A resequencing model for high-speed packet-switching networks

Ka-Cheong Leung *, Victor O.K. Li

Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, China

A R T I C L E   I N F O

Article history:
Received 21 February 2008
Received in revised form 8 January 2009
Accepted 19 October 2009
Available online 24 October 2009

Keywords:
Dispersity routing
Multipath routing
Parallel queues
Performance modelling
Resequencing

A B S T R A C T

In this paper, we propose a framework to study the resequencing mechanism in high-speed networks. This framework allows us to estimate the packet resequencing delay and the resequencing buffer occupancy distributions when data traffic is dispersed on multiple disjoint paths. In contrast to most existing work, the end-to-end path delay distribution is decoupled from the resequencing model. Thus, once the end-to-end path delay distribution is obtained, such as from historical data, our model may be used. In this paper, we illustrate our proposed model with Gaussian distributed path delays. Our results show that the packet resequencing delay and the resequencing buffer occupancy drop when the traffic is spread over a larger number of homogeneous paths, although the network performance improvement quickly saturates when the number of paths increases. We find that the number of paths used in multipath routing should be small, say, up to three.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

The popular routing approach for the Internet consists of finding a single shortest path from a source to a destination based on some link cost metrics, which are updated periodically. Although such unipath routing protocols can adapt very quickly to changing network conditions, they become unstable under heavy loads as the link cost metrics used in the routing algorithms are related to delays or congestion experienced over the network links [22]. Moreover, at a given time instant, some subnets can be heavily congested, while other subnets along alternate paths are under-utilized. Thus, traffic engineering should be employed to balance the use of network resources such as link bandwidths. This advocates the use of multiple paths simultaneously for data transmission.

Multipath routing [3,7,15,16,18] is a load balancing technique to spread the traffic load across the network in order to alleviate network congestion. It utilizes a set of active paths for transmitting packets from a source to a destination. It has been shown [12,20] that multipath routing balances the load significantly better than single-path routing and provides better performance in alleviating congestion in wired/wireless networks. Packet-level multipath routing allows packets of the same traffic flow to be forwarded over different routes to a destination so as to achieve load balancing in packet-switching networks. However, these packets may be reordered on arrival at the destination due to the differences in path delays. Those packets arriving out of order may have to be stored in a buffer until they can be delivered to the end process in the proper order. This is called resequencing, and the buffer is called the resequencing buffer. Recent studies [1,5,19] show that packet reordering is not a rare event. To provide better performance, the need for resequencing should be minimized [2]. To see whether and when multipath routing can have a better performance than unipath routing, models are required to characterize the resequencing mechanism.

The existing work can be grouped into two major categories. The first category consists of work that characterizes the disordering network as a queuing system with several servers sharing a single queue [4,24]. In [4], distributions of resequencing delay and total delay have been evaluated for three different queuing models with homogeneous servers, namely, the $G/M/m$ model, the $G/M/\infty$ model, and the $M/H_\infty$ model. The queue length distribution can be computed as in [10]. The resequencing delay distribution is then calculated by conditioning on the number of other customers (or packets) being served when a tagged customer goes into service. The tagged customer is an arbitrary customer whose resequencing delay distribution is computed.

Besides, an analysis on packet resequencing for reliable network protocol has been studied in [24]. Packet disordering was modelled by including an independently and identically distributed random propagation delay to each packet. The packet resequencing delay and buffer size distributions were analyzed for data transmission over an end-to-end reliable transport protocol. Packet retransmission

---

* Corresponding author. Tel.: +852 2857 8481; fax: +852 2559 8738.
E-mail addresses: kcleung@ieee.org (K.-C. Leung), vli@eee.hku.hk (V.O.K. Li).
was handled by the following automatic repeat request (ARQ) protocols: the stop-and-wait ARQ or the selective-repeat ARQ.

In the second category of work, the disordering network is described as a queueing system with several parallel servers and queues, and each server has its own dedicated queue [6,9,11,13,21,25]. In this case, any orderly dispersion sequences can be easily incorporated into resequencing models. In [6], two parallel paths have been modelled as two heterogeneous M/M/1 queues. The mean resequencing delay of a tagged customer was derived by conditioning on the number of customers being passed (i.e. they have arrived earlier but completed their services later) in the other queue when the tagged customer has just finished its service. Besides, different fixed delays on the two paths is considered to illustrate the effect of fixed delays on the optimal splitting probability. The probability for large deviations of the resequencing buffer occupancy with two parallel paths modelled by two heterogeneous M/M/1 queues was derived in [25]. The idea is to compute the probability distribution on the number of customers being passed (i.e. they have arrived later but completed their services earlier) in the other queue when the oldest customer has just completed its service.

In [9], the resequencing delay distribution was derived from a disordering network with K heterogeneous M/G/1 queues. In [13], the exact asymptotics of the resequencing delay and the resequencing buffer occupancy distributions for two heterogeneous G/G/1 queues were obtained under heavy-tailed path delay assumptions. The tail behaviour of the resequencing delay distribution for K heterogeneous M/G/1 queues under both light-tailed and heavy-tailed path delay assumptions was studied in [8].

Nevertheless, all the aforementioned models assume probabilistic or Bernoulli routing, whereas our proposed framework assumes that packets are delivered on deterministic paths.

The effect of deterministic routing on queueing performance has also been considered in [11,21]. In [11], two parallel paths are modelled as two heterogeneous H2/M/1 queues. Packets are switched from a source in a round-robin manner between these two paths. A multipath routing system with K identical server queues was studied in [21]. The optimality of the round-robin routing mechanism is validated in heavy traffic, the round-robin routing policy is shown to yield the smallest end-to-end delay among all state-independent routing policies. The same conclusion holds for light traffic with Poisson arrivals. However, our proposed framework is more general as it models the resequencing behaviour of a network with a general and independent inter-arrival time distribution under any orderly dispersion of traffic on any number of paths.

1.1. Our contributions

The focus of this work, first described in [14], is to develop a framework to study the resequencing mechanism in high-speed networks. This framework allows us to estimate the packet resequencing delay and the resequencing buffer occupancy distributions under an orderly dispersion of traffic on multiple disjoint paths.

Most existing work for the resequencing delay analysis assumed Markovian arrival processes, and the queueing analysis becomes very difficult or computationally infeasible when a path consists of multiple hops. Our work distinguishes itself from the previous work by decoupling the estimation of the end-to-end path delay distributions from the calculation of resequencing delay distributions. That is, the packet resequencing delay and the resequencing buffer occupancy distributions can be evaluated by plugging a given end-to-end delay distribution for each path into our proposed framework. Each end-to-end path delay is modelled as a generally distributed random variable and independent of each other. By employing the “divide-and-conquer” approach, the computation of the end-to-end path delay distribution can be delegated to any specialized mathematical or experimental model, and the task for computing the resequencing performance metrics can thus be simplified significantly. Our proposed framework belongs to the second category of work we have mentioned.

Furthermore, as far as we know, there is no analytical model, except [13,25] which assume probabilistic routing on two paths, available to estimate the resequencing buffer occupancy distribution for networks supporting multipath routing on any number of paths. Our proposed model attempts to fill this information gap. The distribution is useful for network administrators who must estimate the size of the resequencing buffer so as to satisfy the quality of service constraints.

Here, the words “source” and “destination” are quite general. A source may mean a source host, which generates traffic. It may also mean a source router, from which network traffic departs. A destination can be defined similarly. It may mean a destination host, which absorbs traffic. It may also mean a destination router, to which network traffic arrives. Though our proposed techniques are end-to-end based, they can be applied to perform traffic engineering under various scenarios, ranging from inter-router to inter-host traffic.

We will investigate the effectiveness of multipath routing by examining three basic questions:

- Does multipath routing improve the system performance? If so, when?
- What is the optimal split of traffic to achieve the best performance?
- What is the cost of employing multipath routing?

1.2. Organization of the paper

This paper is organized as follows. Section 2 gives a traffic model for the proposed framework. Section 3 presents an analytical model to compute the packet resequencing delay, the packet total delay, and the buffer occupancy distributions. Section 4 examines the analytical and simulation results derived from the framework and studies the effectiveness of multipath routing. Section 5 concludes and discusses some possible extensions to our work.

2. Traffic model

Our traffic model consists of a disordering network connecting a source to a destination, and flows of packets. The disordering network consists of a set of N disjoint paths, namely, Path 1, Path 2, ..., Path N, connecting the source to the destination, such that customers may arrive at the destination in a different order as they are sent. We assume that packets are delivered to paths according to a deterministic schedule. Let \( \#(\cdot) \) be the order in which packets are placed onto the paths. \( \#(m) \) denotes the particular path for the mth packet. The packets are numbered and routed according to the order of arrivals. If \( \#(m) = j \), the mth packet will be routed on Path j. Denote by \( s_{jm} \), \( d_{jm} \), and \( q_{jm} \) respectively the sending time of the first bit of the mth packet from the source, the packet end-to-end path delay (also called sojourn time), and the arrival time of the last bit of the packet at the destination, which is delivered on Path j. The relationship among these three quantities can be formulated as:

\[ q_{jm} = s_{jm} + d_{jm}. \]  

Thus \( d_{jm} \) includes the propagation, queueing, and transmission delay of the mth packet on Path j.
Given any two packets $m$ and $n$, they arrive out of order only when Packet $m$ is sent before Packet $n$ at the source but arrives later at the destination. This means that $s_{k(m),m} < s_{k(n),n}$ and $d_{k(m),m,n} > d_{k(n),n}$, The condition can be simplified as:

$$d_{k(m),m} - d_{k(n),n} - l_{k(m),m} > 0,$$

where $l_{k(m),m}$ is the inter-departure time between packets $m$ and $n$ at the source. Because of the finite packet size and transmission bandwidth (or data rate), $s_{k(m),m} > s_{k(n),n}$, where $l_{m}$ is the length of Packet $m$ in bits and $c_{j}$ is the bandwidth of Path $j$ in bits per time unit.

Besides, these two packets may arrive out of order only when they are sent on two different paths, i.e., $\not\equiv_\theta(m) \not= \not\equiv_\theta(n)$. This is true by the in-order channel assumption, which states that if an arbitrary Packet $m$ is transmitted from the source before another Packet $n$ on the same path connecting the source to the destination, Packet $m$ will arrive at the destination before Packet $n$.

3. Resequencing model

Fig. 1 illustrates the resequencing model. It consists of $N$ queues in parallel where each Queue $j, 1 \leq j \leq N$, which corresponds to a transmission path, has a sojourn time $D_{j}$. The sojourn times are assumed to be mutually independent. The service discipline at each queue is first come first served (FCFS) so that the in-order channel assumption is preserved. Customers or packets arrive into the system with a general, independent, and identically distributed inter-arrival time $\lambda$. Each customer is routed deterministically according to the routing sequence function $\theta(\cdot)$. The routing weight of Queue $i$, $p_{i}$, is defined as the portion of dispersed traffic to be routed to Queue $i$, where $\sum_{i=1}^{N} p_{i} = 1$. Upon service completion, these customers join the resequencing buffer to await the arrivals of all those customers that have entered the system before them. The total delay of Customer $i$ in the system is defined as the sum of its sojourn time and resequencing delay. That is,

$$t_{\theta(i)} = d_{\theta(i)} + r_{\theta(i)},$$

(3)

The discussion will proceed as follows. Section 3.1 discusses how to determine the resequencing delay and total delay distributions. An estimation of buffer occupancy distribution is presented in Section 3.2.

3.1. Determination of resequencing delay and total delay distributions

The in-order channel assumption guarantees that the last packet transmitted before an arbitrary packet on a path other than the one taken by the arbitrary packet arrives at the destination last among all packets transmitted before this last packet on that path. The resequencing delay of the arbitrary packet is determined by the maximum time it waits for a packet that has been transmitted before it and received after it. Thus, it is sufficient to consider the last packet to be transmitted before it on every path except for the one taken by it.

Define the random variables $D_{i}, R_{i},$ and $T_{i}$ as the end-to-end path delay, the resequencing delay, and the total delay incurred for a packet to be transmitted on Path $i$. Thus,

$$T_{i} = D_{i} + R_{i},$$

(4)

The cumulative distribution functions of the resequencing delay and the total delay for a packet, namely, $F_{T_{i}}(\cdot)$ and $F_{T}(\cdot)$, respectively, can be deduced by following these steps:

(i) Derive $F_{R_{i}}(\cdot)$, the probability density function of the inter-arrival time to the system between a packet transmitted on Path $i$ and another packet transmitted on Path $j$, $j \neq i$, given that a tagged packet is on Path $k$, where $k$ can either be $i$ or $j$. The tagged packet is an arbitrary packet whose statistical parameters are calculated.

(ii) Deduce the cumulative distribution function of the resequencing delay for a packet, $F_{R_{i}}(\cdot)$.

(iii) Determine the cumulative distribution function of the total delay for a packet, $F_{T}(\cdot)$.

3.1.1. Evaluation of $F_{R_{i}}(t)$

Consider a flow of $\eta$ packets, namely, Packet 1, Packet 2, ..., Packet $\eta$ to be sent from a source to a destination via a disordered network with $N$ disjoint paths. Denote by $\not\equiv_\theta(m)$ and $\equiv_\theta(m)$, respectively the next packet and the previous packet to be routed on Path $j$ with respect to Packet $n$. These two quantities can be expressed as:

$$\not\equiv_\theta(m) = \min \{ n | n > m \land \not\equiv_\theta(n) = j \};$$

$$\equiv_\theta(m) = \max \{ n | n < m \land \equiv_\theta(n) = j \}.$$  

(5)

Let the random variable $R_{i,j,k}$ be the difference in the packet numbers between a packet transmitted on Path $i$ and another packet transmitted on Path $j$, $j \neq i$, given that the tagged packet is on Path $k$, where $k$ can either be $i$ or $j$. The probability mass function of $R_{i,j,k}$ can be computed as:

$$P_{R_{i,j,k}}(d) = \begin{cases} \lim_{\eta \to \infty} \frac{\sum_{m=1}^{\infty} \sum_{n=1}^{\infty} 1(\not\equiv_\theta(m) = i) \not\equiv_\theta(n) = j - m + d}{\sum_{m=1}^{\infty} \sum_{n=1}^{\infty} 1(\not\equiv_\theta(m) = i) \not\equiv_\theta(n) = j} & \text{if } k = i; \\ \lim_{\eta \to \infty} \frac{\sum_{m=1}^{\infty} \sum_{n=1}^{\infty} 1(\equiv_\theta(m) = i) \not\equiv_\theta(n) = j - m + d}{\sum_{m=1}^{\infty} \sum_{n=1}^{\infty} 1(\equiv_\theta(m) = i) \not\equiv_\theta(n) = j} & \text{if } k = j; \end{cases}$$

(7)

where $d$ is any positive integer, and $1(\cdot)$ is an indicator function which returns one when the condition is true, and zero otherwise.

Define another random variable $R_{i,k}$ as the inter-arrival time to the system between a packet transmitted on Path $i$ and another packet transmitted on Path $j$, $j \neq i$, given that the tagged packet is on Path $k$, where $k$ can either be $i$ or $j$. The probability density function of $R_{i,k}$ can be evaluated as:

$$f_{R_{i,k}}(t) = \sum_{d=1}^{\infty} P_{R_{i,k}}(d) \cdot f_{R_{i,k}}(d)(t) = \sum_{d=1}^{\infty} P_{R_{i,k}}(d) \cdot f^{d}(t),$$

(8)

where $f^{d}(t)$ denotes the $d$-folded convolution of $f(t)$.

The computation of the probability density function of $R_{i,k}$ can be illustrated with the use of a two-path example. The time diagram of the packet arrival process for the example is shown in Fig. 2. Consider the periodic routing sequence such that two consecutive packets are transmitted on Path 1, followed by a packet transmitted on Path 2, and so on. $P_{R_{i,k}}(\cdot)$ can be calculated as:

$$P_{R_{1,1}}(d) = P_{R_{2,1}}(d) = \begin{cases} 0.5 & \text{if } d = 1, 2; \\ 0 & \text{otherwise.} \end{cases}$$

(9)

$$P_{R_{1,2}}(d) = P_{R_{2,2}}(d) = \begin{cases} 1 & \text{if } d = 1; \\ 0 & \text{otherwise.} \end{cases}$$

(10)

If the inter-arrival time between two successive packets to the system is exactly one time unit, this implies that:

$$f(t) = \delta(t - 1),$$

(11)

and hence

$$f_{R_{i,k}}(t) = \begin{cases} 0.5 \cdot [\delta(t - 1) + \delta(t - 2)] & \text{if } k = 1; \\ \delta(t - 1) & \text{if } k = 2; \end{cases}$$

(12)

where $(i,j) = (1,2)$ or $(2,1)$, and $\delta(t)$ is the Dirac delta function.
3.1.2. Evaluation of $F_R(t)$

Let $R_i$ be a random variable denoting the waiting time of a packet transmitted on Path $i$, given that it has to wait until a packet transmitted on Path $j$ arrives. $R_i$ is the maximum $R_{ij}, i = 1, 2, \ldots, i - 1, i + 1, \ldots, N$. Fig. 3 illustrates how $R_i$ is computed. The cumulative distribution function of $R_i$ can be estimated as:

$$
F_R(t) = \text{Prob}(R_i \leq t) = \text{Prob}(\max(R_{i1}, R_{i2}, \ldots, R_{in}) \leq t)
$$

$$
= \prod_{j=1}^{n} \text{Prob}(R_{ij} \leq t) = \prod_{j=1}^{n} \text{Prob}([D_j - D_t - I_{ij}^-] \leq t)
$$

$$
= \prod_{j=1}^{n} \int_{-\infty}^{t} f_{D_j - D_t - I_{ij}^-} d\phi \cdot u(t)
$$

$$
= \prod_{j=1}^{n} \int_{-\infty}^{t} \int_{-\infty}^{\xi} f_{D_j - D_t - I_{ij}^-}(\xi, \eta) d\eta d\xi \cdot u(t)
$$

$$
= \prod_{j=1}^{n} \int_{-\infty}^{t} f_{I_{ij}}(\xi) \cdot \int_{-\infty}^{\xi} f_{D_j - D_t - I_{ij}^-}(\eta|\xi) d\eta d\xi \cdot u(t)
$$

$$
= \prod_{j=1}^{n} \int_{-\infty}^{t} f_{I_{ij}}(\xi) \cdot \int_{-\infty}^{\xi} f_{D_j - D_t}(\eta) d\eta - f_{D_j - D_t}(-\eta|\xi) d\eta \cdot d\xi \cdot u(t),
$$

where $f_X(t)$ is the density function of a random variable $X$, $f_{X|Y}(s, t)$ is the joint density function of random variables $X$ and $Y$, and $u(t)$ is the unit step function.

By the law of total probability [23], the cumulative distribution function of the packet resequencing delay is given by:

$$
F_\text{re}(t) = \sum_{i=1}^{N} p_i F_R(t).
$$

3.1.3. Evaluation of $F_T(t)$

Given that $D_i$ equals $\theta$, the conditional cumulative distribution function of $R_i$ can be computed as:

$$
F_{R_i|D_i}(t|\theta) = \prod_{j=1, j\neq i}^{N} \int_{-\infty}^{t} f_{D_j - D_t - I_{ij}^-}(\phi|\theta) d\phi \cdot u(t)
$$

Since the total delay of a packet is the sum of its end-to-end path delay and resequencing delay, the cumulative distribution function of the total delay for a packet can be calculated as:

$$
F_T(t) = \sum_{i=1}^{N} p_i F_R(t)
$$

$$
= \sum_{i=1}^{N} p_i \left[ \int_{-\infty}^{t} f_{\theta}(\phi) d\phi \right]
$$

$$
= \sum_{i=1}^{N} p_i \left[ \int_{-\infty}^{t} f_{D_i - D_j}(\phi) d\phi \right]
$$

$$
= \sum_{i=1}^{N} p_i \left[ \int_{-\infty}^{t} \int_{-\infty}^{\xi} f_{D_j - D_t - I_{ij}^-}(\xi, \eta) d\eta d\xi \right]
$$

$$
= \sum_{i=1}^{N} p_i \left[ \int_{-\infty}^{t} \int_{-\infty}^{\xi} f_{D_j - D_t}(\eta|\xi) d\eta d\xi \right]
$$

$$
= \sum_{i=1}^{N} p_i \left[ \int_{-\infty}^{t} f_{D_j}(\xi) \cdot \int_{-\infty}^{\xi} f_{D_j - D_t}(\eta|\xi) d\eta \cdot d\xi \right].
$$

Fig. 1. The resequencing model.

Fig. 2. The time diagram for two-path example.
3.2. Estimation of buffer occupancy distribution

It has been shown [21] that, when customers are routed deterministically in a cyclical fashion, the end-to-end delay can be minimized. To achieve the best system performance, we assume that weighted round-robin (WRR), also known as generalized round-robin (GRR), is employed. WRR distributes packets to each path such that the number of packets allocated to each path relative to the sum on all paths is as close to its routing weight as possible.

The resequencing buffer occupancy seen by an arbitrary packet is the total number of packets, each of which has arrived at the destination before the arbitrary packet and is waiting for the arrival of a packet (which may or may not be the arbitrary packet) which has been transmitted before it. A packet arriving at the destination ahead of the arrival of any packet which has been transmitted before it has to be stored in the resequencing buffer. At any time instant, the total number of such “early arrived” packets gives the current resequencing buffer occupancy.

This suggests that the resequencing buffer occupancy seen by an arbitrary packet can be found by counting the total number of packets that arrive at the destination before the arbitrary packet but are waiting for the arrivals of some packets that have been transmitted before them. Fig. 4 illustrates how the resequencing buffer occupancy is estimated when a tagged packet arrives. In

![Diagram](image)

**Fig. 3.** A diagram showing how the resequencing delay is computed.

![Diagram](image)

**Fig. 4.** A diagram showing how the resequencing buffer occupancy is estimated.
the figure, all stored packets, which have been transmitted on the same path, are grouped on the same row. Similarly, all stored packets transmitted in the same round are grouped on the same column. A round is defined such that a series of consecutive packets is sent on a set of paths with increasing path identification numbers. Choosing any two packets from a row, a packet on the left has a smaller sequence number and has been sent earlier by the source than the one on the right. Similarly, choosing any two packets from a column, a packet on the top has a smaller sequence number than the one on the bottom.

According to the figure, all stored packets can be divided into two major categories, with reference to an arbitrary tagged packet transmitted on Path $i$. The first group of stored packets is those packets (denoted as shaded circles) that are transmitted after the tagged packet. The second group of stored packets are those packets (denoted as unfilled circles) that are sent before the tagged packet. All stored packets have to wait in the resequencing buffer until a packet transmitted on Path $j$ arrives. We call Path $j$ the bottleneck path of the tagged packet, transmitted on Path $i$. If such a bottleneck path does not exist, this means that all stored packets have larger sequence numbers than that of the tagged packet and thus $j = i$. Therefore, there is at most one bottleneck path seen by any tagged packet.

Based on the above observation, the resequencing buffer occupancy distribution, $P_{A}(\cdot)$, can be derived by following these steps:

(i) Derive the probability distribution of the resequencing buffer occupancy on the condition that the waiting packets have been transmitted on Path $q$ after the tagged packet transmitted on Path $i$, $P_{A}(\cdot)$. $\mathbb{P}(\cdot)$.

(ii) Deduce the probability distribution of the resequencing buffer occupancy on the condition that the waiting packets have been transmitted on any path before the tagged packet and the bottleneck path of the tagged packet is Path $j$, $P_{A}(\cdot)$. $\mathbb{P}(\cdot)$.

(iii) Determine the unconditional resequencing buffer occupancy probability distribution, $P_{A}(\cdot)$. $\mathbb{P}(\cdot)$.

3.2.1. Evaluation of $P_{A}(\cdot)$

Suppose the random variable $G_i$ is the inter-arrival time of two consecutive packets transmitted on Path $q$. Let the random variable $G_q$ be the sum of $k$ independent, identically distributed random variables, each of which is statistically identical to the random variable $G_i$. If $A_i(q, i) \leq k$, this means that the $(k + 1)$th packet transmitted on Path $q$ after an arbitrary tagged packet transmitted on Path $i$ arrive at the destination after it. This means that $D_i - D_q - I_{iij} - C_q < 0$. Thus, the probability that $A_i(q, i) \leq k$ is given by:

$$U_{A_i(q, i)}(k) = \mathbb{P}(A_i(q, i) \leq k) = \mathbb{P}(D_i - D_q - I_{iij} - C_q < 0)$$

$$= \int_{-\infty}^{0} f_{D_i - D_q - I_{iij} - C_q} (\varphi) \, d\varphi. \tag{17}$$

The probability mass function of the resequencing buffer occupancy on the condition that those waiting packets have been transmitted on Path $q$ after the tagged packet can be written as:

$$P_{A_i(q, i)}(k) = \begin{cases} U_{A_i(q, i)}(0) & \text{if } q \neq i, k = 0; \\ U_{A_i(q, i)}(k) - U_{A_i(q, i)}(k - 1) & \text{if } q \neq i, k = 1, 2, \ldots; \\ 1 & \text{if } q = i, k = 0; \\ 0 & \text{otherwise}. \end{cases} \tag{18}$$

3.2.2. Evaluation of $P_{A}(\cdot)$

The evaluation of $P_{A}(\cdot)$ is outlined as follows. It first computes the distribution of the number of packets transmitted on Path $q$ earlier than the tagged packet and which arrive at the destination after it, $P_{A}(\cdot)$. $\mathbb{P}(\cdot)$. That is, it counts the number of “missing” packets which are sent before the tagged packet and have not arrived at the destination yet. The conditional probability mass function of $M(q, i)$ is determined. Given the number of packets transmitted on Path $i$ waiting in the resequencing buffer, the maximum number of packets sent earlier than the tagged packet waiting in the resequencing buffer is found. The conditional probability mass function of the number of awaiting packets transmitted on any path before the tagged packet, $P_{A}(\cdot)$, is estimated.

Let random variable $M(q, i)$ denote the number of those packets transmitted on Path $q$ earlier than an arbitrary tagged packet transmitted on Path $i$ and which arrive at the destination after it. The probability mass function of $M(q, i)$ is given by:

$$P_{M(q, i)}(k) = \begin{cases} V_{M(q, i)}(0) & \text{if } q \neq i, k = 0; \\ V_{M(q, i)}(k) - V_{M(q, i)}(k - 1) & \text{if } q \neq i, k = 1, 2, \ldots; \\ 1 & \text{if } q = i, k = 0; \\ 0 & \text{otherwise}, \end{cases} \tag{19}$$

where $V_{M(q, i)}(k) = \mathbb{P}(M(q, i) \leq k) = \mathbb{P}(D_q - D_i - I_{iij} - C_q < 0)$. Given that $M(q, i) \leq u$, the conditional probability mass function of $M(q, i)$ can be determined as:

$$P_{M(q, i)|u}(k) = \frac{P_{M(q, i)}(k)}{\sum_{k=0}^{u} P_{M(q, i)}(k)}. \tag{20}$$

If the tagged packet sees $n$ packets transmitted on Path $i$ waiting in the resequencing buffer, it can be inferred that the maximum number of packets transmitted on Path $q$ before the tagged packet waiting in the buffer is:

$$w(n|q, i) = \left\lceil \frac{(n + 1) \cdot p_i}{p_q} \right\rceil. \tag{21}$$

If the tagged packet sees $n$ packets transmitted on Path $i$ waiting in the resequencing buffer and the bottleneck path of the tagged packet is Path $j$, $j \neq i$, the maximum number of packets transmitted before the tagged packet waiting in the buffer can be calculated as:

$$Y(j|i) = \sum_{q=1}^{N} w(n|q, i) + X_{ij}(I_{ij}) - \frac{I_{ij}}{Z} - 1, \tag{22}$$

where $G_i > I_{ij}, X_{ij}(I_{ij})$ denotes the number of packets transmitted on Path $j$ during the time period $I_{ij}$, given that a packet is transmitted on Path $i$ at the beginning of the time period and a packet is transmitted on Path $j$ at the end of the time period, and $Z$ is the random variable denoting the inter-arrival time between any two successive packets to the system. The first term of the equation corresponds to the maximum number of packets transmitted on any paths except Path $j$. The second and third terms correspond to the adjustment terms to count the number of packets transmitted during the period $I_{ij}$. These packets are sent before the packet with the smallest sequence number transmitted on Path $j$ being awaited, and hence they would not be in the resequencing buffer. The final term is to discount the tagged packet itself.

The number of awaiting packets transmitted on any path before the tagged packet transmitted on Path $i$ and the bottleneck path of the tagged packet is Path $j$ can be found as:
\[ A_h(j|i) = Y(j|i) - \sum_{q=1}^{N} M(q|i). \]  

(23)

Given that the tagged packet sees \( n \) packets transmitted on Path \( i \) waiting in the resequencing buffer, the conditional probability mass function of \( A_h(j|i) \) can be written as:

\[ P_{A_h}(k(n)|n) = \sum_{q=1}^{N} z_{q,j} P_{M(q|i)}(-k) w(n-1|q,i) \]

\[ + z_{q,j} P_{M(q|i)}(-k) w(n,q,i)), \]

(24)

where \( z_{q,j} = \text{Prob}(k|q > j | l,j) | k, q = j, l \).

Denote \( P_m(j|i) \) as the probability that Path \( j \) is the bottleneck path of the tagged packet transmitted on Path \( i \). Noting that there is at most one bottleneck path seen by any tagged packet, the probability can be computed as:

\[
P_m(j|i) = \begin{cases} 
\text{Prob}(D_j - D_i - I_{ij} > 0) \cdot \left[ 1 - \sum_{q=1}^{N} z_{q,j} P_m(q|i) \right] & \text{if } j \neq i; \\
1 - \sum_{q=1}^{N} P_m(q|i) & \text{otherwise.}
\end{cases}
\]

(25)

By considering the probability that the tagged packet sees at most \( n \) packets sent on Path \( i \) waiting in the resequencing buffer, the probability that the tagged packet sees \( n \) packets transmitted on Path \( i \) waiting in the resequencing buffer can be found as:

\[
P_r(n|i) = \prod_{q=1}^{n} \left\{ \sum_{k=0}^{N} P_{M(q|i)}(k) \left[ w(n,q,i) - \left( n+1 \right) \cdot P_k \right] / P_i - P_m(q|i) \right\}.
\]

(26)

Finally, the probability mass function of the number of awaiting packets transmitted on any path before the tagged packet and given that the bottleneck path is Path \( j, j \neq i \), can be approximated as:

\[ P_{A_{ij}}(k) \approx \sum_{n=0}^{\infty} P_r(n|i) P_{A_{ij}}(k(n))/1 - P_m(l|i). \]

(27)

Since we have used the average value of \( P_r(n|i) \), instead of the corresponding value, conditioned on the bottleneck path being Path \( j \).

Simulation experiments have shown that our analytical results closely match with the simulation results within an acceptable order of accuracy (refer to Figs. 8 and 10 for details).

3.2.3. Evaluation of \( P_r(k) \)

Denote random variable \( A(j|i) \) as the total number of packets waiting in the resequencing buffer and seen by an arrival of a packet from Path \( i \) and the bottleneck path is Path \( j \). By using the results in Sections 3.2.1 and 3.2.2, the probability mass function of \( A(j|i) \) can be calculated as:

\[
P_{A_{ij}}(k) = \begin{cases} 
N \sum_{h=1}^{N} P_{A_{ij}}(k) & \text{if } j = i; \\
N \sum_{h \neq i} P_{A_{ij}}(k) & \text{otherwise.}
\end{cases}
\]

(28)

By applying the law of total probability [23], the probability mass function of the resequencing buffer occupancy seen by a packet from Path \( i \) can be written as:

\[ P_h(k) = \sum_{j=1}^{N} P_r(n|i) P_{A_{ij}}(k). \]

(29)

Further application of the law of total probability results in the resequencing buffer occupancy probability mass function:

\[ P_h(k) = \sum_{i=1}^{N} p_i P_h(k), \]

(30)

where \( k \) is any non-negative integer, and \( \sum_{k=1}^{\infty} p_k = 1 \).

The packet loss probability can also be estimated as follows. Let \( \mathscr{B} \) be the size of the resequencing buffer, which is in terms of packets. Therefore, the loss probability of any packet from Path \( i \) can be approximated as:

\[ P_l \approx \text{Prob}(B_i \geq \mathscr{B}) = 1 - \sum_{k=0}^{\mathscr{B}} P_h(k), \]

(31)

since we have neglected the effect on the queue due to lost packets because of overflow at the resequencing buffer.

Using the law of total probability, the packet loss probability can be computed as:

\[ P_l = \sum_{i=1}^{N} p_i P_l \approx 1 - \sum_{k=0}^{\mathscr{B}} p_i P_h(k). \]

(32)

4. Performance evaluation

This section discusses the numerical results based on the analytical expressions obtained in Section 3. With the help of some numerical examples, we can illustrate the effectiveness of multipath routing by answering the three basic questions posed in Section 1.

The expressions we have obtained are applicable to general and independent path delay distributions. To illustrate our multipath routing scheme, and without loss of generality, the following tandem queue path delay model is used.

A path is modelled as an \( L \)-node G/M/1 tandem network, as exhibited in Fig. 5. L G/M/1 queues are connected in tandem. The ith queue receives input from two traffic sources: the deterministically distributed tagged dispered traffic of rate \( \lambda_i \) and the exponentially distributed interfering or background traffic of rate \( b_i \). The service rate of the ith server is \( \mu_i \). L is set to 5 for our subsequent studies. As there is no closed form solution for the mean \( \mathcal{D} \) and the variance \( \sigma_2 \) of this type of tandem network, simulation experiments have been carried out to find these required statistics, i.e. \( \mathcal{D} \) and \( \sigma_2 \). Since \( \mathcal{D} \) and \( \sigma_2 \) are obtained by simulation, there is no analytical results reported for these statistics in the studies.

The central limit theorem [23] suggests that the end-to-end path delay, which is the sum of a sufficiently large number of hop delays, is approximately normally distributed. The mean and the variance of the end-to-end path delay provide sufficient information to generate an approximate distribution, which can then be utilized to compute the resequencing delay distribution. This approach, which was used to solve the end-to-end percentile-type delay objective allocation problem for networks supporting Switched Multi-megabit Data Service (SMDS), has been shown to provide the best approximation to the reference values [17]. Thus, the end-to-end path delay used in the analytical studies is assumed to be Gaussian or normally distributed with mean \( \mathcal{D} \) and variance \( \sigma_2 \). The resulting end-to-end path delay distribution is then plugged into our resequencing model to compute the packet
resequencing delay and the resequencing buffer occupancy distributions. Despite some approximations taken in the resequencing model and the aforementioned end-to-end path delay distribution, our analytical results closely match with the simulation results within an acceptable order of accuracy (refer to Figs. 6, 7, and 10 for details).

An event-driven simulator (written in C programming language) is employed to emulate the resequencing behaviour of a network with a set of disjoint paths. Simulation experiments have also been carried out to serve two purposes. First, they can help to verify the correctness of our proposed resequencing model. We find that our analytical results closely match the simulation results in general, except that there exists some differences in distributions where the probability function is in the order of $10^{-3}$ or less. Second, they can provide the simulated mean and variance of the end-to-end path delay to the path model mentioned above, that can be plugged into our proposed analytical resequencing model for computing the resequencing performance metrics.

The inter-sending time between any two consecutive packets from the source, denoted as the inter-packet time, is a constant. This resembles the scenario when the source transmits, say, streaming audio traffic to the destination. Each packet is routed to the paths based on weighted round-robin (WRR) routing discipline so that packets are spread as uniformly as possible on every path when they are injected into the network. A path is modelled as a five-node $G/M/1$ tandem network. Each server serves a packet with an average service time of one time unit. Its service discipline is first come first served (FCFS). Its end-to-end path delay is assumed to be Gaussian or normally distributed with its mean and variance computed from the path model aforementioned.

For each simulation run, the end-to-end path delay and the resequencing statistics are collected over $10^8$ packets of tagged
dispersed traffic, excluding the statistics for the first $10^5$ packets of tagged dispersed traffic. Since the packet generation process involves the use of random numbers, a total of ten runs have been done to find the queueing statistics, and a 95% confidence interval for the average value of each statistical metric is also computed for reference. We have plotted the 95% confidence intervals in the figures, but since they have end points which were very close to the corresponding mean values, in most cases, they were indistinguishable from the mean values.

The results are provided in two sets. The first set studies the effectiveness of multipath routing with different number of homogeneous paths used, where the routing weight to each path is the same. It provides the mean total delays, the mean path delays, the mean resequencing delays, the mean resequencing buffer occupancies, and the bounds on the complementary functions for the resequencing buffer occupancies, for each of the 40 cases (two background loads, two inter-packet times, and 10 path configurations). It also provides the resequencing buffer distributions for three cases (three path configurations). The second set studies the effectiveness of multipath routing using two heterogeneous paths, with possibly different routing weights according to a given dispersion ratio. The study includes the mean total delays, the mean path delays, the mean resequencing delays, the mean resequencing buffer occupancies, and the bounds on the complementary functions for the resequencing buffer occupancies, for each of the eight cases (eight dispersion ratios).

We examine the first set of results. Fig. 6 shows the mean total delays and the mean path delays. Two different background loads, namely link utilizations of 0.5 and 0.75, and two different inter-packet times, namely five and 10 time units, are chosen. The mean total delay and the mean path delay drop as the number of paths used increases. The performance improvement is more significant when the background load increases or the inter-packet time for tagged traffic decreases. Clearly, the decrease in the mean path delay comes from the decrease in utilization of each path, since the tagged traffic is spread over multiple paths, thereby achieving network load balancing. However, the improvement flattens with further increases in the number of dispersed paths. Surprisingly, the mean resequencing delay, which is exhibited in Fig. 7, increases slightly and then flattens (or falls slightly) as the number of paths used increases. This results from the fact that when the number of dispersed paths is sufficiently large (say three), the fluctuation of delays among different paths is offset by the reduction in variances of path delays. The delay fluctuation is due to the packets travelling on different paths experiencing variable delays, causing them to arrive at the destination out of order. Such fluctuation becomes more serious when the number of paths used increases. On the other hand, using more paths for data transmission reduces the tagged traffic load on each path, thereby diminishing the path delay variance. Furthermore, the performance improvement is more significant when the inter-packet time is smaller, as the load from the dispersed traffic to each path is higher. Thus, multipath routing is effective in performance improvement when the dispersed traffic load and the network load are both high.

Fig. 8 exhibits the mean resequencing buffer occupancies, and two bounds of their complementary distributions, namely at $10^{-3}$ and $10^{-5}$. They represent, to some extent, the system cost of multipath routing, because network administrators need to allocate sufficient buffer resources before a multipath connection can be established. From the figure, the mean resequencing buffer occupancy flattens as the number of dispersed paths increases, but this may not be the case when a packet loss bound is considered.
distribution to each dispersed path to further improve the performance. Consider there are two heterogeneous paths, with background traffic of different loads. Define $R$ as the dispersion ratio such that packets are transmitted on these two paths in a ratio of $R : 1$. Our result in Fig. 10 shows that the resequencing delay attains its minimum when the total traffic loads along these paths are the same and equal to 0.9, where $R = 3$. In other words, an optimal split of traffic is to ensure that the total load of each dispersed path is more or less the same, or network traffic load is well balanced, which is intuitively satisfying.

5. Conclusions

In this paper, we have proposed a framework to study the resequencing mechanism in high-speed networks. This framework allows us to estimate the packet resequencing delay and the resequencing buffer occupancy distributions when traffic is dispersed on multiple disjoint paths.

The traffic model has been constructed in a flexible manner so that any multipath routing mechanisms can be modelled easily. The resequencing model has been devised to allow us to compute all necessary performance metrics for resequencing. The end-to-end path delay model has been built to allow the queueing model for resequencing to be decoupled from that for a path. This leads to a simple yet general model, which can be used with other measurement-based tools for estimating end-to-end path delay distributions to find an optimal split of traffic.

To illustrate our proposed framework, we have considered a multiple-node $G/M/1$ tandem network as a path model. Our results show that the packet resequencing delay and the resequencing buffer occupancy drop when the traffic is spread over a larger number of homogeneous paths, although the network performance
improvement quickly saturates when the number of paths used increases. Multipath routing is effective in using a small number of paths, say, up to three.

Now we re-visit the three questions posed in Section 1. As expected, multipath routing improves the system performance by spreading traffic load into multiple paths, thereby achieving network load balancing. An optimal traffic split ensures that the network load is well balanced. However, the use of packet-based multipath routing may require a destination to resequence those packets that arrive out of order. Our results show that the probability distribution of the resequencing buffer occupancy tends to have a heavier tail for the cases when a larger number of paths is used. This argument does not favor using a large number of paths, say more than three, in multipath routing.

There are several possible extensions to our work, some of which are listed below:

- devise an adaptive multipath protocols for packet-switching networks;
- incorporate quality of service routing [2] with multipath routing;
- extend the framework to consider reliability and fault tolerant issues.

Acknowledgements

This research is supported in part by the University of Hong Kong Strategic Research Theme of Information Technology. We would like to express our gratitude to the anonymous reviewers for their valuable comments and suggestions which assisted us in improving the quality of this paper.

References

[25] Y. Xia, D.N.C. Tse, On the large deviations of resequencing queue size: 2-