrival rate into a number of rates which corresponds to the number of states. Each rate corresponds to a state in the Markov chain. The transition rate from state $i$ to state $j$, denoted by $q_{ij}$, is estimated by quantizing the empirical data and by calculating the fraction of times that the state (rate) $i$ switched to state (rate) $j$. MMPPs can model a mixture of voice and data traffic. In this case, the arrivals of voice packets while in state $k$ is assumed to be Poisson with rate $\lambda_k$. Data packets are also Poisson with rate $\lambda_d$. The resulting rate of state $s_k$ will be $\lambda_d + \lambda_k$.

Transform-Expand-Sample (TES) is a methodology for traffic modeling that aims to capture both the autocorrelation structure and the nature of the marginal distribution of empirical data [30]. TES models consist of two major processes: $TES^+$ and $TES^-$. $TES^+$

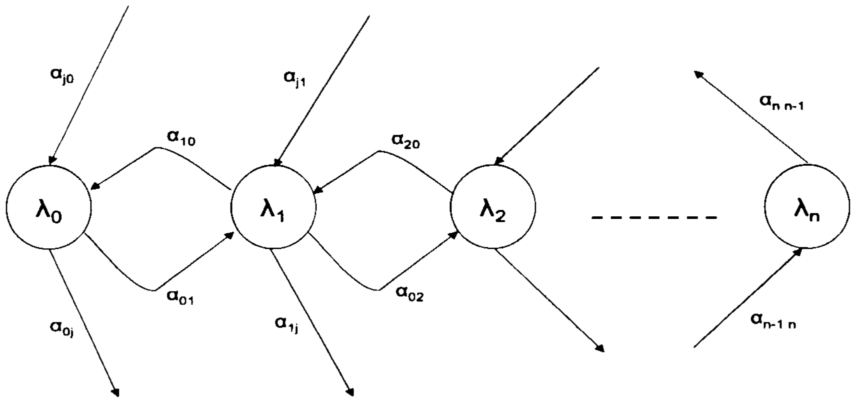


**Fig. 2.3** Markov Modulated Poisson Process, constructed by varying the arrival rate of a Poisson process according to an $m$-state Markov chain.

produces a sequence with positive correlation at lag 1, and $TES^-$ produces a corresponding sequence with negative correlation. $TES^+(L, R)$ is characterized by two parameters, $L$ and $R$. The recursive construction of TES processes is defined as follows:

$$U_n^+ = \begin{cases} U_0 & \text{for } n = 0, \\ \langle U_{n-1}^+ + V_n \rangle & \text{for } n > 0. \end{cases} \tag{2.5}$$

$$U_n^- = \begin{cases} U_n^+ & n \text{ even} \\ 1 - U_n^+, & n \text{ odd.} \end{cases} \tag{2.6}$$

where $V_n$ is a sequence of independent identically distributed random variables, independent from $U_0$. The operator $\langle\cdot\rangle$ is defined for any real $x$ by $\langle x \rangle = x - \lfloor x \rfloor$, where $\lfloor\cdot\rfloor$ is the floor operator. The resulting sequences $U_n^+$ and $U_n^-$ are uniformly distributed in $[0, 1)$. The autocorrelation function of TES processes decays exponentially.

### Long-Range Dependent Traffic Models

Recent empirical studies indicate that the traffic in broadband networks has properties such as long-range dependence (LRD) and heavy-tailed marginal distributions [35–37]. The traditional Poisson models have a correlation that decays exponentially. Willinger *et al.* [35] suggested that the autocorrelation of traffic volumes in modern data networks decays to zero at a slower rate than exponential. This slowly decaying correlation structure is indicative of LRD. Let $\{Y_k\}_{k \in Z}$ be a discrete-time second-order stationary process, with auto-covariance function $\gamma(\tau) := cov(Y_{k+\tau}, Y_k)$. If $\sum_{\tau=-\infty}^{\infty} |\gamma(\tau)| = \infty$, the corresponding stationary process is said to be LRD. The authors of [35] argued that LRD tends to be caused by user and application characteristics and has little to do with the network itself.

Crovella *et al.* [36] presented empirical results indicating that a number of file size distributions in the Web exhibit heavy tails. A random variable $Z$ has a heavy-tailed distribution if

$$Pr\{Z > x\} \sim cx^{-\alpha}, x \rightarrow \infty$$

where $0 < \alpha < 2$ is called the tail index or shape parameter, and $c$ is a positive constant. A distinguishing characteristic of these distributions is that they have infinite variance for $0 < \alpha < 2$. Park *et al.* [38] suggested that there is an intimate causal relationship between the presence of these heavy-tailed distributions in file sizes and the long-range dependent behaviour of network traffic .

Corradi *et al.* analyzed the applicability of f-ARIMA processes to traffic modelling [31]. An $f$-ARIMA$(p, d, q)$ process, $\{x_t\}$, is defined, indirectly, by the relationship

$$\phi(B)\nabla^d x_t = \theta(B)a_t, \tag{2.7}$$

where $a_t$ is a white Gaussian noise series, $\phi(B)$ and $\theta(B)$ are two polynomial functions of degree $p$ and $q$ respectively, i.e. $\phi(B) = 1 - \phi_1 B - \ldots - \phi_p B^p$ and $\theta(B) = 1 - \theta_1 B - \ldots - \theta_q B^q$. The parameters $p$ and $q$ are integers, whereas $B$ indicates the backward-shift operator $(Bx_t = x_{t-1})$. $\nabla^d = (1 - B)^d$ can be expressed using the binomial expansion

$$(1 - B)^d = \sum_{k=0}^{\infty} \binom{d}{k} (-1)^k B^k, \tag{2.8}$$

where $d$ can take values between 0 and 1/2 in order to generate LRD processes. $f$-

ARIMA$(p, d, q)$ models have three parameters $p$, $d$, and $q$ that control the auto-correlation structure. The parameter $d$ determines the LRD properties of the process, whereas $p$, $q$ and the corresponding functions $\phi(B)$ and $\theta(B)$ allow for more flexible modelling of Short-Range Dependent (SRD) properties. Therefore, $f$-ARIMA models can capture both short-range and long-range dependence.

Willinger *et al.* presented an on/off traffic model, based upon an alternating fractal renewal process (AFRP) [32, 33]. In this model, a single source-destination active pair alternates between two states, "ON" periods, during which packets arrive at regular intervals, and "OFF" periods, where there are periods with no packet arrivals. Traditionally, such models assume exponential distributions for their ON- and OFF-periods. However, it has been recognized that multiplexing a large number of sources with such distributions results in aggregate traffic that is inconsistent with actual traffic measurements. In [32, 33] Willinger *et al.* proposed a model that abandoned the traditional exponential distributions for the ON- and OFF- periods of the sources in favour of infinite-variance distributions, such as the Pareto. The incorporation of these distributions gives rise to sources that display the "Noah effect", a term coined by Mandelbrot [39] to describe the tendency of persistent time-series to display abrupt and discrete changes. The superposition of many of these sources leads to aggregate traffic that displays self-similarity and long-range dependence, characteristics observed in many empirical observations of network traffic [35–37]. However, while the AFRP model provides insight into the essential self-similar characteristics of modern high-speed network traffic, its Gaussian aggregated result is inconsistent with real traffic data, which depart greatly from Gaussianity. Yang *et al.* proposed an extended AFRP (EAFRP) to overcome the limitations of AFRP model [40]. They treated the single-user bit rate as a random variable with heavy-tailed characteristics. For both single-user and aggregated traffic, it results in impulsiveness and long-range dependence.

In [41], Bonald and Roberts modelled elastic traffic in terms of flows; a flow is defined as the sequence of packets pertaining to one instance of some application. A flow is simply characterized by an arrival time and a volume of data to be transmitted on a network path. The model in [41] assumes that flows arrive according to a Poisson process with rate $\lambda$. This is appropriate when there are a large number of sessions and the spacing of flows within a session is large compared to the average inter-flow interval. Bonald and Roberts used Pareto distributions to fit the observed heavy tail of flow size

$$Pr[\sigma > x] = (\frac{b}{x})^a, \qquad \text{for } x > b, \tag{2.9}$$

where the random variable $\sigma$ denotes the flow size, $a > 1$ is a fixed parameter and $b$ is the minimum flow size.

### Wavelet Representations

Recent studies show that although network traffic has the complicated short- and long-range temporal dependence, the corresponding wavelet coefficients are all "short-range" dependent [42,43]. Wavelets are complete orthonormal bases which can be used to represent a signal as a function of time. The discrete wavelet transform is a multi-scale signal representation of the form

$$x(t) = \sum_k u_k 2^{-J_0/2}\phi(2^{-J_0}t - k) + \sum_{j=-\infty}^{J_0}\sum_k d_{j,k}2^{-j/2}\psi(2^{-j}t - k), \qquad j, k \in Z \tag{2.10}$$

with $J_0$ the coarsest scale and $u_k$ and $d_{j,k}$ the scaling and wavelet coefficients, respectively. The scaling coefficients may be viewed as providing a coarse approximation of the signal, with the wavelet coefficients providing higher frequency "detail" information.

The Haar scaling function $\phi$ and Haar wavelet function $\psi$ are

$$\phi(t) = \begin{cases} 1 & \text{for } 0 \le t \le 1, \\ 0 & \text{otherwise.} \end{cases} \tag{2.11}$$

$$\psi(t) = \begin{cases} 1 & 0 \le t \le 1/2, \\ -1 & 1/2 \le t \le 1, \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

In the Haar transform, the scaling and wavelet coefficients can be recursively computed using

$$u_{j,2k} = 2^{-1/2}(u_{j+1,k} + d_{j+1,k}),$$
$$u_{j,2k+1} = 2^{-1/2}(u_{j+1,k} - d_{j+1,k}). \tag{2.13}$$

Fig. 2.4(a) shows the Haar scaling and wavelet functions, and Fig. 2.4(b) is the binary tree of scaling coefficients from coarse to fine scales.

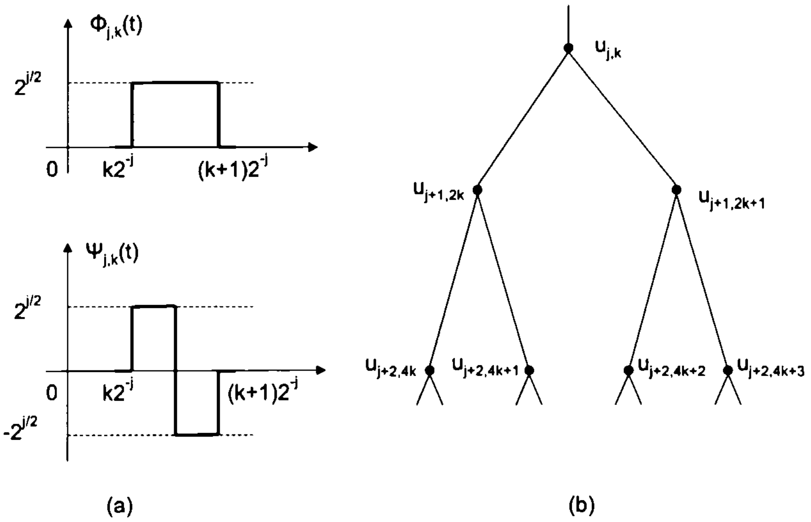


**Fig. 2.4** Wavelet examples (a) Haar scaling function $\phi_{j,k}(t)$ and wavelet function $\psi_{j,k}(t)$. (b) Binary tree of scaling and wavelet coefficients.

Crouse *et al.* presented a multiscale, multiplicative signal model for positive processes [43]. Let $A_{j,k}$ be a random variable supported on the interval $[-1, 1]$ and define the wavelet coefficients recursively by

$$W_{j,k} = A_{j,k} U_{j,k}. \tag{2.14}$$

Then the scaling coefficients can be recursively computed by

$$U_{j,2k} = 2^{-1/2}(1 + A_{j+1,k})U_{j+1,k},$$
$$U_{j,2k+1} = 2^{-1/2}(1 - A_{j+1,k})U_{j+1,k}. \tag{2.15}$$

When reaching the finest scale $j = J_1$ we obtain the desired process $X$, given by

$$X[k] = 2^{J_1/2} U_{J_1,k}, \qquad k = 0, ..., 2^{J_0 - J_1} - 1. \tag{2.16}$$

Ma *et al.* showed that an independent wavelet model is sufficiently accurate, and they developed a time-scale shaping algorithm that extends the (Gaussian) wavelet models to non-Gaussian traffic [42]. The wavelet models have computational complexity $O(N)$ in developing a model from a training sequence of length $N$, and $O(M)$ in generating a synthetic traffic trace of length $M$.

Veitch *et al.* presented a general wavelet based framework for the on-line generation of time-series, particularly fractal and certain multifractal time series [44]. This framework, coupled with a method interpreting time series as byte counts, can generate and transmit synthetic fractal and multifractal traffic at high rates with very low memory requirements.

## 2.3 Traffic Prediction

One of the key issues in measurement-based AAPN control is the prediction of the bandwidth requirements in the next control time interval based on the observed traffic characteristics. Therefore, accurate traffic prediction can be very advantageous. The goal is to forecast future variations as precisely as possible, based on the measured traffic history.

### 2.3.1 Telephony Network Traffic Prediction

Traffic prediction has a long history in telephony. A rich variety of traffic models has been employed, including autoregressive models, ARIMA models and econometric models [45]. In order to more accurately predict traffic demand, the terminals can be divided into different classes, e.g., residential, business, coin. Then each class of traffic is predicted individually and combined to generate the total traffic prediction. When the traffic measurements are available, numerous forecasting methods exist for predicting the future traffic matrix, e.g., Kruithof's method, the extension of Kruithof's method, weighted least squares and the Kalman predictor [46].

Kruithof's method forecasts the traffic matrix using a linear predictor, based on the last known traffic matrix and forecasts of the row and column sum, and then decomposes the sum of the row and column according to the gravity model. The method only uses the last known traffic matrix, and does not consider the information embedded in earlier traffic matrices. When the growth of the point-to-point traffic varies substantially, this is disadvantageous.

### 2.3.2 Data Network Traffic Prediction

In this thesis, we focus on traffic prediction for data networks. Compared with the relatively stable growth of telephone traffic, Internet traffic growth has been explosive. Even after the downturn in the telecom sector in 2000, Internet traffic continued to grow vigorously, approximately doubling each year [47]. Furthermore, Internet traffic is extremely heterogeneous [48], caused by the Internet's technical and administrative diversity, sustained growth, and immense variations over time in which applications are used and in what fashion. The heterogeneity is one factor that makes Internet traffic prediction immensely challenging.

The problem of traffic prediction can be addressed at two levels: predicting source traffic, originating from a single host, or predicting aggregate traffic, combined from many subscriber terminals.

#### Source Traffic Prediction

When predicting traffic at the source level, it is difficult to track the fine-grained behaviour of individual services or applications. It is more effective to classify traffic into a small number of basic categories each with sizable contribution to the aggregate traffic demand. In general, the four most important traffic types are (i) non-real-time large file transfer; (ii) real-time interaction such as video conference and remote manipulation; (iii) streaming services; (iv) transactions with very short messages. It is assumed that they constitute the traffic types supported in edge nodes of the future AAPN.

Measurements in 2003 show that TCP is the dominant transfer mode, supporting *http*, *ftp* (web and file transfer), as well as other common applications [49]. Fraleigh *et al.* measured the traffic in the Sprint IP backbone. In all traffic traces they analyzed, above 90% of the traffic is TCP.

#### Aggregate Traffic Prediction

Predicting traffic at the source level is not the focus of our work because the so-called edge nodes in AAPN are not data sources but are routers or switches within the Internet, hence predicting aggregate traffic is more important. Aggregate traffic prediction is more reliable and flexible because traffic aggregation offers some protection from the variability induced by the traffic generated by individual services and applications.

Define $Y(t)$ as the random process that gives the number of bytes arriving at a node in the interval $[0, t)$. The prediction goal is to estimate $Y(t+\tau)$ from the measured traffic history $Y(r)|r \in (-\infty, t]$, where $\tau$ is the prediction interval. The result $\hat{Y}(t+\tau)$ is called the $\tau$-step predictor and is used for control purposes.

Assuming a confident prediction requires that the normalized $\tau$-step prediction error $\overline{err}(\tau) \equiv \frac{\hat{Y}(t+\tau-Y(t+\tau)}{\hat{Y}(t+\tau)}$ should not exceed a percentage $\varepsilon$ with a probability $P_\varepsilon$, then the optimal prediction performance can be represented by the Maximum Prediction Interval (MPI) as

$$MPI \equiv \max\{\tau | P_{\overline{err}}(\tau, \varepsilon) \leq P_\varepsilon\}, \tag{2.17}$$

where $P_{\overline{err}}(\tau, \varepsilon) = Pr[\overline{err}(\tau) > \varepsilon]$. If the MPI provides enough time for measurement-based network control and prediction behaviours, then the arrival traffic is deemed predictable.

One method for performing the prediction step is to use a linear predictor [50]. Let $Z[n+1] = Y((n+1)\tau) - Y(n\tau) = Y(t+\tau) - Y(t), n = 0, 1, ...N-1$ with $t = n\tau$. The linear predictor that minimizes the variance of the prediction error $Z[n+1] - \hat{Z}[n+1]$ is given by

$$\begin{aligned} \hat{Z}^h[n+1] &= \hat{Y}^h(t+\tau) - Y(t), \\ &= \alpha_1 Z[n] + \alpha_2 Z[n-1] + ... + \alpha_h Z[n-h+1] \end{aligned} \tag{2.18}$$

where $h$ is the predictor length and $\bar{\alpha}$ is a column vector with values $\alpha_i (1 \leq i \leq h)$ which are the mean squared linear regression coefficients.

An alternative method is to model $Y(t)$ as a continuous-time stochastic process $Y(t) = X(t) + \mu$, where $\mu$ is the mean rate, and $X(t)$ is a purely random process with continuous integrated spectrum and zero mean [51].

### 2.3.3 Prediction Techniques

This section gives a review of prediction techniques, which can be used to estimate the unobservable system states from a vector of measurements. In order to make them applicable to traffic prediction, specific traffic models can be developed to describe the relationship between the unobservable system states and the vector of measurement.

### Kalman Filter

This section provides a brief review of the Kalman filter, following the development in [52]. The Kalman filter is a linear, discrete-time, finite-dimensional system with a recursive structure [53]. It is the minimum mean-square estimator of the state of a linear dynamical system.

The Kalman filter addresses the general problem of trying to estimate the state $\mathbf{x}(n)$ of a discrete-time controlled process with measurement. The state $\mathbf{x}(n)$, assumed to be of dimension $M$, is unobservable. To estimate it, we use a set of observed data, denoted by the vector $\mathbf{y}(n)$, which is assumed to be of dimension $N$. Then the process equation is

$$\mathbf{x}(n+1) = \mathbf{F}(n+1, n)\mathbf{x}(n) + \mathbf{v}_1(n) \tag{2.19}$$

where $\mathbf{F}(n+1, n)$ is assumed to be known and deterministic. In this equation, $\mathbf{v}_1$ represents process noise with dimension $M$, modelled as a zero-mean, white noise process with correlation matrix defined by

$$E[\mathbf{v}_1(n)\mathbf{v}_1^H(k)] = \begin{cases} \mathbf{Q}_1(n), & n = k \\ \mathbf{0}, & n \neq k \end{cases} \tag{2.20}$$

where $\mathbf{Q}_1(n)$ is assumed to be known and deterministic, and $\mathbf{Q}_1(n) \geq 0$.

The measurement equation, which describes the relation between the state process $\mathbf{x}(n)$ and the observation vector $\mathbf{y}(n)$, is

$$\mathbf{y}(n) = \mathbf{C}(n)\mathbf{x}(n) + \mathbf{v}_2(n), \tag{2.21}$$

where $\mathbf{C}(n)$ is a known $N$-by-$M$ measurement matrix. The $N$-by-1 vector $\mathbf{v}_2(n)$ represents measurement noise, modelled as a zero-mean, white-noise process with correlation matrix as

$$E[\mathbf{v}_2(n)\mathbf{v}_2^H(k)] = \begin{cases} \mathbf{Q}_2(n), & n = k \\ \mathbf{0}, & n \neq k \end{cases} \tag{2.22}$$

where $\mathbf{Q}_2(n)$ is assumed to be known and deterministic, and $\mathbf{Q}_2(n) \geq 0$.

The initial state $\mathbf{x}(0)$ is assumed to have zero mean, covariance matrix $\prod_0$, and to be uncorrelated with both $\mathbf{v}_1(n)$ and $\mathbf{v}_2(n)$ for $n \geq 0$. The noise vectors $\mathbf{v}_1(n)$ and $\mathbf{v}_2(n)$ are statistically independent $E[\mathbf{v}_1(n)\mathbf{v}_2^H(k)] = \mathbf{0}$ for all $n$ and $k$.

Define the innovation process as

$$\mathbf{a}(n) = \mathbf{y}(n) - \hat{\mathbf{y}}(n|\mathbf{y}_{n-1}), n = 1, 2, \ldots \tag{2.23}$$

where $\hat{\mathbf{y}}(n|\mathbf{y}_{n-1})$ is the orthogonal projection of $\mathbf{y}(n)$ onto $\mathbf{y}_{n-1}$.

The Kalman filter equation is

$$\hat{\mathbf{x}}(n+1|\mathbf{y}_n) = \mathbf{F}(n+1, n)\hat{\mathbf{x}}(n|\mathbf{y}_{n-1}) + \mathbf{G}(n)\mathbf{a}(n) \tag{2.24}$$

with the initial condition $\hat{\mathbf{x}}(1|\mathbf{y}_0) = 0$. The Kalman gain at time $n$ is defined as

$$\mathbf{G}(n) = \mathbf{F}(n+1, n)\mathbf{K}(n, n-1)\mathbf{C}^H(n)[\mathbf{C}(n)\mathbf{K}(n, n-1)\mathbf{C}^H(n) + \mathbf{Q}_2(n)]^{-1}. \tag{2.25}$$

$K(n, n-1)$ is calculated by the Riccati equation for the innovation process

$$\mathbf{K}(n) = \mathbf{K}(n, n-1) - \mathbf{F}(n, n+1)\mathbf{G}(n)\mathbf{C}(n)\mathbf{K}(n, n-1) \tag{2.26}$$

$$\mathbf{K}(n+1, n) = \mathbf{F}(n+1, n)\mathbf{K}(n)\mathbf{F}^H(n+1, n) + \mathbf{Q}_1(n) \tag{2.27}$$

with initial condition $\mathbf{K}(1, 0) = E[(\mathbf{x}(1) - E[\mathbf{x}(1)])(\mathbf{x}(1) - E[\mathbf{x}(1)])^H] = \prod_0$.

The relationship of (2.24) is the one-step ahead predictor. So the filtering technique incorporates a prediction framework.

### Extended Kalman Filter

The Kalman filter addresses the estimation of a state vector in a linear model of a dynamical system. If the model is mildly nonlinear, a linearization procedure can be applied, and the result is referred to as the extended Kalman filter. We follow the development in [52].

The nonlinear state-space model has the form

$$\mathbf{x}(n+1) = \mathbf{F}(n, \mathbf{x}(n)) + \mathbf{v}_1(n) \tag{2.28}$$

$$\mathbf{y}(n) = \mathbf{C}(n, \mathbf{x}(n)) + \mathbf{v}_2(n), \tag{2.29}$$

where, as in the linear model, $\mathbf{v}_1(n)$ and $\mathbf{v}_2(n)$ are uncorrelated zero-mean white-noise processes with correlation matrices $\mathbf{Q}_1(n)$ and $\mathbf{Q}_2(n)$. $\mathbf{F}(n, \mathbf{x}(n))$ denotes a nonlinear transition matrix function that is possibly time varying. Similarly, $\mathbf{C}(n, \mathbf{x}(n))$ denotes a nonlinear measurement matrix that may be time varying.

The basic idea of the extended Kalman filter is to linearize the state-space model at each time instant around the most recent state estimate. Once a linear model is obtained, the standard Kalman filter equations are applied.

The extended Kalman filter equation becomes

$$\hat{\mathbf{x}}(n|\mathbf{y}_n) = \hat{\mathbf{x}}(n|\mathbf{y}_{n-1}) + \mathbf{G}_f(n)\mathbf{a}(n) \tag{2.30}$$

$$\hat{\mathbf{x}}(n+1|\mathbf{y}_n) = \mathbf{F}(n, \hat{\mathbf{x}}(n|\mathbf{y}_n)) \tag{2.31}$$

with the initial condition $\hat{\mathbf{x}}(1|\mathbf{y}_0) = 0$. The extended Kalman gain at time $n$ is

$$\mathbf{G}_f(n) = \mathbf{K}(n, n-1)\mathbf{C}^H(n)[\mathbf{C}(n)\mathbf{K}(n, n-1)\mathbf{C}^H(n) + \mathbf{Q}_2(n)]^{-1} \tag{2.32}$$

The innovation process is now

$$\mathbf{a}(n) = \mathbf{y}(n) - \mathbf{C}(n, \hat{\mathbf{x}}(n|\mathbf{y}_{n-1})), \tag{2.33}$$

and the Riccati equation governing the evolution of $K$ is

$$\mathbf{K}(n) = [\mathbf{I} - \mathbf{G}_f(n)\mathbf{C}(n)]\mathbf{K}(n, n-1) \tag{2.34}$$

$$\mathbf{K}(n+1, n) = \mathbf{F}(n+1, n)\mathbf{K}(n)\mathbf{F}^H(n+1, n) + \mathbf{Q}_1(n) \tag{2.35}$$

with initial condition $\mathbf{K}(1,0) = E[(\mathbf{x}(1) - E[\mathbf{x}(1)])(\mathbf{x}(1) - E[\mathbf{x}(1)])^H] = \prod_0$.

### Sequential Monte Carlo Methods

Sequential Monte Carlo (SMC) methods are based on Monte Carlo simulation, and provide a convenient and attractive approach for computing posterior distributions. These methods have the great advantage of not being subject to any linearity or Gaussianity constraints on the model, and they also have appealing convergence properties. We now summarize the development in [20] as follows.

The unobserved signal $\{\mathbf{x}_t; t \in \mathbf{N}\}$, $\mathbf{x}_t \in \mathbf{X}$ is modelled as a Markov process of initial distribution $p(\mathbf{x}_0)$ and transition equation $p(\mathbf{x}_t|\mathbf{x}_{t-1})$. The observations $\{\mathbf{y}_t; t \in \mathbf{N}^*\}$, $\mathbf{y}_t \in \mathbf{Y}$, are assumed to be conditionally independent given the process $\{\mathbf{x}_t; t \in \mathbf{N}\}$ and of marginal distribution $p(\mathbf{y}_t|\mathbf{x}_t)$. The aim is to estimate recursively in time the posterior distribution $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$, $p(\mathbf{x}_t|\mathbf{y}_{1:t})$, and the expectations

$$I(f_t) = \mathbf{E}_{p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})}[f_t(\mathbf{x}_{0:t})] = \int f_t(\mathbf{x}_{0:t})p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})d\mathbf{x}_{0:t} \tag{2.36}$$

where $\mathbf{x}_{0:t} = \mathbf{x}_0, ..., \mathbf{x}_t$ and $\mathbf{y}_{1:t} = \mathbf{y}_1, ..., \mathbf{y}_t$. Examples of $f_t$ include the conditional mean, in which case $f_t(\mathbf{x}_{0:t}) = \mathbf{x}_{0:t}$, and the conditional covariance of $\mathbf{x}_t$ in which case $f_t(\mathbf{x}_{0:t}) = \mathbf{x}_t\mathbf{x}_t^T - \mathbf{E}_{p(\mathbf{x}_t|\mathbf{y}_{1:t})}[\mathbf{x}_t]\mathbf{E}^T_{p(\mathbf{x}_t|\mathbf{y}_{1:t})}[\mathbf{x}_t]$.

If we could sample directly from the posterior $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$ then a Monte Carlo Estimate of $I(f_t)$ would be $\hat{I}(f_t) = \frac{1}{N}\sum_{i=1}^N f_t(\mathbf{x}_{0:t}^{(i)})$. However, it is usually impossible to sample efficiently from the posterior distribution $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$ at any time $t$, since $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$, in general, is multivariate, non-standard, and only known up to a proportionality constant.

An alternative solution consists of using the importance sampling method. Define the importance sampling distribution as $\pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$, and the importance weight as $w(\mathbf{x}_{0:t}) = \frac{p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})}{\pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})}$. The expectations in (2.36) can then be expressed as

$$I(f_t) = \frac{\int f_t(\mathbf{x}_{0:t})w(\mathbf{x}_{0:t})\pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})d\mathbf{x}_{0:t}}{\int w(\mathbf{x}_{0:t})\pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})d\mathbf{x}_{0:t}} \tag{2.37}$$

If we generate $N$ i.i.d. particles $\{\mathbf{x}_{0:t}^{(i)}, i = 1, ..., N\}$ according to $\pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$, a possible Monte Carlo estimate of $I(f_t)$ is

$$\hat{I}_N(f_t) = \frac{\frac{1}{N}\sum_{i=1}^N f_t(\mathbf{x}_{0:t}^{(i)})w(\mathbf{x}_{0:t}^{(i)})}{\frac{1}{N}\sum_{j=1}^N w(\mathbf{x}_{0:t}^{(i)})} = \sum_{i=1}^N f_t(\mathbf{x}_{0:t}^{(i)})\tilde{w}_t^{(i)}, \tag{2.38}$$

where the normalized importance weights $\tilde{w}_t^{(i)}$ are

$$\tilde{w}_t^{(i)} = \frac{w(\mathbf{x}_{0:t}^{(i)})}{\sum_{j=1}^{N} w(\mathbf{x}_{0:t}^{(j)})} \tag{2.39}$$

The importance function $\pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$ must be designed so that it can be decomposed in the following manner:

$$\begin{aligned} \pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) &= \pi(\mathbf{x}_{0:t-1}|\mathbf{y}_{1:t-1})\pi(\mathbf{x}_t|\mathbf{x}_{0:t-1},\mathbf{y}_{1:t}), \\ &= \pi(\mathbf{x}_0)\prod_{k=1}^{t}\pi(\mathbf{x}_k|\mathbf{x}_{0:k-1},\mathbf{y}_{1:k}). \end{aligned} \tag{2.40}$$

The importance weights can then be evaluated recursively in time as

$$\tilde{w}_t^{(i)} \propto \tilde{w}_{t-1}^{(i)} \frac{p(\mathbf{y}_t|\mathbf{x}_t^{(i)})p(\mathbf{x}_t^{(i)}|\mathbf{x}_{t-1}^{(i)})}{\pi(\mathbf{x}_t^{(i)}|\mathbf{x}_{0:t-1}^{(i)},\mathbf{y}_{(1:t)})}. \tag{2.41}$$

An important particular case is when we adopt the prior distribution as importance distribution

$$\pi(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) = p(\mathbf{x}_{0:t}) = p(\mathbf{x}_0)\prod_{k=1}^{t}p(\mathbf{x}_k|\mathbf{x}_{k-1}). \tag{2.42}$$

Several closely related algorithms have appeared in a range of research fields, e.g., bootstrap filters, condensation, particle filters, Monte Carlo filters, interacting particle approximations and survival of the fittest. Here we only describe the bootstrap filter algorithm. Please refer to [20] for a more detailed description of other algorithms.

**Bootstrap Filter** The key idea of the bootstrap filter is to eliminate the particles having low importance weights $\tilde{w}_t^{(t)}$ and to multiply particles having high importance weights [54]. This is achieved by introducing a selection (or resampling) step at selected time instants. The algorithm consists of the following steps:

1. Initialization, $t = 0$
   - For $i = 1, ..., N$, sample $\mathbf{x}_0^{(i)} \sim p(\mathbf{x}_0)$ and set $t = 1$.
2. Importance sampling step

- For $i = 1, ..., N$, sample $\tilde{\mathbf{x}}_t^{(i)} \sim p(\mathbf{x}_t|\mathbf{x}_{t-1}^{(i)})$ and set $\tilde{\mathbf{x}}_{0:t}^{(i)} = (\mathbf{x}_{0:t-1}^{(i)}, \tilde{\mathbf{x}}_t^{(i)})$.
- For $i = 1, ..., N$, evaluate the importance weights

$$\tilde{w}_t^{(i)} = p(\mathbf{y}_t|\tilde{\mathbf{x}}_t^{(i)}) \tag{2.43}$$

- Normalize the importance weights.

3. Selection step

- Resample (with replacement) $N$ particles $(\mathbf{x}_{0:t}^{(i)}; i = 1, ..., N)$ from the set $(\tilde{x}_{0:t}^{(i)}; i = 1, ..., N)$ with probability of selection proportional to the normalized importance weight.
- Set $t \leftarrow t + 1$ and go to step 2.

# Chapter 3

# Traffic Models

## 3.1 Introduction

We propose a new traffic modelling approach for an all-photonic network architecture. Gravity models have a long history in Origin-Destination (OD) traffic modelling. They have been observed to be relatively accurate in representing the spatial distribution of the total incoming traffic [26]. The time-varying or Markov-Modulated Poisson Process (MMPP) has been used for a long time to approximate aggregate traffic arrival processes [29].

In this chapter, we combine a gravity model and a time-varying Poisson model to form a dynamic parametric traffic model for aggregate OD traffic, as shown in Fig. 3.1. The dynamic model maintains state parameters that capture the underlying rates of traffic between origin-destination pairs. Each state parameter is determined by a Gravity-model decomposition of a shared rate component for the origin node, and its previous value. The observed traffic between origin-destination pair is governed by a Poisson distribution with the rate of state parameter, with some fractional Gaussian noise added.

## 3.2 Time-Varying Poisson Process

Poisson models are the oldest traffic models, dating back to the advent of telephony and renowned pioneering telephone engineer A. K. Erlang. As explained in Chapter 2, the identification of long-range dependence discredited Poisson based models for data networks traffic. However, due to the increase of the link speed and the number of hosts, the Poisson traffic model is becoming a better model for current network traffic for sub-second time

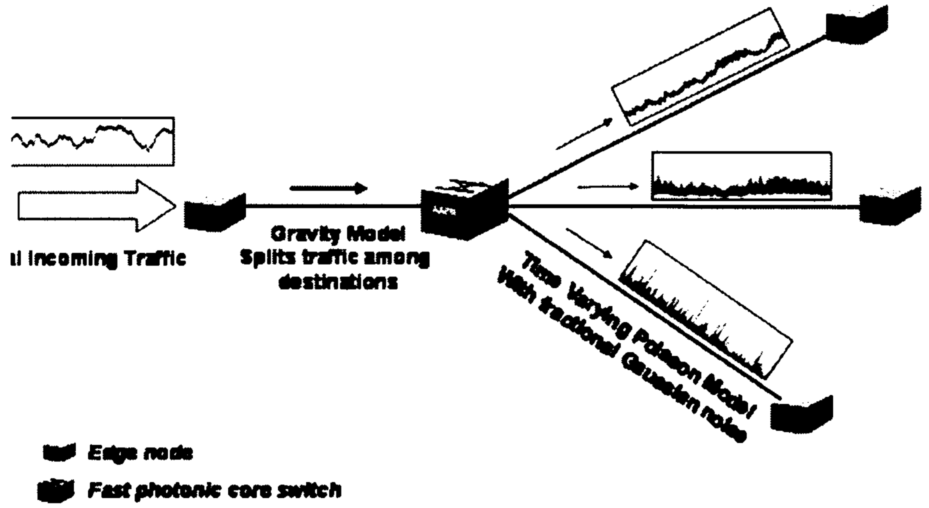

l Incoming Traffic
Gravity Model
Splits traffic among
destinations
Time Varying Poisson Model
With fractional Gaussian noise
Edge node
Fast photonic core switch

**Fig. 3.1** Graphical representation of the dynamic parametric traffic model. The dynamic parametric traffic model maintains state parameters that capture the underlying rates of traffic between origin-destination pairs. Each state parameter is determined by a Gravity-model decomposition of a shared rate component for the origin node and its previous value. The observed traffic between an origin-destination pair is governed by a Poisson distribution with the rate of state parameter.

scales [55–58].

Our proposed model focuses on short-range traffic modelling and prediction. We aim to develop a traffic model that can be used for traffic prediction at a time scale appropriated for traffic control of an AAPN, which is determined by the round-trip time between the edge and core nodes (approximately 10 – 80ms). We model the rate of the total incoming traffic to an edge node, $\lambda_i(t)$, as follows:

$$\lambda_i(t) = \max\{0, \lambda_i(t-1) + w_i(t)\}, \tag{3.1}$$

where $w_i(t) \sim N(0, \sigma_i^2)$. At time interval $t$, the total traffic to flow into an edge node $i$ with rate $\lambda_i(t)$ shall always be positive. This rate is renewed at each time interval by a normal distribution, with the mean of itself in the previous time interval $\lambda_i(t-1)$ and variance $\sigma_i^2$.

The rate $\lambda_i(t)$ in part determines the Poisson rate of the traffic between individual OD pair $\lambda_{ij}(t)$. We model the traffic between individual OD pairs as the sum of a time-varying Poisson process and fractional Gaussian noise.

## 3.3 Gravity Model

Gravity models, which drive their name from Newton's law of gravitation, are commonly used by social scientists to model the movement of people, goods or information between geographic areas [26]. In Newton's law of gravitation, the force is proportional to the product of the masses of the two objects divided by the distance squared:

$$F_g = G\frac{m_1 m_2}{r^2}, \tag{3.2}$$

where $F_g$ is the gravitational force, $m_1$ and $m_2$ are the masses of the two objects, $r$ is the separation between the two objects, and $G$ is the universal gravitational constant. Similarly, in gravity models for geographic areas, the relative strength of the interaction between two geographic areas might be modelled as proportional to the product of the populations. A general formulation of gravity model is given by the following equation:

$$X_{ij} = \frac{R_i \cdot A_j}{f_{ij}}, \tag{3.3}$$

where $X_{ij}$ is the matrix element representing the force from $i$ to $j$; $R_i$ represents the repulsive factors that are associated with "leaving" from $i$; $A_j$ represents the attractive factors that are associated with "going" to $j$; and $f_{ij}$ is a friction factor from $i$ to $j$.

We are using the gravity model to derive the decomposition factor to split the total incoming traffic to an edge node between individual destinations. Denote $m_i$ as the size of the population which generates network traffic that flows into edge node $i$. In our approach, we first modify the general formulation of gravity model as follows:

$$X_{ij} = \frac{m_i \cdot m_j}{1}, \tag{3.4}$$

where $X_{ij}$ is the *gravitational force* between two areas. The repulsive factor associated with "leaving" from edge node $i$ is proportional to the size of the population $m_i$, and the attractive factor associated with "going" to edge node $j$ is proportional to $m_j$. The more population there is in two geographic areas, the more network traffic there will likely to be exchanged. The friction factor $f_{ij}$ represents the locality information specific to different OD pairs. In today's data networks, geographic locality is generally not a major factor [26]. In all-photonic networks, it is probable that the majority of the traffic is data. So distance is no longer an important parameter in this scenario. In this thesis, we assume a common constant for the friction factors, which is set to be 1 without loss of generality.

The gravitational force $X_{ij}$ gives the relation between each OD pair $(i, j)$ individually. In order to derive the decomposition vector which is used to split up the shared rate component among individual OD pairs, we need to get the fractional relation of the attractive force of each OD pair compared to the whole destination set. Then the gravitational force is normalized as follows:

$$z_{ij} = \frac{m_i \cdot m_j}{\sum_j m_i \cdot m_j}, \tag{3.5}$$

where $z_{ij}$ is the decomposition factor between OD pair $(i, j)$ compared to all destinations.

## 3.4 Fractional Brownian Motion and Fractional Gaussian Noise

Fractional Brownian motion (fBm) with self-similarity parameter (Hurst parameter) $H$ is a stochastic process $\{Z_t, t \in (-\infty, \infty)\}$ which has the following properties [59]:

- $Z_t$ has stationary increments;
- $Z_0 = 0$, and $\mathbf{E}[Z_t] = 0$ for all $t$;
- $\mathbf{E}[Z_t^2] = |t|^{2H}$ for all $t$;
- $Z_t$ has continuous paths;
- $Z_t$ is Gaussian.

The Hurst parameter $H$ determines the characteristics of the fBm. $Z_t$ is a long-range dependent process if $H > 0.5$, and a short-range dependent process if $H < 0.5$. In the special case $H = 1/2$, $Z_t$ is the standard Brownian motion. The incremental sequence $Z_t - Z_{t-1}$ is called fractional Gaussian noise.

In [59], Norros models the aggregated connectionless traffic based on fractional Brownian motion:

$$A_t = qt + \sqrt{pq}Z_t, \qquad t \in (-\infty, \infty), \tag{3.6}$$

where $A_t$ is the aggregate traffic to time $t$, $q$ is the mean arrival rate, $p$ is the variance coefficient, and $Z_t$ is determined by a fractional Brownian motion with Hurst parameter $H$. The incremental sequence of a fractional Brownian motion is called fractional Gaussian noise.

We use the fractional Gaussian noise (fGn) to capture bursts and variations in the traffic that occur over relatively short time-scales. Let $\lambda_{ij}(t)$ be the mean rate of the traffic between OD pair $(i, j)$. The traffic between this OD pair $y_{ij}(t)$ is modulated by the fractional Gaussian noise, as follows:

$$y_{ij}(t) = x_{ij}(t) + \sqrt{A_{ij}M_{ij}}[Z_{ij}(t) - Z_{ij}(t-1)], \tag{3.7}$$

where $x_{ij}(t) \sim Poisson(\lambda_{ij}(t))$, $A_{ij}$ is the variance coefficient, and $M_{ij}$ is the mean rate of the traffic between pair $(i, j)$. $Z_{ij}(t)$ is generated by fractional Brownian motion with Hurst parameter $H_{ij}$, and $Z_{ij}(t) - Z_{ij}(t-1)$ is the fractional Gaussian noise.

## 3.5 Combined Model

Combining the time-varying Poisson model and gravity model, we can get our proposed traffic model by which the edge node can estimate the traffic matrix element for all destinations. As shown in Fig. 3.2, the dynamic parametric traffic model maintains state parameters $\lambda_{ij}(t)$ that capture the underlying rates (the traffic fluctuates around the underlying rate) of traffic between OD pairs $(i, j)$ during time-slot $t$. Each state parameter is updated through the weighted summation of a Gravity-model decomposition of a shared rate component $\lambda_i(t)$, its previous value $\lambda_{ij}(t-1)$, and Gaussian innovation noise $v_{ij}(t)$. The observed traffic $y_{ij}(t)$ is then the sum of a draw from a Poisson distribution with rate parameter $\lambda_{ij}(t)$ and fractional Gaussian noise. The total traffic to an edge node $i$ is the

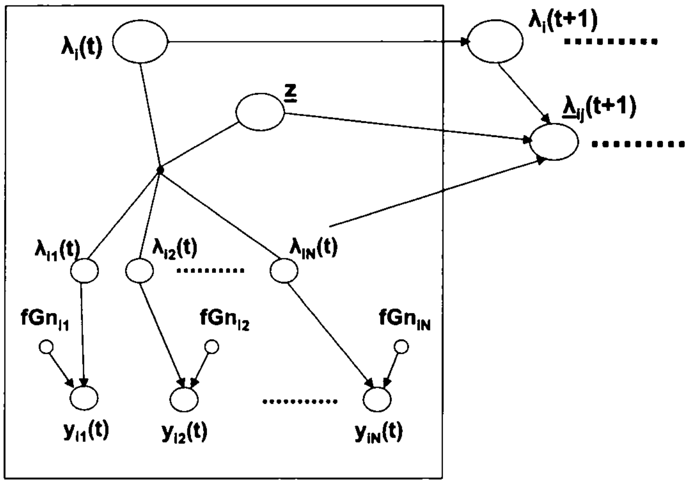


**Fig. 3.2** The key elements of the proposed dynamic parametric traffic model. The dynamic parametric traffic model maintains state parameters $\lambda_{ij}(t)$ that capture the underlying rates of traffic between OD pairs $(i,j)$ during time-slot $t$. Each state parameter is updated through the weighted summation of a Gravity-model decomposition of a shared rate component $\lambda_i(t)$, its previous value $\lambda_{ij}(t-1)$, and Gaussian innovation noise $v_{ij}(t)$. The observed traffic $y_{ij}(t)$ is then the sum of a draw from a Poisson distribution with rate parameter $\lambda_{ij}(t)$ and fractional Gaussian noise (fGn). The total traffic to an edge node $i$ is the sum of the observed $y_{ij}(t)$.

sum of the observed $y_{ij}(t)$. The traffic model is described by the following equation set:

$$\begin{array}{rcl}
\lambda_i(t) & = & \max\{0, \lambda_i(t-1) + w_i(t)\} \\
z_{ij} & = & \dfrac{m_i \cdot m_j}{\sum_j m_i \cdot m_j} \\
\lambda_{ij}(t) & = & \max\{0, a z_{ij}\lambda_i(t) + (1-a)\lambda_{ij}(t-1) + v_{ij}(t)\} \\
x_{ij}(t) & \sim & Poisson(\lambda_{ij}(t)) \\
y_{ij}(t) & = & x_{ij}(t) + \sqrt{A_{ij}M_{ij}}[Z_{ij}(t) - Z_{ij}(t-1)] \\
y_i(t) & = & \sum\limits_j y_{ij}(t)
\end{array} \qquad (3.8)$$

where the initial conditions are $\lambda_i(0) = \lambda_i$, $\lambda_{ij}(0) = z_{ij}\lambda_i(0)$ and $Z_{ij}(0) = 0$. $\lambda_i(t)$ is the rate of all the traffic flowing into edge node $i$ during time-slot $t$, and $w_i(t) \sim N(0, \sigma_1^2)$. $z_{ij}$ is the decomposition factor for the OD pair $(i, j)$, which is derived from a gravity model where $m_i$ is the size of the population in area $i$, as explained in Section. 3.3. $\lambda_{ij}(t)$ is the state parameter representing the rate of the traffic between OD pair $(i, j)$, and $v_{ij}(t) \sim N(0, \gamma_{ij}^2)$. $\lambda_i(t)$ is a determining factor of $\lambda_{ij}(t)$, and parameter $a \in [0, 1]$ defines the influence of $\lambda_i(t)$ and $\lambda_{ij}(t-1)$ on the value of $\lambda_{ij}(t)$. $y_{ij}(t)$ is the traffic between individual OD pair $(i, j)$ in time interval $t$, and it is determined by modulating the incremental sequence of fractional Brownian motion $Z_{ij}(t)$, with Hurst parameter $H_{ij}$, onto the Poisson process with rate $\lambda_{ij}(t)$. $A_{ij}$ specifies the variance coefficient for OD pair, and $M_{ij}$ is the mean of the traffic between individual OD pair $(i, j)$. The total incoming traffic to an edge node $i$ is the sum of the traffic from the same origin node to all destinations.

## 3.6 Statistical Analysis

The proposed traffic model is intended to capture the network traffic behaviour so that the expected increments of the traffic elements shall be zero. Suppose that the parameters at time slot $t-1$ are $\lambda_i(t-1)$, $\lambda_{ij}(t-1)$, $y_{ij}(t-1)$, $y_i(t-1)$. We assume that $P\{\lambda_i(t-1)+w_i(t) > 0\} \simeq 1$, which will be the case if $\lambda_i(t-1) \gg \sigma^2$. The expectation of the increment of $\lambda_i(t)$ is

$$\begin{aligned} \mathbf{E}[\lambda_i(t) - \lambda_i(t-1)] &= \mathbf{E}[max\{0, \lambda_i(t-1) + w_i(t)\} - \lambda_i(t-1)] \\ &\simeq \mathbf{E}[\lambda_i(t-1) + w_i(t) - \lambda_i(t-1)] \\ &= \mathbf{E}[w_i(t)] \\ &= 0 \end{aligned} \tag{3.9}$$

since $w_i(t) \sim N(0, \sigma_1^2)$. We also assume $P\{az_{ij}\lambda_i(t)+(1-a)\lambda_{ij}(t-1)+v_{ij}(t) > 0\} \simeq 1$, which will be the case if $a > 0$, $0 < z_{ij} < 1$, $\lambda_i(t) > 0$ and $az_{ij}\lambda_i(t) + (1-a)\lambda_{ij}(t-1) \gg \gamma_{ij}^2$. From the initial condition $\lambda_{ij}(0) = z_{ij}\lambda_i(0)$, we can get $\mathbf{E}[\lambda_{ij}(t)] = \mathbf{E}[z_{ij}\lambda_i(t)]$. The expectation of the increment of $\lambda_{ij}(t)$ is

$$\begin{aligned} & \mathbf{E}[\lambda_{ij}(t) - \lambda_{ij}(t-1)] \\ = \; & \mathbf{E}[max\{0, az_{ij}\lambda_i(t) + (1-a)\lambda_{ij}(t-1) + v_{ij}(t)\} - \lambda_{ij}(t-1)] \\ \simeq \; & \mathbf{E}[az_{ij}\lambda_i(t) + (1-a)\lambda_{ij}(t-1) + v_{ij}(t) - \lambda_{ij}(t-1)] \\ = \; & \mathbf{E}[az_{ij}\lambda_i(t) - a\lambda_{ij}(t-1)] \\ = \; & \mathbf{E}[az_{ij}\lambda_i(t-1) - a\lambda_{ij}(t-1)] \\ = \; & 0 \end{aligned} \tag{3.10}$$

since $v_{ij}(t) \sim N(0, \sigma_2^2)$ is independent of $\lambda_{ij}(t-1)$ and $\lambda_i(t-1)$. The expectation of the increment of $x_{ij}(t)$ is

$$\begin{aligned} & \mathbf{E}[\mathbf{E}[x_{ij}(t) - x_{ij}(t-1) | \lambda_{ij}(t), \lambda_{ij}(t-1)]] \\ = \; & \mathbf{E}[\lambda_{ij}(t) - \lambda_{ij}(t-1)] \\ = \; & 0 \end{aligned} \tag{3.11}$$

as $x_{ij}(t) \sim Poisson(\lambda_{ij}(t))$, and the expectation of $\lambda_{ij}(t) - \lambda_{ij}(t-1)$ is zero from (3.10). The expectation of the increment of $y_{ij}(t)$ is

$$\begin{aligned} & \mathbf{E}[y_{ij}(t) - y_{ij}(t-1)] \\ = \; & \mathbf{E}[x_{ij}(t) + \sqrt{A_{ij}M_{ij}}(Z_{ij}(t) - Z_{ij}(t-1)) - x_{ij}(t-1) \\ & + \sqrt{A_{ij}M_{ij}}(Z_{ij}(t-1) - Z_{ij}(t-2))] \\ = \; & \mathbf{E}[x_{ij}(t) - x_{ij}(t-1) + \sqrt{A_{ij}M_{ij}}(Z_{ij}(t) - 2Z_{ij}(t-1) + Z_{ij}(t-2)) \\ = \; & 0 \end{aligned} \tag{3.12}$$

since $\mathbf{E}[Z_t] = 0$ for all $t$, and (3.11). The expectation of the increment of the total incoming traffic to an edge node $y_i(t)$ is

$$\begin{aligned} & \mathbf{E}[y_i(t) - y_i(t-1)] \\ = & \mathbf{E}[\sum_j y_{ij}(t) - \sum_j y_{ij}(t-1)] \\ = & \sum_j \mathbf{E}[y_{ij}(t) - y_{ij}(t-1)] \\ = & 0 \end{aligned} \tag{3.13}$$

## 3.7 Examples of Synthesized Traffic

**Network Setup** Consider a network consisting of $N = 9$ edge nodes, with an arbitrary network topology. Suppose the population in each of the areas at the locations of edge nodes 1-3 ($m_1 - m_3$) is $1,000,000$, edge nodes 4-6 ($m_4 - m_6$) is $2,000,000$, and edge nodes 7-9 ($m_7 - m_9$) is $4,000,000$. As discussed in Section 3.2, the time scale of the proposed model is set to 10ms. The initial traffic, $y_{ij}(0)$, between origin node $i$ and destination node $j$ is set to $10^{-8} \times m_i \times m_j$ bytes per time unit. For instance, the traffic between origin node and destination node with population $2,000,000$ and $1,000,000$ respectively is $20,000$ bytes per time unit, i.e. 2Mbytes/s.

**Key Parameters** The rate of the total traffic flowing into origin node $i$, $\lambda_i(t)$, is modelled as a standard Brownian motion as in (3.1), and the parameter affecting the behaviour of $\lambda_i(t)$ is the variance of $w_i(t)$, i.e. $\sigma_i^2$. In all-photonic networks, the traffic from a large number of hosts is aggregated to an edge node. The variation of the traffic is expected to be relatively small compared to the mean of the total traffic to an edge node. Therefore, the variance of $w_i(t)$ is set to be $\sigma_i^2 = 10^4$. Fig. 3.3 shows the behaviour of $\lambda_i(t)$.

The total incoming traffic is decomposed between individual destinations as (3.8). $z_{ij}$ is a constant for a specific OD pair and helps draw the traffic fraction back to the constant which is proportional to the population. The parameters affecting the behaviour of $\lambda_{ij}(t)$ are the variance of $v_{ij}(t)$ (i.e. $\sigma_{ij}^2$), and the weight coefficient $a$. Fig. 3.4(a) shows the traffic flowing into origin node 1 destined to an individual destination $y_{1j}(t)$ with corresponding $\gamma_{1j}^2$, and $a = 0.8$. The first panel in Fig. 3.4(a) and 3.4(b) show the rate of the total

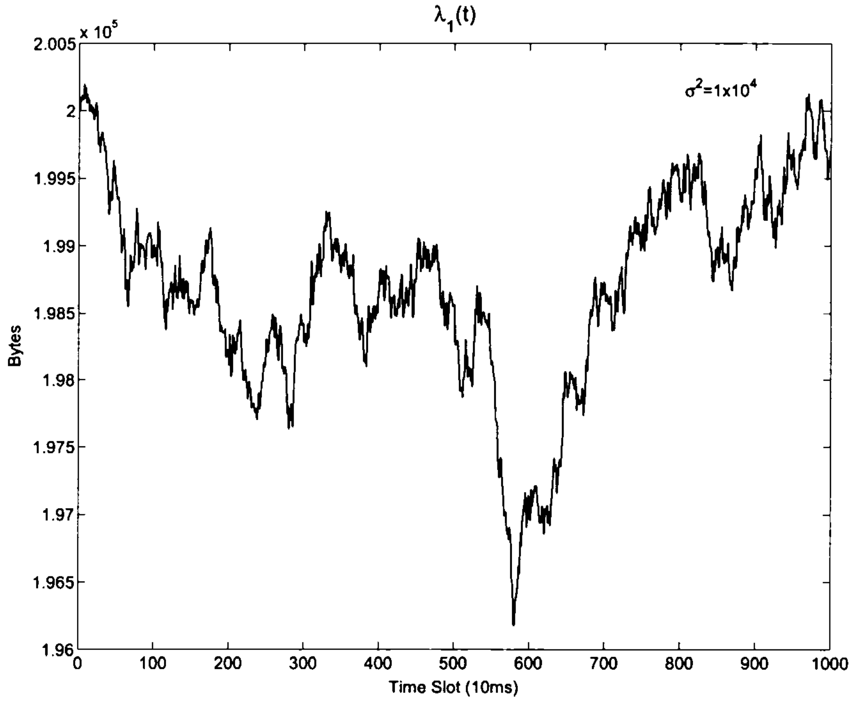
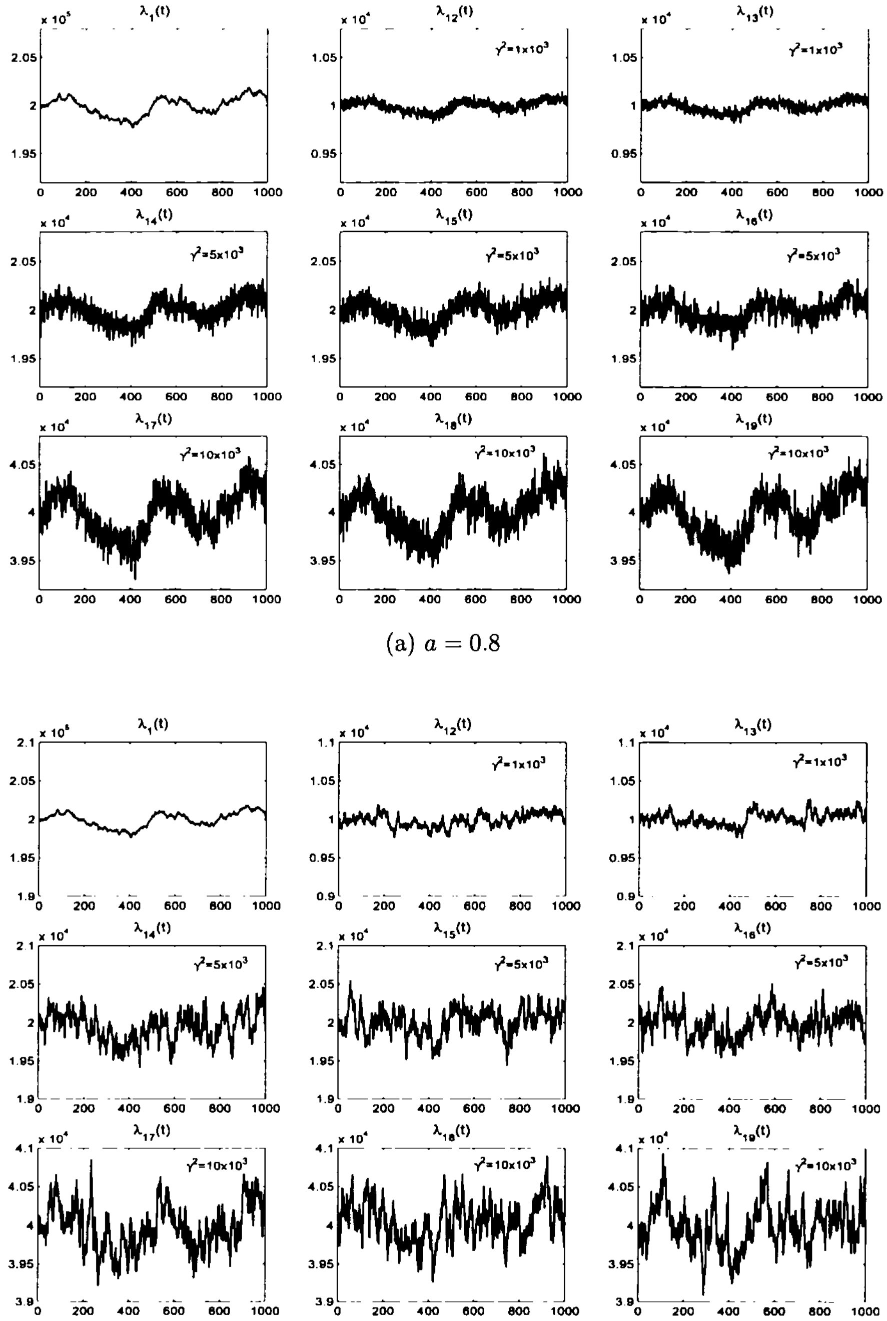


**Fig. 3.3** The mean rate of the total traffic flowing into origin node 1, $\sigma_1^2 = 1 \times 10^4$.

incoming traffic to edge node 1, $\lambda_1(t)$. Due to the greater weight on the total incoming traffic to the origin edge node, the traffic mean for individual OD pair varies in a similar way to the $\lambda_1(t)$. In Fig. 3.4(b), the variations for each OD pair is still the same as Fig. 3.4(a), but $a$ is reduced to be 0.1. The reduction of $a$ makes $\lambda_{ij}(t)$ determined more by itself in the previous time slot than the total mean to the origin edge node. The smaller $a$ induces more variation in $\lambda_{ij}(t)$.

In this model, the $\lambda$-component is used to capture the long-time scale variations and the fractional Gaussian noise is intended to capture bursts and variations in the traffic that occur over relatively short time-scales. Fig. 3.5 presents two realizations of fractional Gaussian noise with Hurst parameters. It shows that the fractional Gaussian noise with $H = 0.3$ presents more variations in short time-scales. In the case of $H = 0.8$, the fractional Gaussian noise fluctuates relatively slowly. Its value is likely to be positive when it is positive in the previous time unit. In the case of $H = 0.3$, the value of the fractional Gaussian noise at time slot $t$ does not have as much effect on the value at $t + 1$. Fig. 3.6



(a) $a = 0.8$

(b) $a = 0.1$

**Fig. 3.4** The effect of the parameter $a$ on the synthesized traffic, $\sigma_1^2 = 10^4$.

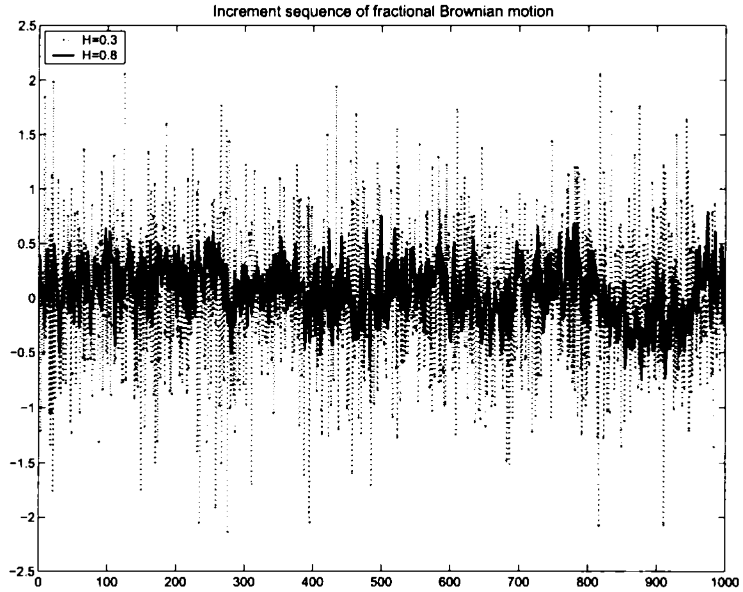


**Fig. 3.5** Examples of the incremental sequence of fractional Brownian motion, with Hurst parameter $H = 0.3$ and $H = 0.8$.

shows the influence of the Hurst parameter, as the bigger Hurst parameter introduces more variation into the traffic.

As described in Eqn. 3.7, besides the Hurst parameter $H_{ij}$, the value of $A_{ij}$ is another way to change the peakedness of $y_{ij}(t)$ in short time-scales. Fig. 3.7 shows the effect of varying the parameters $A_{ij}$. The bigger $A_{ij}$, the more variation of the traffic. However, the nature of the variation introduced by varying the parameters $A_{ij}$ is different with that of the parameters $H_{ij}$. In the case of bigger $H_{ij}$ as shown in Fig. 3.6, the traffic fluctuates more because the fractional Gaussian noise is more dependent on its previous value. The parameter $A_{ij}$ is a variance coefficient, which specifies the variation of the noise at each time slot independently, as shown in Fig. 3.7.

Fig. 3.8 shows the traffic between individual OD pair which is generated by varying the parameters $H_{ij}$,' $A_{ij}$, and $\gamma_{ij}^2$. We can use this model to generate both bursty (large variations) traffic (e.g. OD pair $(1,9)$) and non-bursty (small variations) traffic (e.g. OD pair $(1,2)$). The resultant total traffic to the origin node $y_i(t)$ is the sum of the traffic $y_{ij}(t)$. The total traffic has the similar behaviour to the shared rate component $\lambda_i(t)$ but includes some noises introduced from each OD pair.

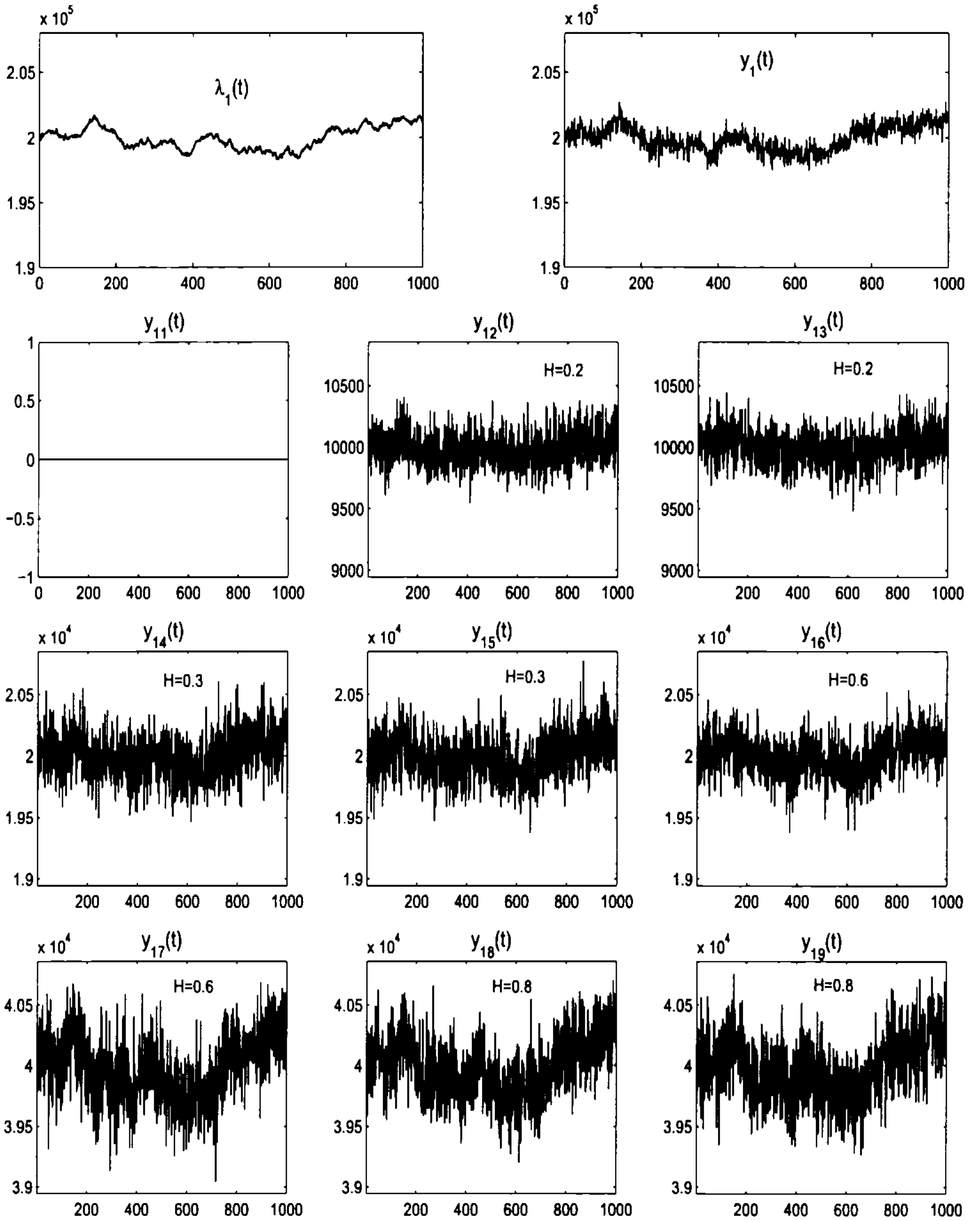


**Fig. 3.6** The effect of varying the Hurst parameters $H_{ij}$. Traffic originating from edge node 1 destined to individual nodes with $a$=0.8, $\sigma_1^2 = 10^4$, $\gamma_{1j}^2 = 10^3$. The traffic with bigger Hurst parameters fluctuates more.

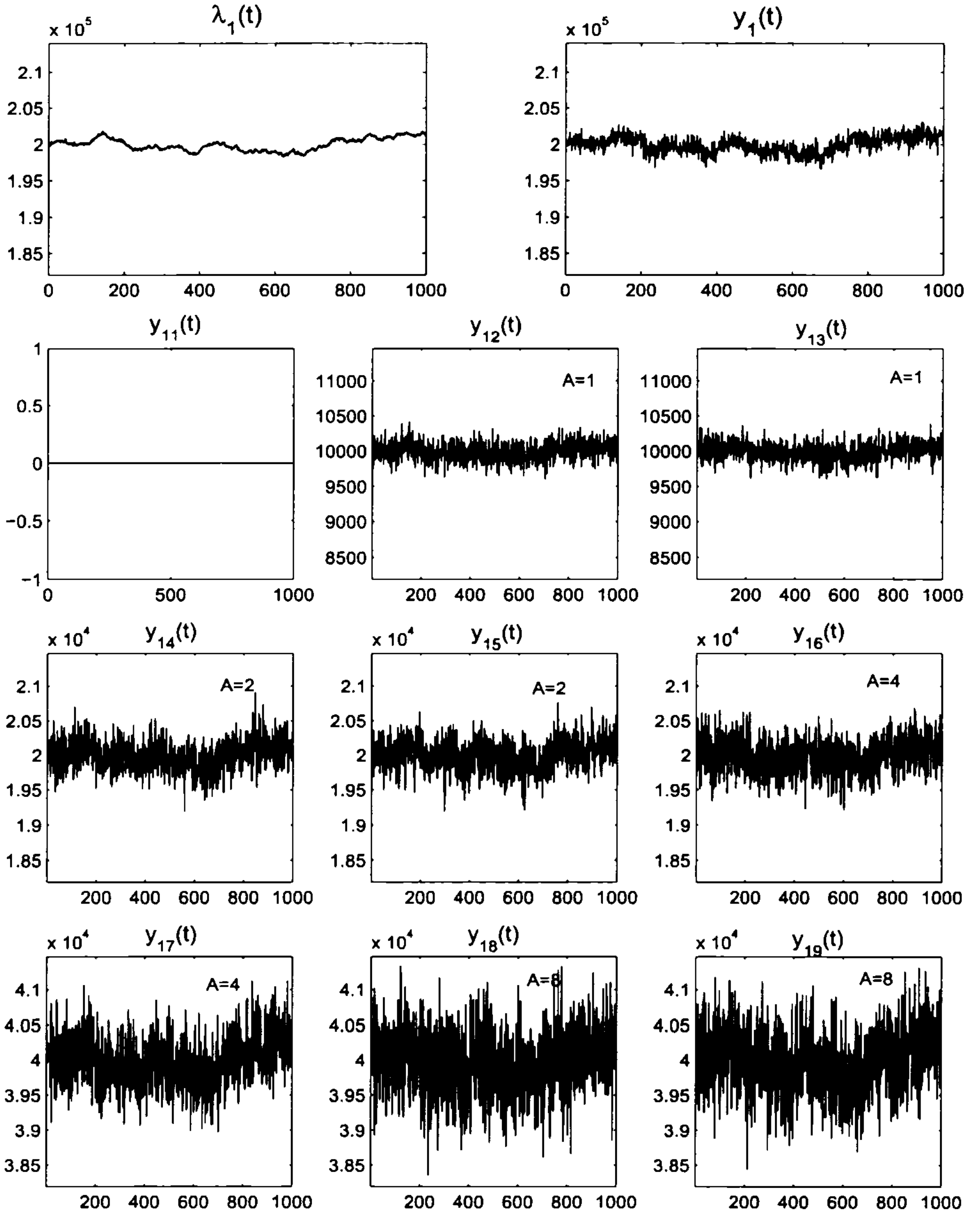


**Fig. 3.7** The effect of varying the parameters $A_{ij}$. Traffic originated from edge node 1 destined to individual nodes with $a = 0.8$, $\sigma_1^2 = 10^4$, $\gamma_{1j}^2 = 10^3$ and $H_{ij} = 0.3$. The parameter $A_{ij}$ is a variance coefficient, which specifies the variation of the noise at each time slot independently.

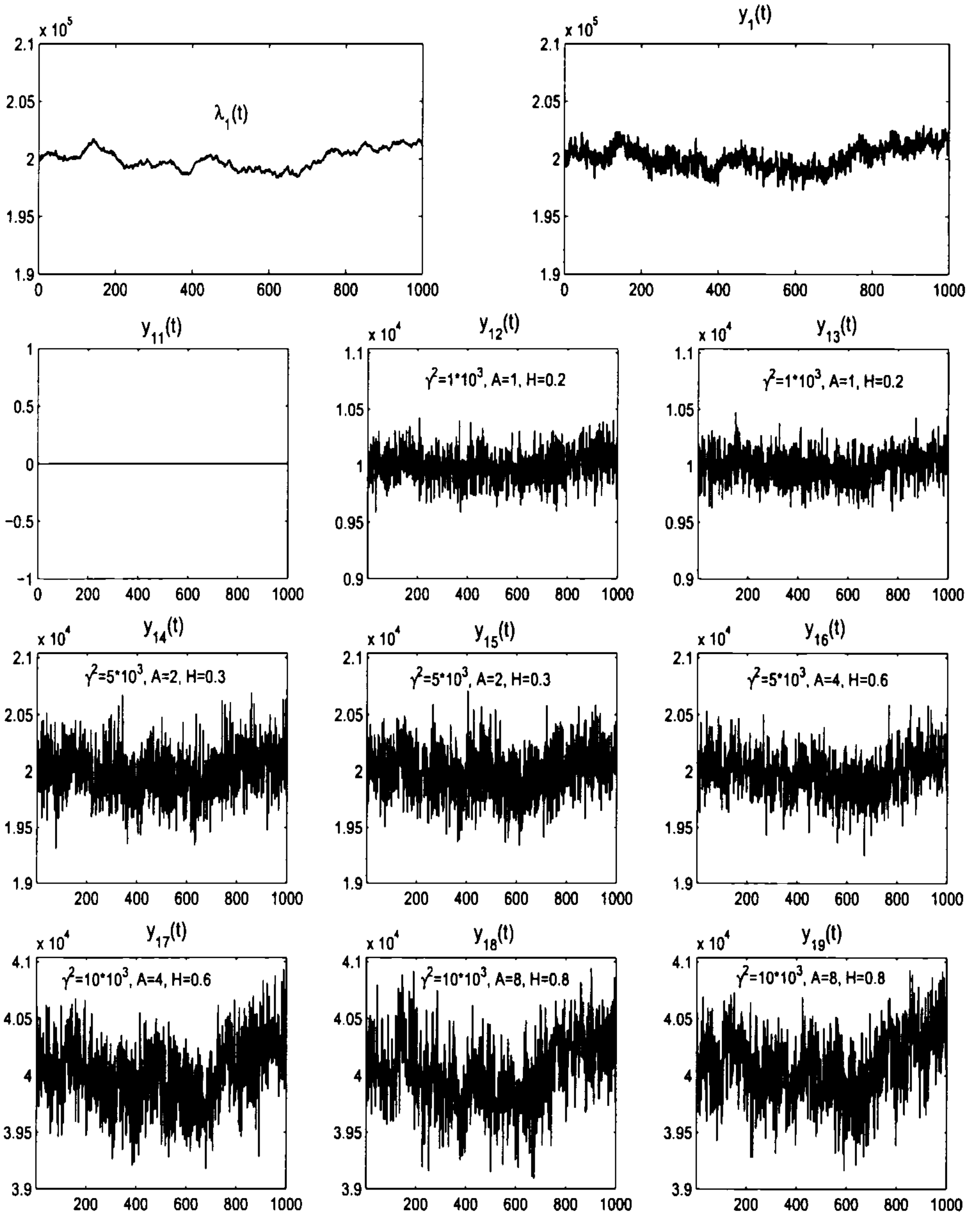


**Fig. 3.8** The effect of variation combination of all parameters. Traffic originating from edge node 1 destined to individual nodes with $a$=0.8, $\sigma_1^2 = 10^4$.

# Chapter 4

# Data Analysis

In this chapter, we analyze examples of empirical and synthesized traffic traces. We evaluate the Normalized Correlation Function (NCF) to measure the similarity between the time series quantized from the empirical and synthesized traffic traces. We also examine the long-range dependence and test the time constancy of scaling parameters of the traffic traces, using the tools developed by Abry and Veitch [4]. The tools generate an analysis framework for time series, including Long-Range Dependent (LRD) and Self-Similar (SS). They allow the estimation of key parameters based on the coefficients of a discrete wavelet decomposition.

## 4.1 Empirical Traffic Traces

The two empirical traffic traces we analyzed were recorded at Columbia University and are provided by Passive Measurement and Analysis (PMA) of NLANR [1]. The first trace, which we label *Columbia1*, includes the records between 12:28:37 and 12:30:07 on March 2, 2004. The second one, which we label *Columbia2*, was recorded between 21:35:51 and 21:37:21 on June 15, 2004. The two traces consist of 1,508,779 and 1,408,137 records, respectively, logged in the period of about 90 seconds, which include all the packets that flow into this monitored router. The measured link capacity of this PMA site is OC3c, and the measured throughputs are $97.5 Mbit/s$ and $91.3 Mbit/s$, respectively. Fig. 4.1(a) and Fig. 4.1(b) show the total traffic passing through the monitored link during the two recording sessions. In

[1] Data traces are available at http://pma.nlanr.net//

the data traces provided by the NLANR project, IP addresses are encoded out of concern for the privacy and security of those who use the network. This encoding does not impact any statistical analysis. In the Columbia1 data trace, there are in total 8,464 source IP addresses and 9,848 destination IP addresses, and Columbia2 data trace consists in total 5,041 source IP addresses and 5,061 destination addresses.

## 4.2 Normalized Correlation Function Analysis

We examine the correlation behaviour of the traces by generating the Normalized Correlation Function (NCF), $\rho(k)$, which measures the similarity between a series $y(t)$, and a shifted version of itself, $y(t+k)$:

$$\rho(k) = \frac{E[y(t)y(t+k)]}{\sigma^2}, \tag{4.1}$$

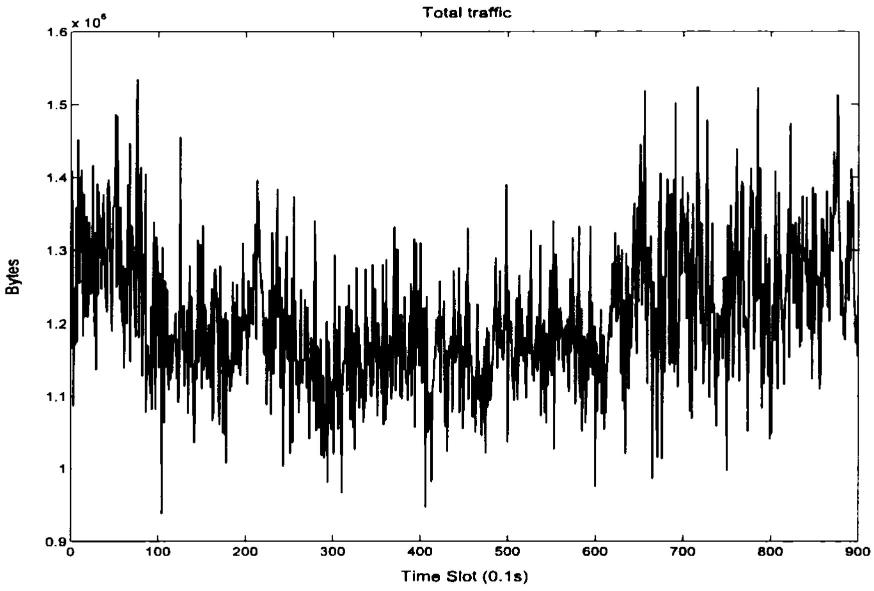
where $\sigma^2$ is the standard deviation.

The Columbia data traces contain the records for all arriving packets with time stamp and packet size. We quantized the data traces to time series based on the time stamps, with the time interval 1ms. We assume the quantized time series are stationary processes (an assumption which will be tested in Section 4.5). The autocorrelation function is generated by

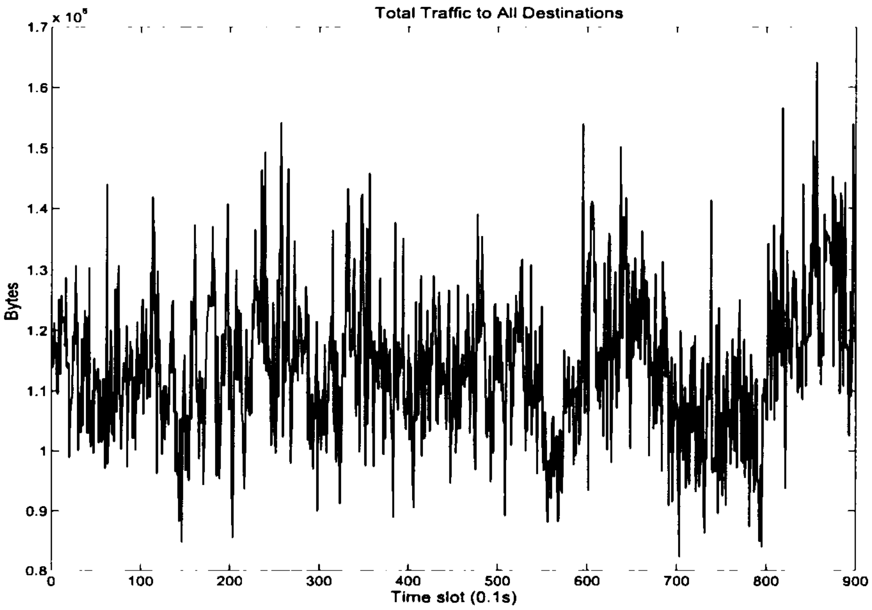
$$\hat{R}_{yy}(k) = \begin{cases} \sum_{n=0}^{N-k-1} y(n+m)y(n) & m \geq 0, \\ \hat{R}_{yy}(-m) & m < 0, \end{cases} \tag{4.2}$$

where $k = 1, ..., N-1$. It is then normalized so that the autocorrelation at zero lag is 1. Fig. 4.2(a) and Fig. 4.2(b) show the NCF calculated for Columbia data traces, which both decay to zero as lag increases.

Fig. 4.3(a) and Fig. 4.3(b) present the NCF of an example of the synthesized traffic generated by the proposed model. In Fig. 4.3(a), the parameters are selected as $\sigma_i^2 = 10^4$, $\gamma_{ij}^2 = 10^3$, $a = 0.8$, $A_{ij} = 1$ and $H_{ij} = 0.3$, which are intended to generate slowly varying traffic. The parameters for Fig. 4.3(b) are $\sigma_i^2 = 10^4$, $\gamma_{ij}^2 = 10^3$, $a = 0.8$, $A_{ij} = 1$ and $H_{ij} = 0.8$, which generate traffic that varies faster than in Fig.4.3(a). This was analyzed in Chapter 3.7. The Normalized Correlation functions behave similarly in these two synthesized traffic traces, and they both decay to zero as lag increases.

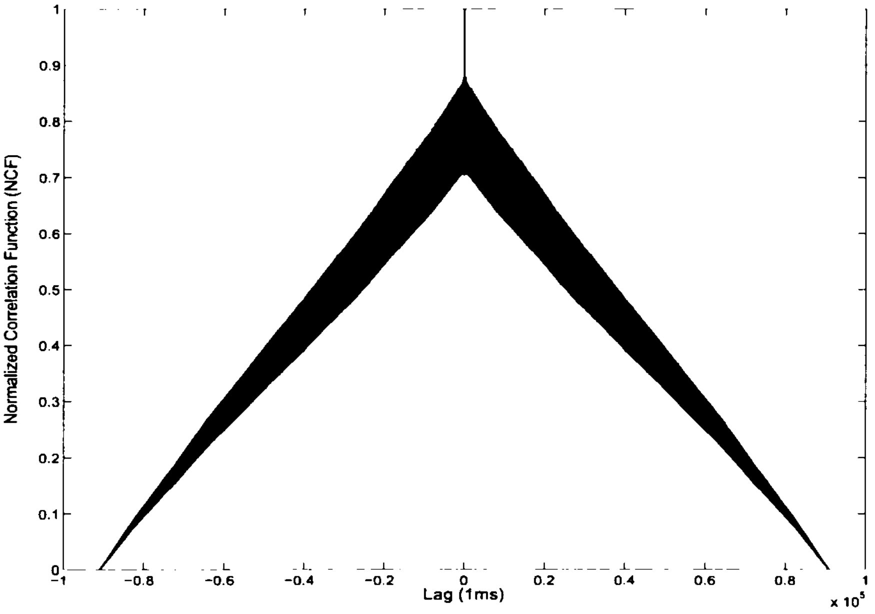


(a) Total incoming traffic measured between 12:28:37 and 12:30:07 on March 2, 2004 (Columbial).

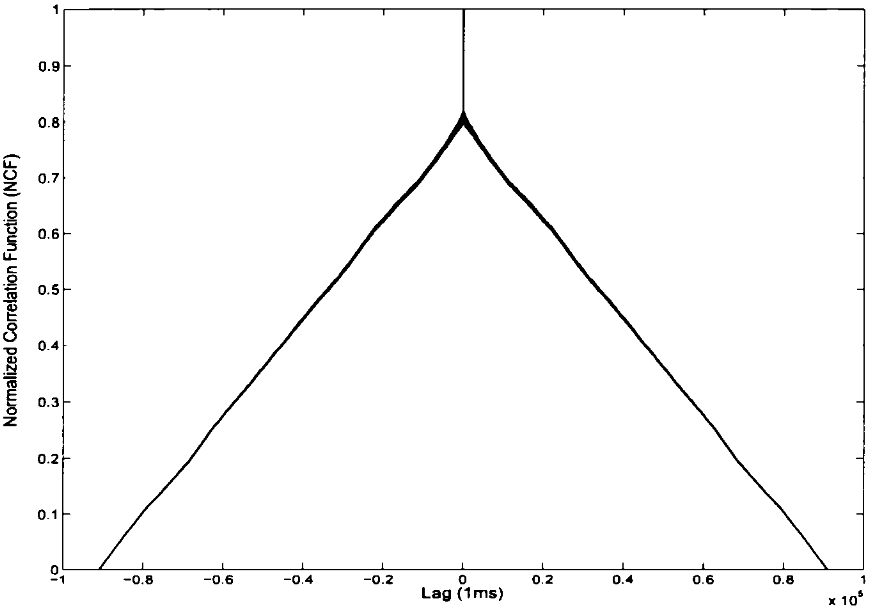


(b) Total incoming traffic measured between 21:35:51 and 21:37:21 on June 15, 2004 (Columbia2).

**Fig. 4.1** Total incoming traffic measured at Columbia University.

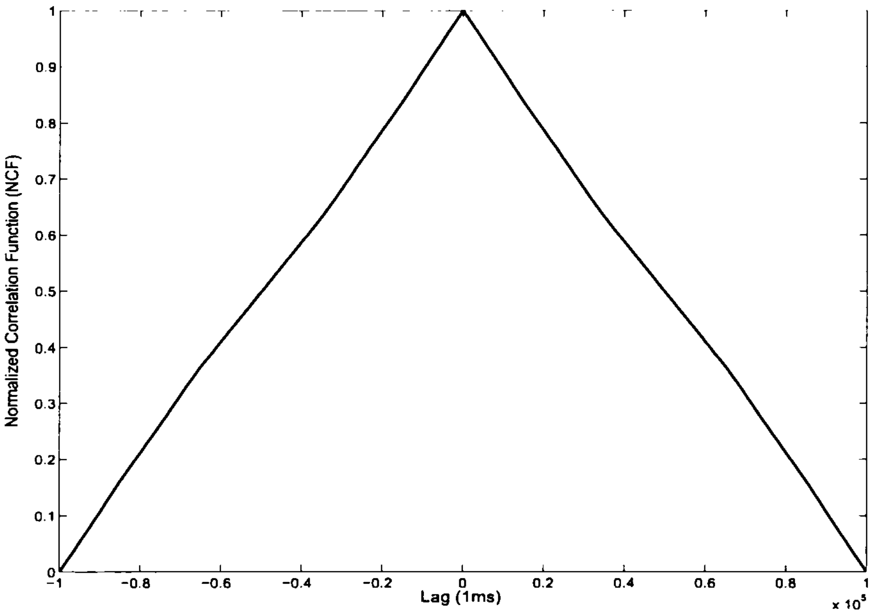


(a) Columbial NCF

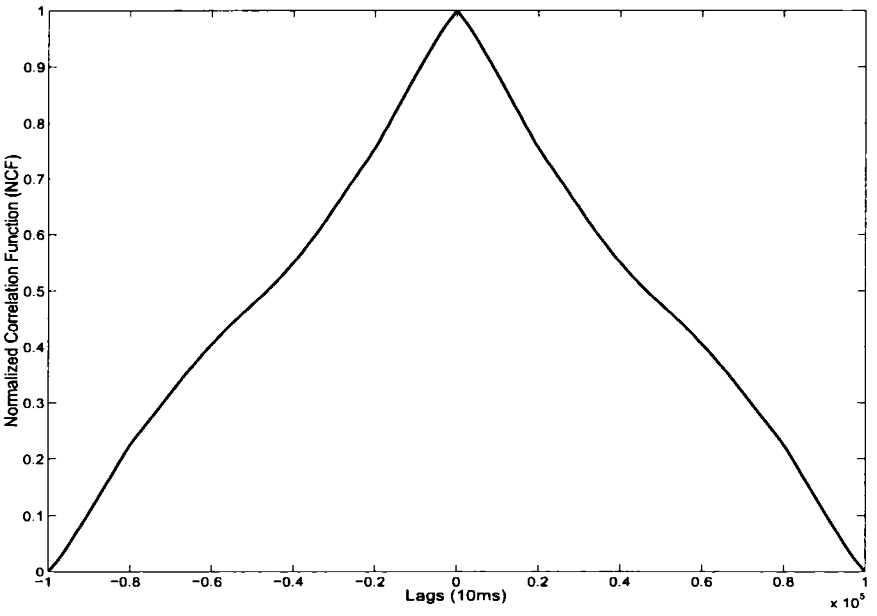


(b) Columbia2 NCF

**Fig. 4.2** Normalizcd Corrclation Function (NCF) of Columbia traffic traccs.



(a) Parameters: $\sigma_i^2 = 1 \times 10^4$, $\gamma_{ij}^2 = 1 \times 10^3$, $a = 0.8$, $A_{ij} = 1$, $H_{ij} = 0.3$.



(b) Parameters: $\sigma_i^2 = 1 \times 10^5$, $\gamma_{ij}^2 = 1 \times 10^4$, $a = 0.2$, $A_{ij} = 100$, $H_{ij} = 0.8$.

**Fig. 4.3** Normalized Correlation Function (NCF) of examples of the synthesized traffic generated by the proposed model.

## 4.3 Visual Interpretation of Empirical Traffic Traces

In this section, we decompose the Columbia traffic traces to several groups. We regard the monitored router in Columbia University as an edge node in an Agile All-Photonic Network. The measured traffic is considered as the total incoming traffic to an edge node. The total traffic is then decomposed to different destination edge nodes, which is implemented by dividing the packets to different groups according to their destination IP addresses. Fig. 4.4(a) and Fig. 4.4(b) shows the spatial distribution of the total incoming traffic between different destination groups.

Fig. 4.4(a) and Fig. 4.4(b) show that some traffic traces are slowly time-varying, for example that destined to group 1 or 2 in Fig. 4.4(a), while some are bursty, like those destined to group 7 in Fig. 4.4(a). The burstiness occurs in the traffic scenario with a quite low mean, such as panel 5, 6 and 7 in Fig. 4.4(a) and panel 4, 6, 7 and 8 in Fig. 4.4(b). In both traffic traces, the traffic volume in group1 is large relative to the others. We also noticed that most of the packets with a specific origin IP address have the same destination IP address. One explanation is that the time duration of the data traces is only around 90 seconds, during which there exists a dominant traffic flow for one host. The dominant traffic flows contribute significantly to the overall traffic, and the packets in the flow have the same IP source address and the same IP destination address.

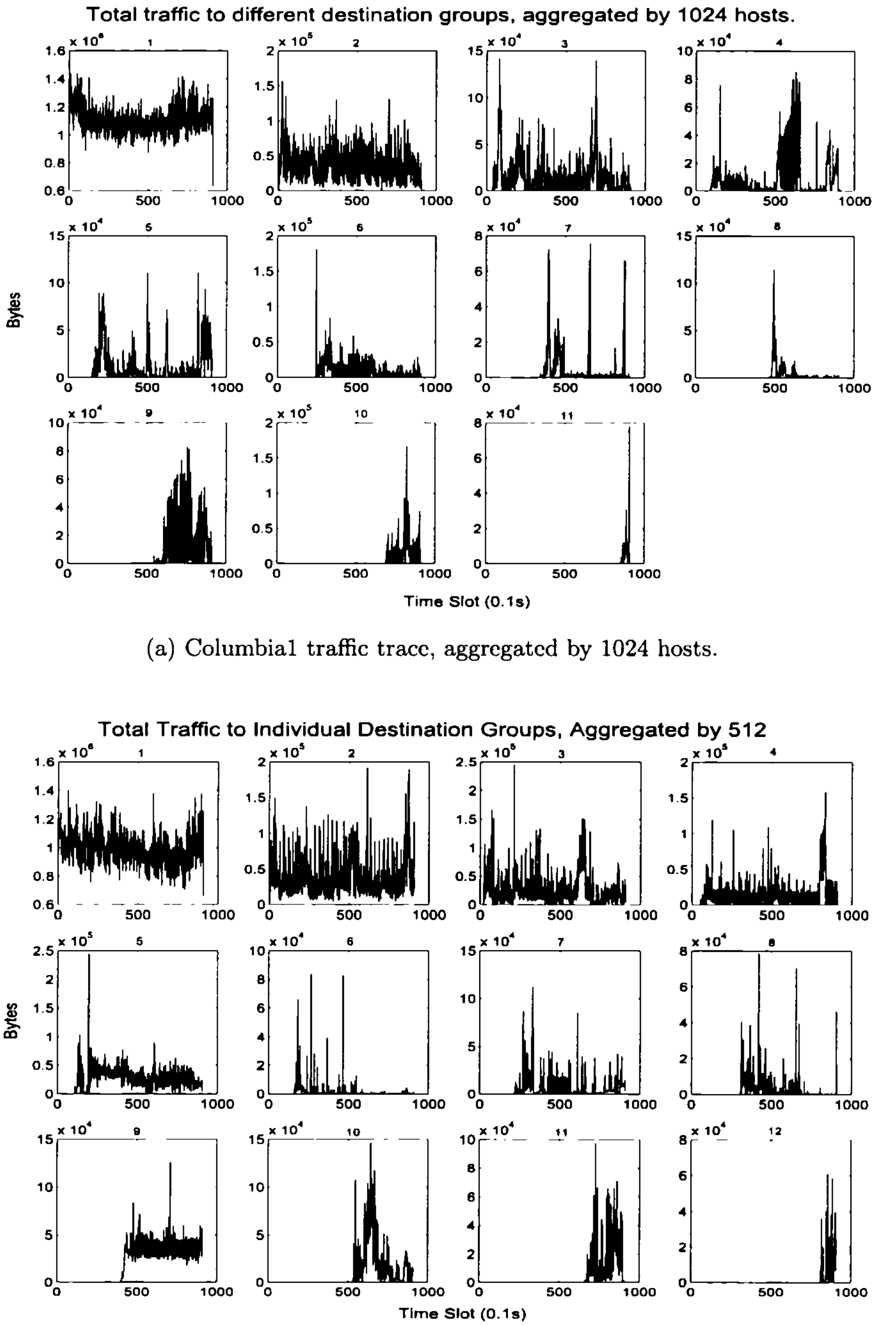
## 4.4 Wavelet-based Estimation of Long-Range Dependence

In this section, we estimate the parameters of long-range dependence and test the time constancy of scaling parameters the data traces, using a wavelet based estimator developed by Abry and Veitch[2].

### 4.4.1 The Long-Range Dependence Phenomenon

As discussed in Section 2.2.2, a common definition of long range dependence (LRD) [35] is the slow, power-law like decrease at large lag of the autocovariance function of a stationary stochastic process $x_t$ given by $r(k) \sim c_r|k|^{-(1-\alpha)}$, $\alpha \in (0,1)$. It can also be defined as the

[2] The codes to perform Long-range Dependence estimation and test the time constancy of scaling parameters are available at http://www.cubinlab.ee.mu.oz.au/~darryl/secondorder_code.html



(a) Columbia1 traffic trace, aggregated by 1024 hosts.

(b) Columbia2 traffic trace, aggregated by 512 hosts.

**Fig. 4.4** The empirical Columbia traffic traces, to different destination groups.

power law divergence at the origin of its spectrum:

$$f_x(\nu) \sim c_f|\nu|^{-\alpha}, \qquad |\nu| \to 0, \tag{4.3}$$

where $f_x(\nu)$ satisfies, in the case of discrete time processes, $r_x(0) = \sigma_x^2 = \int_{-1/2}^{1/2} f_x(\nu)d\nu$, $\sigma_x^2$ being the variance (or power) of $x_t$. This implies that the covariance $r(k)$ decays so slowly, that $\sum_{k=-\infty}^{\infty} r(k) = \infty$. $\alpha$ is the most important parameter, as the constant $\alpha$ governs the characteristic scaling behaviour of a LRD process as well as statistics derived from it. Scaling behaviour with a property of scale invariance is characterized by the fact that the statistics are governed by scales with some invariant parameters within a scaling range. The property of LRD can be interpreted as a scale invariant characteristic, because its spectrum takes the form $|\nu|^{-\alpha}$ for a range of frequencies $\nu$ close to 0.

A process $X = X(t), t \in \Re$ is self-similar with Hurst parameter $H > 0$ if $X(0) = 0$ and $X(ct), t \in \Re$ and $c^H X(t), t \in \Re$ have the same finite-dimensional distributions [32]. There is a close relationship between long-range dependence and self-similar processes. Indeed, the increments of any finite variance self-similar process with stationary increments have long-range dependence, as long as $1/2 < H < 1$, with $H$ and $\alpha$ related through

$$\alpha = 2H - 1. \tag{4.4}$$

### 4.4.2 Discrete Wavelet Transform

The discrete wavelet transform (DWT) is a representation of a signal $x$ using an orthonormal basis consisting of a countably-infinite set of wavelets [42, 43]. The coefficients of the discrete wavelet transform of a process $x$ are denoted as

$$d_{j,k} = \langle x, \psi_{j,k} \rangle, \qquad j \in \Re^+, t \in \Re, \tag{4.5}$$

where the signal $x$ is analyzed with a set of wavelet basis functions

$$\psi_{j,k}(t) = 2^{-j/2}\psi_0(2^{-j}t - k), \qquad j \in \Re^+, t \in \Re. \tag{4.6}$$

The basis is constructed from the mother wavelet $\psi_0$ by the action of a time-shift operator $\psi_{0,k}(t) = \psi_0(t-k)$ and a dilation (change of scale) operator $\psi_{j,0}(t) = 2^{-j/2}\psi_0(2^{-j}t)$. The mother wavelet $\psi_0$ is chosen such that both its spread in time and frequency are relatively limited. Wavelets that are often used in practice include the Haar wavelet and the Daubechies wavelet, indexed by a positive integer parameter $N$. The Haar wavelet $\psi_0(t)$ equals 1 at $0 \leq t < 1/2$, $-1$ at $1/2 \leq t < 1$ and 0 otherwise. The Daubechies wavelet with $N = 1$ is in fact the Haar wavelet, but the other Daubechies wavelets with $N > 1$ have $N$ vanishing moments: $\int t^k \psi_0(t)dt \equiv 0, k = 0, 1, 2, ..., N-1$. The more vanishing moments, the smoother the wavelet [60, 61]. The smoothness of a wavelet can provide numerical stability and better reconstruction of the signals. The LRD property can thus be interpreted as a scale invariance characteristic which is efficiently analyzed by wavelets. Please refer to Section 2.2.2 for more details about wavelet representations.

### 4.4.3 A Wavelet Based Joint Estimator of the Parameters of Long-Range Dependence

A wavelet based estimator for the joint parameters $(\alpha, c_f)$ is developed by Veitch and Abry in [4]. In this approach, several properties of the wavelet decomposition combine to reduce LRD in the time domain to short range dependence (SRD) in the wavelet representation. The key properties that give rise to this behaviour are the bandpass nature of the analyzing wavelets, the fact that the analyzing family of wavelets (and scaling functions) are generated from the change of scale operator which matches the power-law form of LRD spectra, and the fact that the number of vanishing 'moments' of the wavelets can be controlled.

The family of wavelet basis functions $\psi_{j,k}(t) = 2^{-j/2}\psi_0(2^{-j}t - k)$, $j = 1, ..., J$, $k \in Z$ is constructed from the dilation (change of scale) operator: $\psi_{j,0}(t) = 2^{-j/2}\psi_0(2^{-j}t)$. The scale invariance is captured exactly:

$$\mathbf{E}[d_{j,\cdot}^2] = 2^{j\alpha}c_f C, \tag{4.7}$$

where

$$C = \int |\nu|^{-\alpha}|\Psi_0(\nu)|^2 d\nu. \tag{4.8}$$

The relation in (4.7) can be rewritten as $\log_2(\mathbf{E}[d_{j,\cdot}^2]) = j\alpha + \log_2(c_f C)$. It strongly suggests a linear regression approach for estimating $(\alpha, c_f)$, where clearly the slope of the regression would estimate $\alpha$ and the intercept would be related to $c_f$.

$\mathbf{E}[d_{j,\cdot}^2]$ is estimated by an unbiased and efficient estimator

$$\mu_j = \frac{1}{n_j}\sum_{k=1}^{n_j} d_{j,k}^2, \tag{4.9}$$

where $n_j$ is the number of coefficients at octave $j$ available for analysis. By setting $y_j = \log_2(\mu_j)$, we can define a linear regression $\mathbf{E}y_j = bj + a$. Define the quantities $S = \sum 1/\sigma_j^2$, $S_x = \sum j/\sigma_j^2$ and $S_{xx} = \sum j^2/\sigma_j^2$, where $\sigma_j^2 = Var(y_j)$. The unbiased estimator $(\hat{b}, \hat{a})$ of $(b, a)$ is

$$\hat{b} = \sum w_j y_j, \tag{4.10}$$

$$\hat{a} = \sum v_j y_j, \tag{4.11}$$

where

$$w_j = \frac{(Sj - S_x)/\sigma_j^2}{S_{xx}S - S_x^2} v_j = \frac{(S_{xx} - S_x j)/\sigma_j^2}{S_{xx}S - S_x^2}$$

The joint unbiased estimator $(\hat{\alpha}, \widehat{c_f C})$ is given by

$$\hat{\alpha} = \hat{b}, \tag{4.12}$$

$$\widehat{c_f C} = p \cdot 2^{\hat{a}} \tag{4.13}$$

where

$$p = \prod \frac{\Gamma(n_j/2)exp(\psi(n_j/2)v_j)}{\Gamma(v_j + n_j/2)}$$

is a bias correcting factor.

Define the estimator $\hat{C}$ of the integral $C(\alpha, \psi_0) = \int |\nu|^{-\alpha}|\Psi_0(\nu)|^2 d\nu$ as

$$\hat{C} = \begin{cases} C(0, \Psi_0) & \hat{\alpha} \le 0 \\ C(\hat{\alpha}, \Psi_0) & 0 < \hat{\alpha} < 1 \\ C(1, \Psi_0) & \hat{\alpha} \ge 1 \end{cases} \tag{4.14}$$

The LRD estimator $(\hat{\alpha}, \hat{c_f})$ is then given by

$$\hat{\alpha} = \hat{b}, \tag{4.15}$$

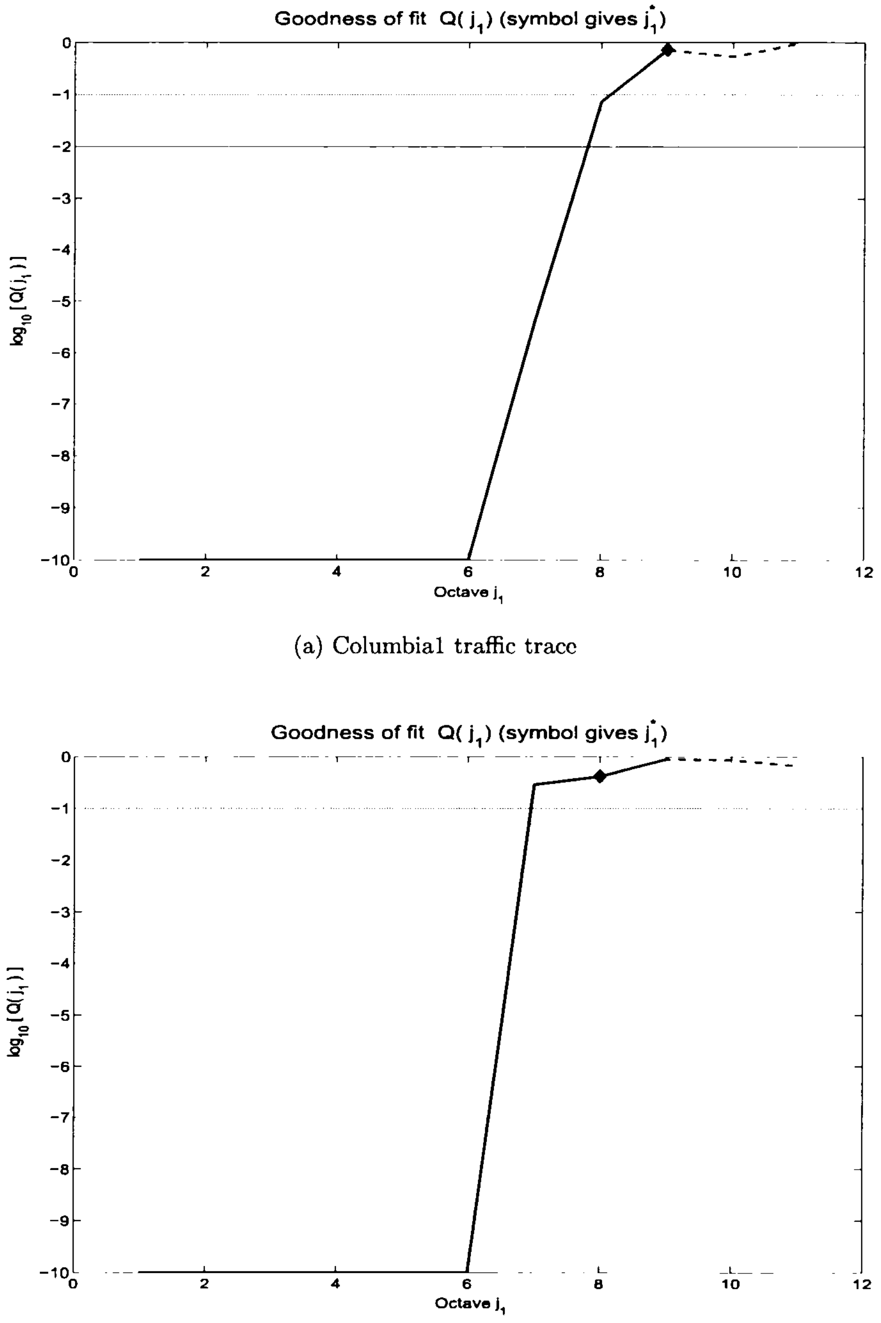
$$\widehat{c_f} = \widehat{c_f C}/\hat{C}. \tag{4.16}$$

### 4.4.4 Traffic Trace Analysis

We apply the estimator is applied to Columbia data traces. We extract an aggregated rate process of arriving traffic, that is, a discrete time series corresponding to the number of bytes transmitted during contiguous constant length time intervals, here of length 1ms.

**Choice of scales** The range of scales $(j_1, j_2)$ over which a scaling phenomenon exists varies. For LRD, $j_2$ is infinite, but $j_1$, where the LRD "begins", must be chosen. Fig. 4.5(a) shows a graph of $Q(j_1)$, the goodness of fit measure with $j_2$ fixed, against $j_1$. The goodness-of-fit measure increases rapidly from $j_1 = 6$ to 8, and then stabilizes. This can be used as the basis of selecting the optimal $j_1$ value. As seen in Fig. 4.5(a), $j_1$ can be selected as 10, which means the LRD "begins" at the time scale of $1\text{m}s \times 2^{10} = 1024\text{m}s$ in Columbial traffic trace. In Fig. 4.5(b), $j_1$ is selected as 8 and LRD "begins" at the time scale of $2^7 = 128$ms. The goodness-of-fit measure choose an "optimal" $j_1$ value, which defines a non-decreasing zone beginning from $j_1$, and finds the last time scale that an improvement of $Q$ (larger than a specified value) is found, then adds one to it [4].

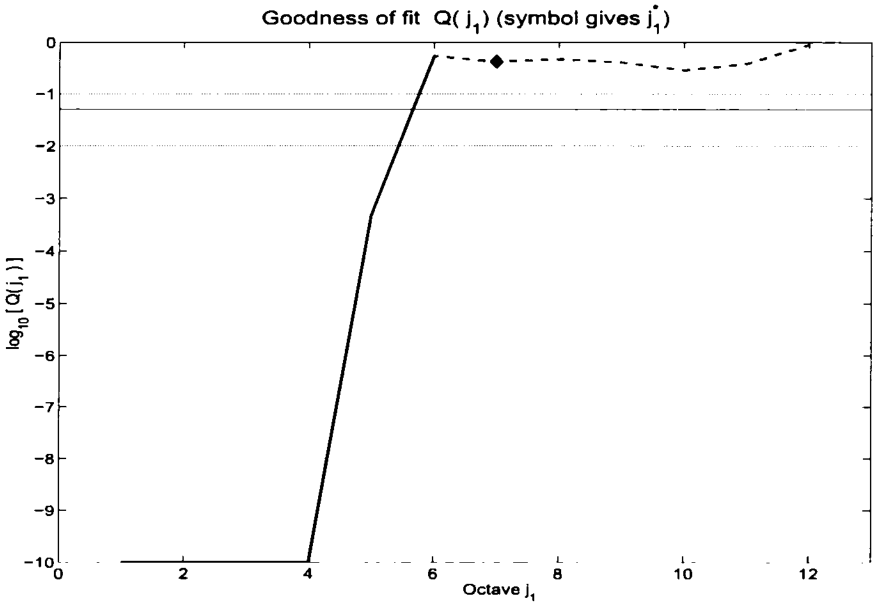
**Logscale Diagram** The (second order) Logscale Diagram (LD) consists in the graph of $y_j$ against scale $j$, together with confidence intervals about the $y_j$, as shown in Fig. 4.7(a) and Fig. 4.7(b) for the two Columbia data traces, respectively. Scaling behaviour is detected through the region(s) of alignment observed in the log-log plot. An alignment region is a range of scales where the $y_j$ fall on a straight line. Estimation of scaling parameters is effectively performed over the region(s). If the scaling parameter $\alpha$ is estimated to lie in $(0, 1)$, and the alignment region of scales is from some initial value $j_1$ up to the largest one in the data, then the scaling corresponds to a LRD. If $\alpha$ is greater than 1 over all or almost all of the scales in the data, then the scaling indicates self-similarity with the relevant Hurst parameter $H = (\alpha - 1)/2$. As shown in Fig. 4.7(a) and Fig. 4.7(b), the scaling exponent $\alpha$ is estimated to be 0.5783 and 0.52 respectively, and both lie in the region of $(0, 1)$. Alignment is observed over scales $[j_1, j_2] = [10, 14]$ in Fig. 4.7(a) and $[j_1, j_2] = [8, 13]$ in Fig. 4.7(b). The scaling can be identified as LRD as the value is in the correct range , $\hat{\alpha} \in (0, 1)$, and the alignment region includes the largest scales in the data. A goodness of fit statistic "$Q$" is determined to help with the choice of scaling range,



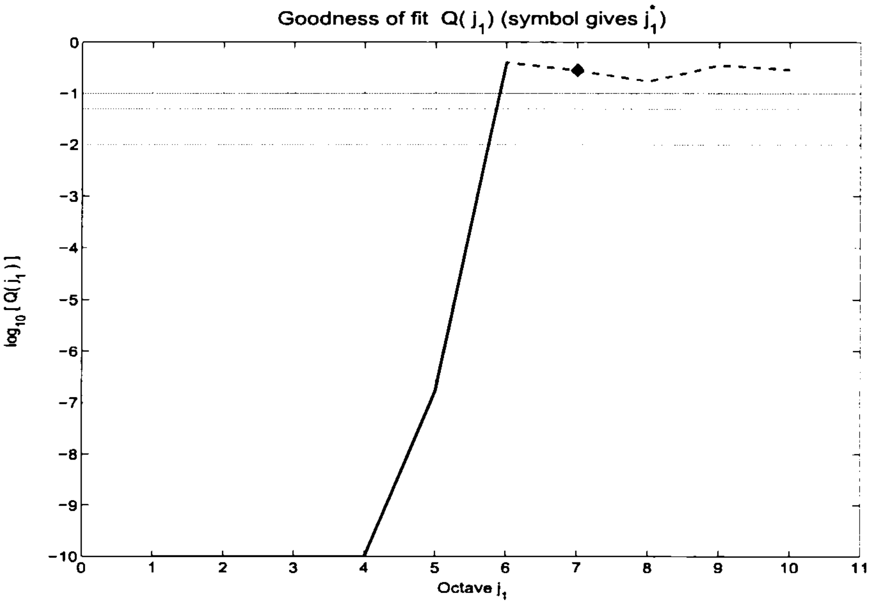
(a) Columbial traffic trace

(b) Columbia2 traffic trace

**Fig. 4.5** Goodness of fit $Q(j_1)$ as a function of $j_1$, based on the empirical Columbia traffic traces.

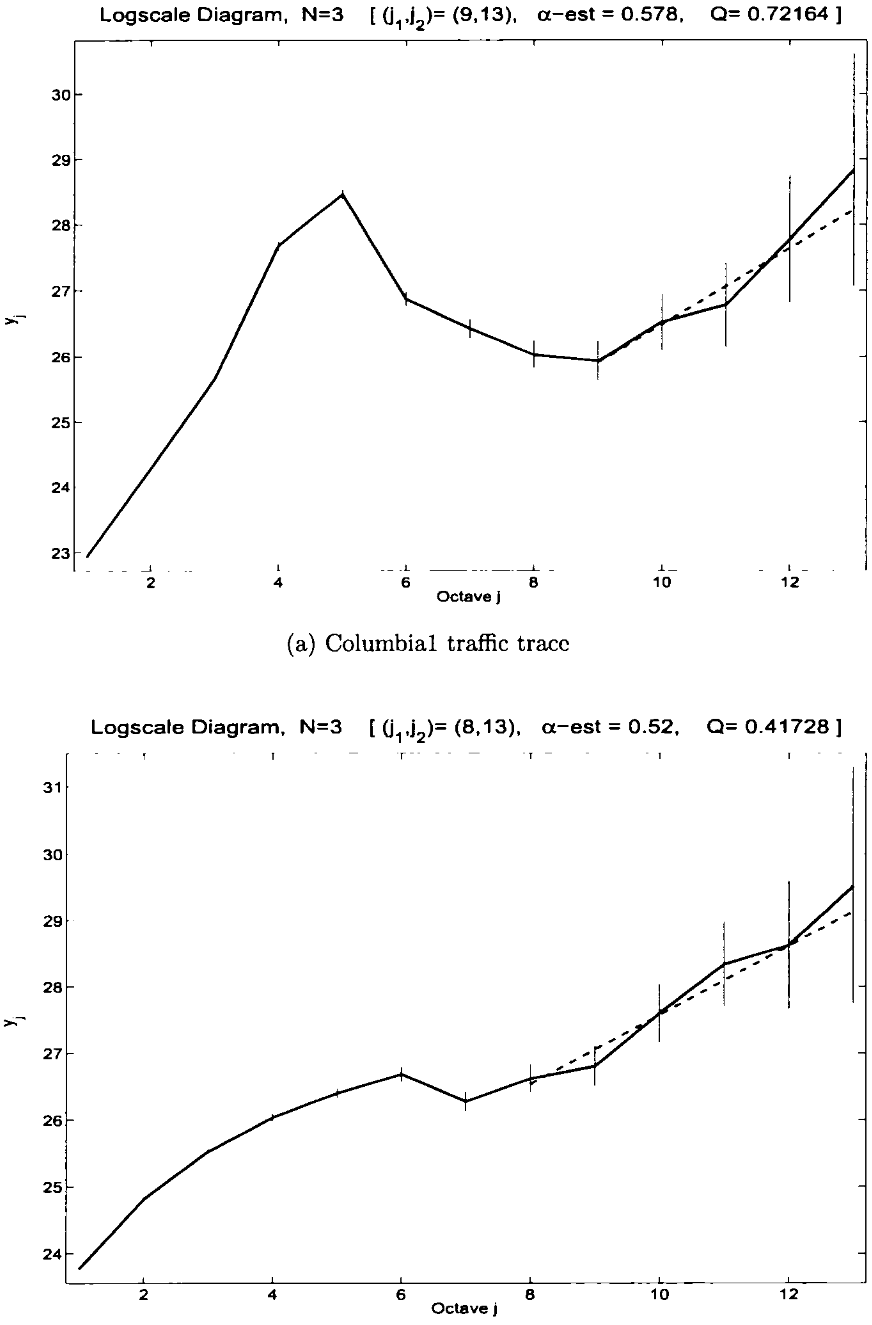


(a)



(b)

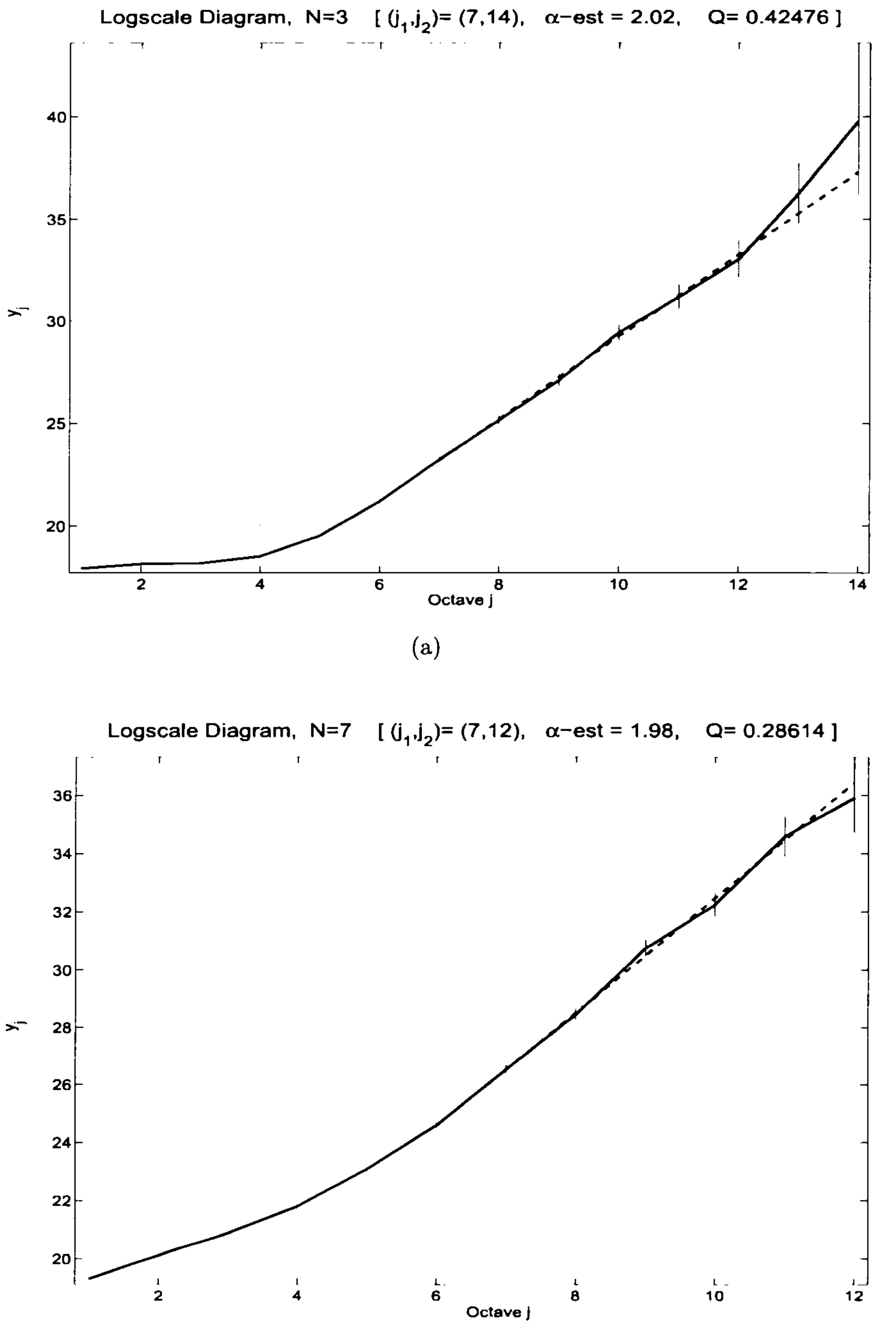
**Fig. 4.6** Goodness of fit $Q(j_1)$ as a function of $j_1$, based on the traffic trace generated by the proposed model with parameters (a) $\sigma_i^2 = 1 \times 10^4$, $\gamma_{ij}^2 = 1 \times 10^3$, $a = 0.8$, $A_{ij} = 1$, $H_{ij} = 0.3$; and (b) $\sigma_i^2 = 1 \times 10^5$, $\gamma_{ij}^2 = 1 \times 10^4$, $a = 0.2$, $A_{ij} = 100$, $H_{ij} = 0.8$.



(a) Columbial traffic trace

(b) Columbia2 traffic trace

**Fig. 4.7** Logscale diagram, based on the empirical Columbia traffic traces.



(a)

(b)

**Fig. 4.8** Logscale diagram, based on the traffic trace generated by the proposed model with parameters (a) $\sigma_i^2 = 1 \times 10^4$, $\gamma_{ij}^2 = 1 \times 10^3$, $a = 0.8$, $A_{ij} = 1$, $H_{ij} = 0.3$; and (b) $\sigma_i^2 = 1 \times 10^5$, $\gamma_{ij}^2 = 1 \times 10^4$, $a = 0.2$, $A_{ij} = 100$, $H_{ij} = 0.8$.

which is $Q = 0.72164$ in Fig. 4.7(a) and $Q = 0.41728$ in Fig. 4.7(b). $Q$ is the probability of observing the data given that the expectations of the variance estimates at each scale really do follow the defining linear form of linear regression. A value greater than say 0.05 is acceptable.

This tool is also used to analyze the synthesized traffic generated by the proposed model. Fig. 4.6(a), 4.6(b), 4.8(a) and 4.8(b) present the analysis for the synthesized traffic with parameters $\{\sigma_i^2 = 1 \times 10^4$, $\gamma_{ij}^2 = 1 \times 10^3$, $a = 0.8$, $A_{ij} = 1$, $H_{ij} = 0.3\}$ and $\{\sigma_i^2 = 1 \times 10^5$, $\gamma_{ij}^2 = 1 \times 10^4$, $a = 0.2$, $A_{ij} = 100$, $H_{ij} = 0.8\}$, respectively. As shown in Fig.4.8(a) and 4.8(b), the parameter $\alpha$ is greater than 1, which are 2.02 and 1.98, respectively, and measured over a range including the largest scales. It implies that the traffic is self-similar, which is consistent with the element of fractional Gaussian noise in our proposed model. The Hurst parameters are expressed as $H = (\alpha - 1)/2$, which are 0.51 and 0.49 respectively. Long-range dependence is the characteristic which is visible in multi-second scales [58] while network traffic can be well represented by the Poisson model for sub-second time scales. Our proposed model focuses on short-range traffic modelling and prediction. This model attempts to capture the characteristics of network traffic at a time-scale which is the round-trip time between the edge node and core node in AAPN, which is about $10 - 80$ms. At this time scale, long-range dependence is not evident.

## 4.5 Testing Time Constancy of Scaling Parameters

The high variability inherent in scaling processes is very easily confused with non-stationarity, both in a "judgement by eye" sense, and in the sense of poor robustness and performance of many standard statistical tools. Sometimes, the scaling behaviour present in a Logscale Diagram is in fact a kind of average of different phenomena taking place in individual data blocks. It is important to examine the constancy of scaling using Logscale Diagrams over data blocks [62].

### 4.5.1 Definition of the Test on Time Constancy of Scaling Parameters

Let $x$ denote the series to be analyzed of length $n$. The series is split into $m$ adjacent non-overlapping blocks, and wavelet-based estimates of the scaling parameters $\{\hat{\alpha}_1, ..., \hat{\alpha}_i, ..., \hat{\alpha}_m\}$, are obtained from $m$ adjacent blocks with length $n_i$, over a common scale range $[j_1, j_2]$. $\{\hat{\alpha}_1, ..., \hat{\alpha}_i, ..., \hat{\alpha}_m\}$ can be considered as uncorrelated Gaussian variables with unknown means and known, but possibly different variances. That is

$$\hat{\alpha}_i \sim N(\alpha_i, \sigma_i^2), \tag{4.17}$$

$$\sigma_i^2 = \frac{2^{j_1-1}(1-2^{-J})}{(\ln^2 2(1-2^{-J}(J^2+4)+2^{-2J}))n_j}. \tag{4.18}$$

To an excellent approximation, $n_i$ being the number of samples in the $i$th block, $J = j_2 - j_1 + 1$ being the width of the scaling range. Testing whether the scaling parameter is constant or not therefore amounts to testing whether $\hat{\alpha}_i$ have identical means or not. We wish to test the null hypothesis $H_0$: the means are identical, against $H_1$: the means differ.

A Uniformly Most Powerful Invariant (UMPI) test is defined by forming the statistic

$$V = \sum \frac{1}{\sigma_i^2}(\hat{\alpha}_i - \frac{\sum \hat{\alpha}_i/\sigma_i^2}{\sum \frac{1}{\sigma_i^2}})^2, \tag{4.19}$$

whose distribution is a function only of the single parameter

$$\theta = \sum \frac{1}{\sigma_i^2}(\alpha_i - \frac{\sum \alpha_i/\sigma_i^2}{\sum \frac{1}{\sigma_i^2}})^2. \tag{4.20}$$

Under $H_0$ we have $\theta = 0$ and $V$ is distributed as a Chi-squared variable with $m-1$ degrees of freedom with density $f_{m-1}$. Under $H_1$, $\theta > 0$ and $V$ is distributed as a non central Chi-squared distribution with $m-1$ degrees of freedom and non centrality parameter $\theta$, with density $f_{m-1,\theta}$.

Let $\beta$ denote the chosen significance level of the test and define the critical region boundary $C = C(\beta)$ via

$$\int_C^{+\infty} f_{m-1}(x)dx \equiv \beta. \tag{4.21}$$

The test reads:

If $V > C$, Reject $H_0$ (in critical region, conclusion, $\alpha$ is not constant),

If $V \leq C$, Accept $H_0$ (conclusion, no evidence that $\alpha$ is not constant).

The power function of the test, that is the probability, as a function of the particular $H_1$ as specified by $\theta$, of accepting $H_1$ when it is true, reads:

$$P(\theta) = \int_C^{+\infty} f_{m-1,\theta}(x)dx. \tag{4.22}$$

### 4.5.2 Choosing the number of blocks $m$

When applying the test to a set of data of length $n$, the number $m$ and sizes $\{n_i\}$ of the blocks into which the data is split have to be selected. For simplicity the blocks are of equal size. For a given $m$, the common variance $\sigma_m^2$ of the estimates $\hat{\alpha}_i$ is roughly inversely proportional to the length of the blocks. Hence $\sigma_m^2$ is roughly proportional to $m$. The choice of optimal $m$ is subject to the trade-off: $m$ has to be large enough to track variations in time of the scaling parameter, but also has to be as small as possible to avoid degrading the power of the test due to an increase in variance of the estimates. Therefore the optimal choice is for $m$ to be such that the data be split into the largest possible blocks within which the scaling parameter is not varying.

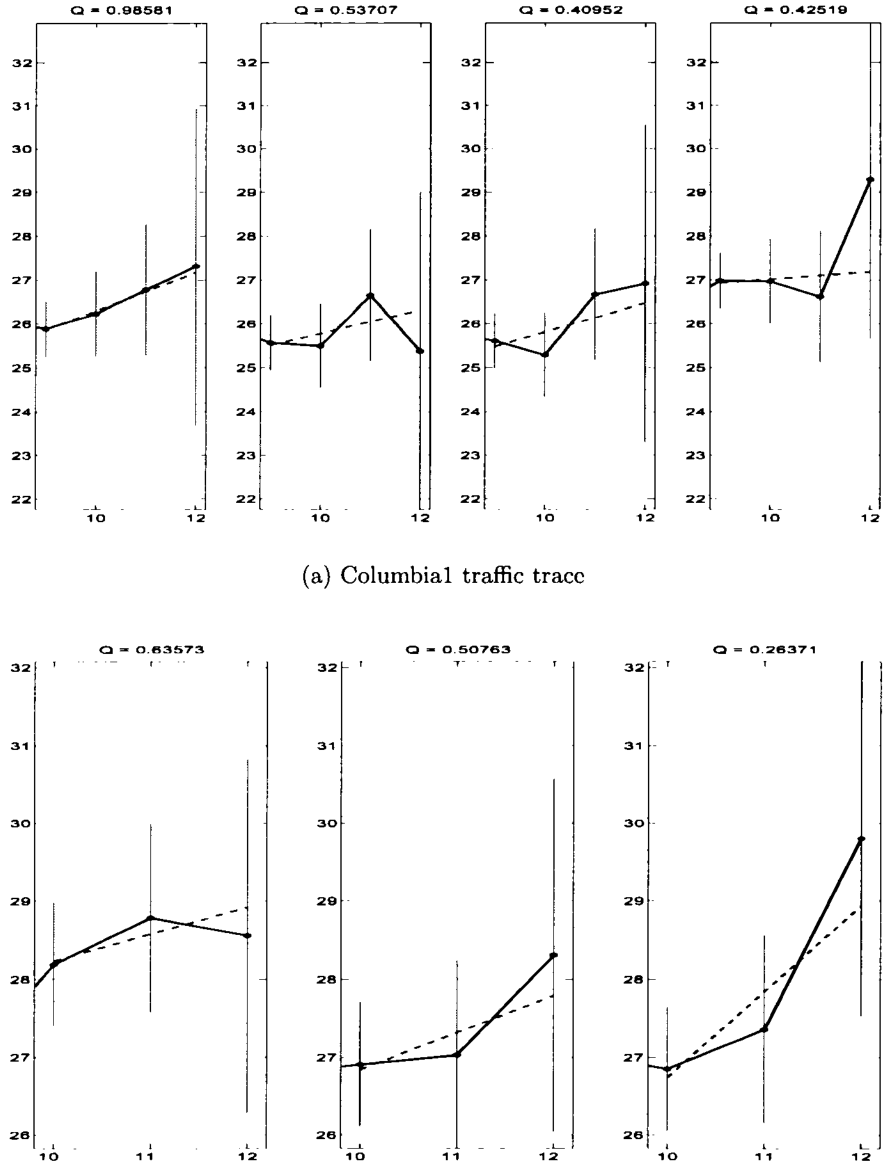
### 4.5.3 Measured traffic trace analysis

We apply the test to the traffic traces of Columbia University. The traffic traces consist of lists of arriving time stamp and length of all packets recorded on a gateway of this university. We extracted an aggregated rate process of arriving work, generating a discrete time series corresponding to the number of bytes transmitted during contiguous time intervals with length of 1ms. The time series are plotted in the top plot in Fig. 4.10 and Fig. 4.11.

For the entire time series the Logscale Diagram has already been computed using Daubechies3 wavelets [60] in Fig. 4.7(a) and Fig. 4.7(b). The Logscale Diagrams exhibit a good alignment over a range of scales $[9, 14]$ and $[7, 13]$ respectively, including the largest scale in the data, and yields an estimated slope (scaling exponent) $\hat{\alpha} = 0.562$ and 0.437 in the range $0 < \hat{\alpha} < 1$.

In order to investigate the time constancy of scaling in this series, we split the Columbia1 traffic trace into $m = 4$ equally spaced blocks. Over each block the following quantities are examined: mean, variance, $\alpha$, $c_f$, and the Logscale Diagram is computed individually, as shown in Fig. 4.9(a) and Fig. 4.10. The $m$ estimates, made in each case with $[j_1, j_2] = [9, 12]$, are performed as if they are mutually independent. For each statistic as shown in Fig. 4.10, at the far left of the plot the sample value and confidence interval is calculated over the entire series. For the confidence intervals on the mean estimates, theoretical asymptotic confidence intervals are displayed based both on IID assumptions and LRD assumptions. The result of the time constancy test of scaling parameter $\alpha$, and the critical level corresponding to the data, is plotted over the graph. The critical level is 0.90 and critical region is to the right of 7.815. The result indicates that the hypothesis of constant scaling is accepted, and the estimated scaling exponent $\hat{\alpha} = 0.578$. Similar testing is also held for the Columbia2 data trace. Fig. 4.9(b) and Fig. 4.11 show that the assumption of time constancy of scaling is accepted, and the estimated scaling exponent $\hat{\alpha} = 0.437$. The constant scaling reveals that the scaling behaviours exist in each data block, and it is not an effect by superposing individual data blocks.

Fig. 4.12(a), 4.13, 4.12(b) and 4.14 show testing of the synthesized traffic traces generated by the proposed model and the parameters selection are presented in the figure. The synthesized traffic traces are splitted into $m = 5$ and $m = 6$, respectively. The $m$ independent estimates are made with the range of $[7, 11]$. The results show that assumption of the time constancy of the scaling exponent are both accepted, and the estimated scaling parameters are 2.02 and 1.98, respectively.



(a) Columbial traffic trace

(b) Columbia2 traffic trace

**Fig. 4.9** Goodness of fit $Q$ for each block, based on the empirical Columbia traffic traces.

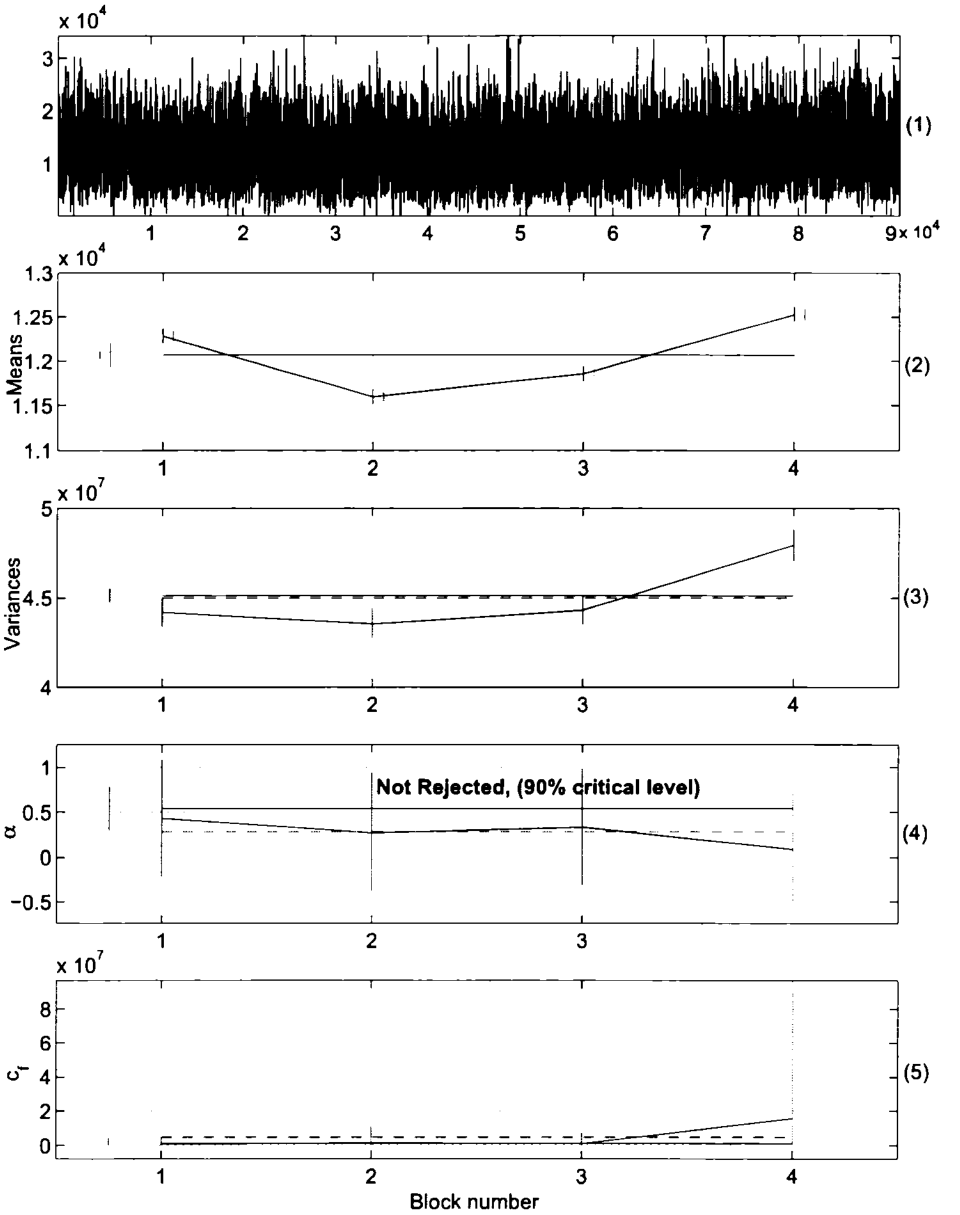


**Fig. 4.10** Stationarity testing of Columbial traffic trace.
(1) The time series of bytes per time unit (1ms); (2) Means over the blocks, the horizontal line gives the overall mean; (3) Variances over the blocks,. the horizontal line gives the overall variance; (4) Scaling parameter of the subseries. The solid (dashed) horizontal line gives the overall (average) value. (5) The second scaling parameter of the subseries. The solid (dashed) horizontal line gives the overall (average) value.

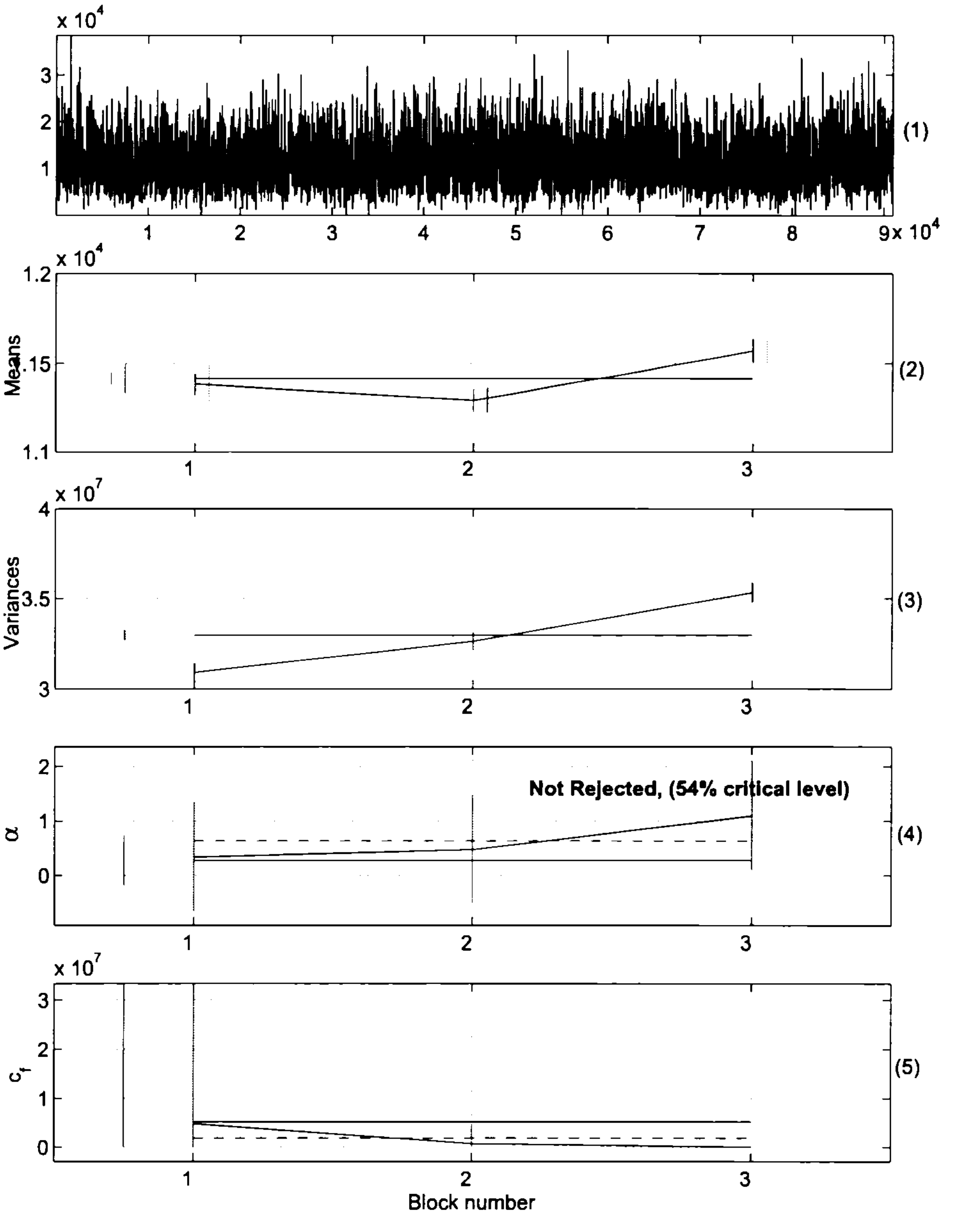
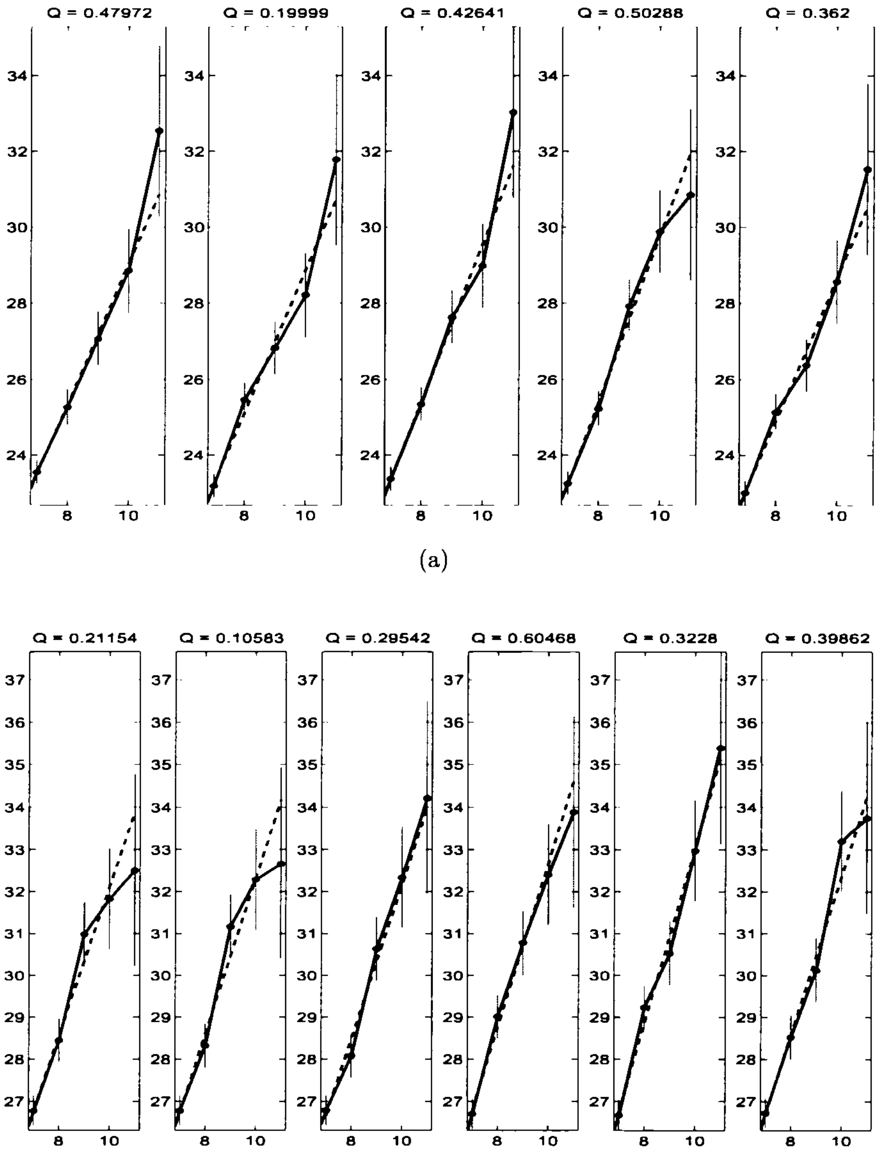


**Fig. 4.11** Stationarity testing of the Columbia2 traffic trace.
(1) The time series of bytes per time unit (1ms); (2) Means over the blocks, the horizontal line gives the overall mean; (3) Variances over the blocks,. the horizontal line gives the overall variance; (4) Scaling parameter of the subseries. The solid (dashed) horizontal line gives the overall (average) value. (5) The second scaling parameter of the subseries. The solid (dashed) horizontal line gives the overall (average) value.



(a)

(b)

**Fig. 4.12** Goodness of fit $Q$ for each block, based on traffic traces generated by the proposed model with parameters
(a) $\sigma_i^2 = 1 \times 10^4$, $\gamma_{ij}^2 = 1 \times 10^3$, $a = 0.8$, $A_{ij} = 1$, $H_{ij} = 0.3$; and (b) $\sigma_i^2 = 1 \times 10^5$, $\gamma_{ij}^2 = 1 \times 10^4$, $a = 0.2$, $A_{ij} = 100$, $H_{ij} = 0.8$.

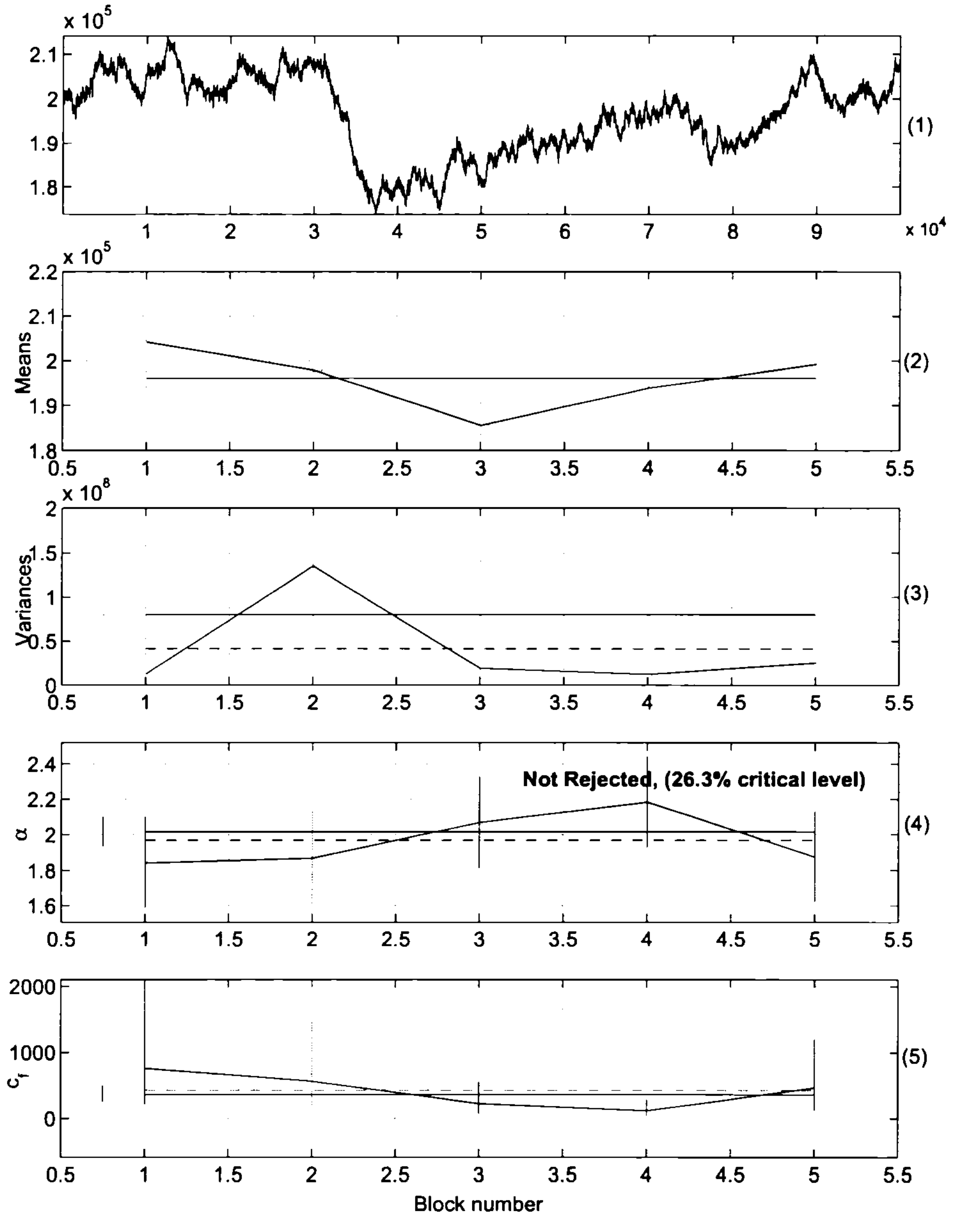


**Fig. 4.13** Stationarity testing of traffic traces generated by the proposed model with parameters $\sigma_i^2 = 1 \times 10^4$, $\gamma_{ij}^2 = 1 \times 10^3$, $a = 0.8$, $A_{ij} = 1$, $H_{ij} = 0.3$. (1) The time series of bytes per time unit (1ms); (2) Means over the blocks, the horizontal line gives the overall mean; (3) Variances over the blocks,. the horizontal line gives the overall variance; (4) Scaling parameter of the subseries. The solid (dashed) horizontal line gives the overall (average) value. (5) The second scaling parameter of the subseries. The solid (dashed) horizontal line gives the overall (average) value.

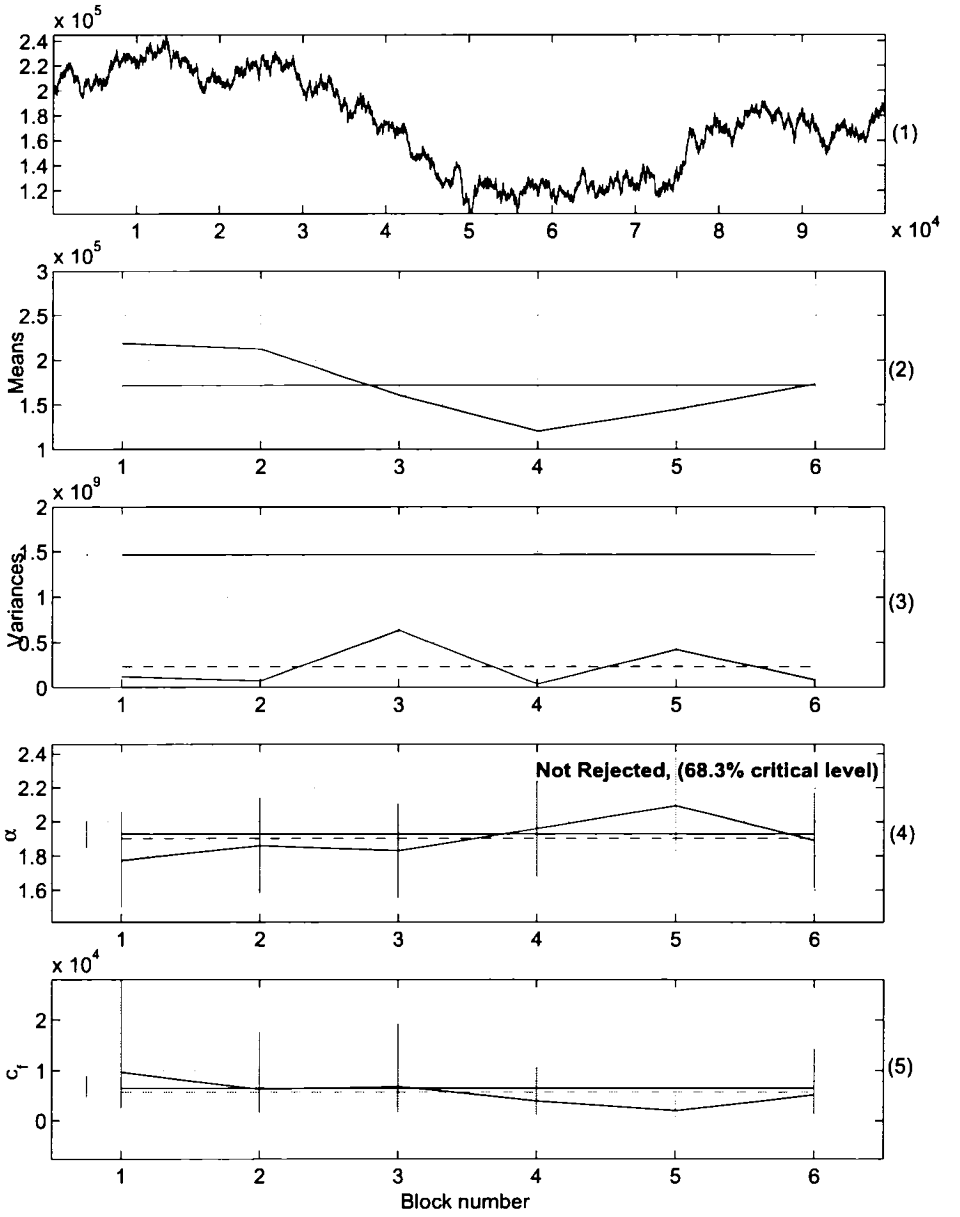


**Fig. 4.14** Stationarity testing of the traffic traces generated by the proposed model with parameters $\sigma_i^2 = 1 \times 10^5$, $\gamma_{ij}^2 = 1 \times 10^4$, $a = 0.2$, $A_{ij} = 100$, $H_{ij} = 0.8$. (1) The time series of bytes per time unit (1ms); (2) Means over the blocks, the horizontal line gives the overall mean; (3) Variances over the blocks,. the horizontal line gives the overall variance; (4) Scaling parameter of the subseries. The solid( dashed) horizontal line gives the overall (average) value. (5) The second scaling parameter of the subseries. The solid (dashed) horizontal line gives the overall (average) value.

# Chapter 5

# Conclusions and Future Work

This thesis has focussed on the traffic modelling and traffic analysis. We proposed a dynamic parametric traffic model which is intended to capture the traffic behavior in an all-photonic network in the time scale of 10 – 80ms. We implemented the Long-Rang Dependence and Stationarity testing tools to analyze the measured and synthesized traffic traces. This chapter will provide a summary of the thesis work. Additionally, directions for future work will be discussed.

## 5.1 Thesis Summary

In Chapter 1, we introduced the emergence of the all-photonic networks, and gave a brief description of the concept of an Agile All-Photonic Network (AAPN) [3]. This chapter presented the proposed overlaid star architecture of AAPNs and described how the control functionality is implemented. The motivation of traffic modelling and prediction in the context of AAPN control was explained.

Chapter 2 presented a review on the recent literature in the fields of network tomography, network traffic modelling, and traffic prediction. Section 2.1 first provided a generalized formulation of the network tomography problem, and the tasks of link-level and path-level network performance parameter estimation were discussed in more detail. In link-level parameter estimation, different approaches exist applicable in both multicast and unicast networks. In path-level parameter estimation, an important example is estimation of the traffic matrix between specified OD pairs, based on the measurements made on the links. Section 2.2 presented a survey on network traffic models that have been described in the literature. It first introduced the classical Poisson model which accurately captures the statistical behavior of voice traffic in telephony networks and then explained the failure of Poisson based models in wide area data networks. Some examples of stationary traffic models were described, including both short-range dependent and long-range dependent models. This section also presented wavelet-representations for traffic, which can mitigate the complication induced by the short- and long-range dependent temporal dependence. Section 2.3 described some methods for traffic prediction in both telephony and data networks, and identified several prediction techniques, including Kalman filtering, extended Kalman filtering and sequential Monte Carlo methods.

In Chapter 3, we proposed a new traffic modelling approach for all-photonic networks. The proposed dynamic parametric traffic model maintains state parameters that capture the underlying rates of the traffic between OD pairs (Section 3.2) and each state parameter is determined by a Gravity-model decomposition of a shared rate component for the origin node (Section 3.3), and its previous value. The observed traffic between origin-destination pair is determined by a Poisson process with the rate of state parameter and added fractional Gaussian noise, as described in Section 3.4. Section 3.5 described the combined model and Section 3.6 presented some basic statistical analysis for this model. Section 3.7 presented some examples of the synthesized traffic generated by the proposed dynamic parametric traffic model and discussed the effects of varying the parameters of this model.

Chapter 4 presented data analysis on the empirical and synthesized traffic traces. The Normalized Correlation Function (NCF) was evaluated to measure the similarity between the time series quantized from the measured and synthesized traffic traces. Section 4.4 described a wavelet-based estimator of Long-Range Dependence (LRD) [4]. The parameters of LRD were estimated for the traffic traces, and the time constancy of the scaling parameters was tested.

## 5.2 Future Work: Application of the model in Agile All-Photonic Networks

As described in Chapter 1, the AAPN is an agile self-configuring network [3], in which a number of core nodes are used to connect high capacity, fast switching edge nodes. The central controllers select paths through associated core nodes and reconfigure internal paths in response to dynamic changes in traffic loads based on reconfiguration requests generated by edge nodes. The edge nodes are geographically distributed. The AAPN project proposes an overlaid star topology. The overlaid star topology has the advantages that it is extremely efficient from the switching viewpoint, robust, and potentially very scalable. In the overlaid star network, the central controller is either combined or collocated with the core node.

AAPNs are intended to support three different switching schemes. The first type is that an entire wavelength is dedicated to the connection between two edge nodes, which fits the scenarios where the traffic between the two relatively large edge nodes is persistent with a substantial volume. The wavelengths are reallocated periodically on the order of hundreds of seconds, by measuring whether the wavelength is used efficiently. The second scheme is time slot reservation, in which each wavelength is divided into frames consisting of a fixed number of time slots. The time slots in the frames are allocated to the connections between edge nodes and the allocation is adjusted as the traffic demand varies on the order of seconds, which is appropriate for the connections with slowly-varying demand. The last scheme is optical burst switching, in which some of the wavelengths are dedicated for the unanticipated bursts in traffic demand.

In the scenario of time slot reservation, the time slots can be either requested by the edge nodes or directly allocated by the central controller. In the first case, the edge nodes send the time slot requests to the central controller and get the feedback for time slot allocation, which introduces a reservation delay. In order to mitigate the effects of the delay, the edge nodes have to send requests well in advance of traffic arrival. Therefore, short-range prediction of origin-destination traffic demands is helpful, allowing the edge nodes to request and relinquish the time slots based on a predicted upper bounds on traffic demands. The central controller allocates time slots to each origin-destination pair based on the requests sent from the edge nodes. In the second case, the central controller allocates the time slots to individual origin-destination pairs by predicting the traffic demands itself based on the information it gathers.

If during the time interval of one frame $\triangle t$, the actual traffic volume flowing into an origin edge node to some destination edge node is more than the capacity of the time slots allocated to this origin-destination pair, some of the traffic will be stored in the buffer at the origin node and may be sent out in the next frame. Denote by $u_{ij}(k)$ the traffic flowing into origin edge node $i$ and destined to edge node $j$ during the last time interval $k-1$. It is assumed that the traffic flowing into the origin node during time interval $k-1$ will be processed and transmitted in the next time interval $k$. Denote by $q_{ij}(k)$ the queue length of OD pair $(i, j)$ measured in origin node $i$ at the beginning of time interval $k$ (this is the traffic left from the last time interval $k-1$). Denote the capacity of one time slot as $C$, and $K_{ij}(k)$ as the number of time slots allocated to the connection between origin node $i$ and destination node $j$ at time interval $k$. $d_{ij}(k)$ is the traffic flowing into the AAPN core network from origin node $i$ destined to edge node $j$ during the time interval $k$. Then the observation equations can be described as follows

$$q_{ij}(k) = u_{ij}(k-1) + q_{ij}(k-1) - d_{ij}(k-1) \tag{5.1}$$

$$d_{ij}(k) = \begin{cases} u_{ij}(k) + q_{ij}(k), & K_{ij}(k)C_{ij} > u_{ij}(k) + q_{ij}(k), \\ K_{ij}(k)C, & \text{otherwise.} \end{cases} \tag{5.2}$$

The task of traffic demands prediction can be implemented either in edge nodes or in the central controller. In the first approach, the edge nodes can get the direct information of the traffic arrivals by measuring $u_{ij}(k)$, which assures the prediction can be relatively more accurate. The central controller just reacts to the requests from the edge nodes, and it is necessary to design a scheme to give fair access to resources for all origin-destination pairs. The distributed calculation of prediction in each edge node reduces the computation burden of the central controller. On the other hand, the edge nodes have to send a signalling message to the central controller for the predicted upper bounds which will increase the signalling flow between edge nodes and central controller. Moreover, the central controller only gets limited information such as the decision made on the predicted traffic demands, which makes it harder for the central controller to design a fair resource allocation scheme.

In the second approach, the central controller itself predicts the traffic demands by meauring the traffic flow between individual origin-destination pairs. The central controller can get the information of the occupancy of the time slots in two ways. In the first method, the origin node $i$ sends an electrical header to the central controller to announce the exact number of bytes from the origin node $i$ to the destination node $j$ during a time interval of one frame sent, associated with the optical data frame. The central controller can thus get the direct information of traffic flowing into the AAPN core network, but with the compensation of signaling consumption. In the second method, there is no electrical header information associated with the optical data frame. The central controller does not know the exact number of bytes for each origin-destination pair. Some slots may not be completely filled. By measuring the power of the optical pulses, the central controller can estimate the usage percentage of each time slot, and hence the number of bytes for each origin-destination pair. This way reduces the signaling consumption dramatically, but the estimation of the occupancy of some set of time slots induces errors. In the second approach that the central controller predicts the traffic demands, the central controller can get the information of the traffic load between all origin-destination pairs so that it can allocate the resources fairly and improve the overall network performance. However, the values that the central controller estimates are not the traffic arrivals to the edge nodes, and the central controller has to estimate the traffic flowing into edge node $u_{ij}(k)$ from the measured traffic flowing into the AAPN core network, $d_{ij}(k)$. They are related by a complicated non-linear form, as shown in (5.1) and (5.2).

In any of the cases, the goal is to estimate the traffic arrivals to the origin edge nodes destined to individual destination nodes $\hat{u}_{ij}(t+\tau)$, where $\tau$ is the length of the control period (prediction interval). In future work, a more detailed comparison will be performed to decide whether the traffic prediction shall be performed by edge nodes or by the central controller. Sequential Monte-Carlo methods or extended Kalman filtering can be used, which are described in Chapter 2.3, to track the key state parameters of our proposed model and predict the upper bounds on the traffic demands. A fair resource allocation scheme can then be designed based on the predicted upper bounds.

# References

[1] B. Mukherjee, *Optical Communication Networks.* New York: McGraw–Hill, 1997.

[2] R. Alferness, "The all-optical networks," (Beijing, China), In Proc. Int. Conf. on Communication Technology, Aug. 2000.

[3] G. V. Bochmann, T. Hall, O. Yang, M. J. Coates, L. Mason, and R. Vickers, "The agile all-photonic network: An architectural outline," (Queen's University, Kingston, Ontario, Canada), 22nd Biennial Symposium on Communications, May 2004.

[4] D. Veitch and P. Abry, "A wavelet based joint estimator of the parameters of long-range dependence," *IEEE Trans. Info. Th.*, vol. 45, pp. 878–897, Apr. 1999.

[5] Y. Vardi, "Network tomography: estimating source-destination traffic intensities from link data," *J. Amer. Stat. Assoc.*, vol. 91, pp. 365–377, Mar. 1996.

[6] M. Coates, A. Hero, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Processing Magazine*, vol. 19, pp. 47–65, May 2002.

[7] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, "Internet tomography: recent developments," *Statistical Science (invited)*, Aug. 2005.

[8] R. Caceres, N. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal loss characteristics," *IEEE Trans. Info. Theory*, vol. 45, pp. 2462–2480, Nov. 1999.

[9] M. Coates and R. Nowak, "Network loss inference using unicast end-to-end measurement," (Monterey, CA), In ITC Seminar on IP Traffic, Measurement and Modelling, Sep. 2000.

[10] M. Coates and R. Nowak, "Network delay distribution inference from end-to-end unicast measurement," (Salt Lake City, Utah), In Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc., May 2001.

[11] M. Coates and R. Nowak, "Sequential monte carlo inference of internal delays in nonstationary data networkds," *IEEE Trans. Signal Processing, Special Issue on Monte Carlo Methods for Statistical Signal Processing*, vol. 50, pp. 366–376, Feb. 2002.

[12] M. Coates, R. Castro, and R. Nowak, "Maximum likelihood network topology identification from edge-based unicast measurements," (Marina Del Rey, CA), In Proc. ACM Sigmetrics, Jun. 2002.

[13] C. Tebaldi and M. West, "Bayesian inference on network traffic using link count data (with discussion)," *J. Amer. Stat. Assoc.*, vol. 93, pp. 557–576, Jun. 1998.

[14] J. Cao, D. Davis, S. V. Wiel, and B. Yu, "Time-varying network tomography: router link data," *J. Amer. Stat. Assoc.*, vol. 95, pp. 1063–1075, 2000.

[15] J. Cao, S. V. Wiel, B. Yu, and Z. Z, "A scalable method for estimating network traffic matrices," tech. rep., Bell Labs, 2000.

[16] G. Liang and B. Yu, "Maximum pseudo likelihood estimation in network tomography," (San Francisco, California), In Proc. IEEE Infocom 2003, Apr. 2003.

[17] K. Harfoush, A. Bestavros, and J. Byers, "Robust identification of shared losses using end-to-end unicast probes," (Osaka, Japan), In Proc. of the 8th IEEE International Conference on Network Protocols (ICNP), November 2000.

[18] N. G. Duffield, J. Horowitz, and F. L. Presti, "Adaptive multicast topology inference," (Anchorage, Alaska), In Proc. IEEE Infocom, Apr. 2001.

[19] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions.* Wiley Interscience, 1997.

[20] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice.* New York: Springer-Verlag, 2001.

[21] F. L. Presti, N. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal delay distributions," tech. rep., CMPSCI 99-55, University of Massachusetts, 1999.

[22] S. Ratnasamy and S. McCanne, "Inference of multicast routing trees and bottleneck bandwidths using end-to-end measurements," (New York, NY), pp. 353–360, In Proc. IEEE Infocom, 1999.

[23] R. Cceres, N. Duffield, J. Horowitz, F. L. Presti, and D. Towsley, "Loss-based inference of multicast network topology," (Phoenix, AZ), In Proc. IEEE Conference on Decision and Control, Dec. 1999.

[24] N. Duffield, J. Horowitz, F. L. Presti, and D. Towsley, "Multicast topology inference from measured end-to-end loss," *IEEE Trans. Info. Theory*, vol. 48, pp. 26–45, 2002.

[25] T. Bu, N. Duffield, F. L. Presti, and D. Towsley, "Network tomography on general topologies," (Marina Del Rey, CA), In Proc. ACM Sigmetrics, Jun. 2002.

[26] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," (San Diego, California, USA), In Proc. ACM Sigmetrics, Jun. 2003.

[27] V. S. Frost and B. Melamed, "Traffic modeling for telecommunications networks," *IEEE Communications Magazine*, pp. 70–81, Mar. 1994.

[28] V. Paxson and S. Floyd, "Wide area traffic: the failure of Poisson modeling," *IEEE/ACM Transactions on Networking*, vol. 3, pp. 226–244, 1995.

[29] W. Fischer and K. Meier-Hellstern, "The Markov-modulated poisson process (MMPP) cookbook," *Performance Evaluation*, vol. 18, pp. 149–171, Sep. 1993.

[30] B. Melamed, "An overview of TES processes and modeling methodology," in *Performance/Sigmetrics Tutorials*, pp. 359–393, 1993.

[31] M. Corradi, R. G. Garroppo, S. Giordano, and M. Pagano, "Analysis of f-ARIMA processes in the modelling of broadband traffic," vol. 3, (Helsinki, Finland), pp. 11–14, IEEE Int. Conf. on Communications, Jun. 2001.

[32] W. Willinger, M. Taqqu, and R. S. nd D. V. Wilson, "Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level," *IEEE/ACM Trans. Networking*, vol. 5, pp. 71–86, Feb. 1997.

[33] S. Lowen and M. Teich, "Fractal renewal processes generate 1/f noise," *Phys. Rev. E*, vol. 47, Feb. 1993.

[34] R. Jain and S. A. Routhier, "Packet trains - measurements and a new model for computer network traffic," *IEEE Journal on selected areas in communications*, vol. SAC-4 No.6, Sep. 1986.

[35] W. Willinger, V. Paxson, R. H. Riedi, and M. S. Taqqu, *Long-Range Dependence and Data Network Traffic.* Birkhäuser, 2001.

[36] M. Crovella, M. Taqqu, and A. Bestavros, "Heavy-tailed probability distributions in the world wide web," in *A Practical Guide To Heavy Tails: statistical techniques and applications*, vol. 1, pp. 3–26, 1998.

[37] O. Cappe, E. Moulines, J.-C. Pesquet, A. Petropulu, and X. Yang, "Long-range dependence and heavy-tail modeling for teletraffic data," *IEEE Signal Processing Magazine*, vol. 19, pp. 14–27, May 2002.

[38] K. Park and W. Willinger, *Self-similar network traffic: An overview.* Wiley Interscience, 1999.

[39] B. B. Mandelbrot, "Long-run linearity, locally gaussian processes, h-spectra and infinite variances," *Internation Economic Review 10*, pp. 82–113, 1969.

[40] X. Yang and A. Petropulu, "The extended alternating fractal renewal process for modeling traffic in high-speed communication networks," *IEEE Trans. Signal Processing*, vol. 49-7, pp. 1349–1363, Jul. 2001.

[41] T.Bonald and J.W.Roberts, "Congestion at flow level and the impact of user behaviour," *Computer Networks*, vol. 42, pp. 521–536, 2003.

[42] S. Ma and C. Ji, "Modeling heterogeneous network traffic in wavelet domain," *IEEE/ACM Trans. Networking*, vol. 9, pp. 634–649, Oct. 2001.

[43] V. Ribeiro, R. Riedi, M. Crouse, and R. Baraniuk, "Multiscale Queuing Analysis of Long-Range-Dependent Network Traffic," in *IEEE Infocom*, March 2000.

[44] D. Veitch, J. Backar, J. Wall, J. Yates, and M. Roughan, "On-line generation of fractal and multi-fractal traffic," (Hamilton, New Zealand), In PAM2000 Workshop on Passive and Active Networking, Apr. 2000.

[45] "Models for forecasting international traffic," *CCITT Recommendation E.507.*

[46] "Forecasting international traffic," *CCITT Recommendation E.506.*

[47] A. M. Odlyzko, "Internet traffic growth: Sources and implications," *In Proc. SPIE*, vol. 5247, pp. 1–15, 2003.

[48] V. Paxton and S. Floyd, "Why we don't know how to simulate the Internet," (Atlanta, GA), In Proc. Winter Simulation Conference, Dec. 1997.

[49] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and C. Diot, "Packet-level traffic measurements from the Sprint IP backbone," *Network, IEEE*, vol. 17, pp. 6–16, Nov. 2003.

[50] D.Morato, J.Aracil, L.A.Diez, M.Izal, and E.Magana, "On linear prediction of Internet traffic for packet and burst switching networks," (Scottsdale, Arizona), pp. 138–143, In Proc. ICCCN, Oct. 2001.

[51] A. Sang and S. Li, "A predictability analysis of network traffic," vol. 39, pp. 329–345, 2002.

[52] S. Haykin, *Adaptive Filter Theory.* Upper Saddle River, New Jersey, USA: Prentice Hall, 2002.

[53] R. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, vol. 82, pp. 35-45, 1960.

[54] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings-F*, vol. 140, pp. 107–113, 1993.

[55] J. Cao, W. S. Cleveland, D. Lin, and D. X. Sun, "Internet traffic tends toward Poisson and independent as the load increases," in *Nonlinear Estimation and Classification* (C. C. Holmes, D. D. Denison, M. H. Hansen, B. Yu, and B. Mallick, eds.), New York: Springer, 2002.

[56] J. Cao, W. S. Cleveland, D. Lin, and D. X. Sun, "The effect of statistical multiplexing on the long range dependence of internet packet traffic," tech. rep., Bell Labs, 2002.

[57] J. Cao, W. S. Cleveland, D. Lin, and D. X. Sun, "On the Nonstationarity of Internet Traffic," (Cambridge, Massachusettes), ACM Sigmetrics, 2001.

[58] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido, "A Nonstationary Poisson View of Intenet Traffic," (Hong Kong, China), IEEE Infocom, Mar. 2004.

[59] I. Norros, "On the use of fractal Brownian motion in the theory of connectionless networks," *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 953–962, Aug. 1995.

[60] P. Abry and D. Veitch, "Wavelet analysis of long-range dependent traffic," *IEEE Trans. on Info. Theory*, vol. 44, no. 1, pp. 2–15, 1998.

[61] P. Abry, D. Veitch, and P. Flandrin, "Long-range dependence: revisiting aggregation with wavelets," *Journal of Time Series Analysis*, vol. 19, no. 3, pp. 253–266, 1998.

[62] D. Veitch and P. Abry, "A statistical test for the time constancy of scaling exponents," *Preprint.*