This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

On Minimum-Delay Data Block Transport over Two-Connected Mesh Networks

Yonggang Wen, *Student Member, IEEE* and Jun Sun, *Student Member, IEEE*

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139
{eewyg, junsun}@mit.edu

Abstract*—In this paper we investigate the problem of sending data blocks containing *finite* number of packets over two independent routing paths in mesh networks. The objective is to minimize the average *block delay* by allocating packets into the two routing paths optimally. Previous researchers have shown that using more paths can reduce *packet delay* and the rate-based allocation policy is optimal, based on the delay metric solely depending on delay mean of each path. In our research, generalizing the multi-path routing scheme to accommodate data block transport, we first establish an upper bound for the average block delay, depending on both delay mean and variance of each path, and then solve a non-linear optimization problem to obtain the optimal packet allocation policy, which either allocates all the packets to the faster path for blocks of small size or allocate all the packets to both paths in proportion to their service rates for blocks of large size. Contrary to conventional results, our analysis suggests that using additional slower routing path could increase the average block delay in some cases. We further characterize the whole spectrum of optimal packet allocation policies as a function of block sizes, and conclude that the existing rate-based allocation policy is a special case for large data block.

I. INTRODUCTION

Recently, network designers are facing new challenges to provide end-to-end communications with desirable quality of service (QoS), mainly due to the proliferation of content delivery over Internet [1, 2]. Traditionally, network designers have focused on optimizing performance metrics for individual packet, such as, the *packet delay*. Compared to the individual packet delay, new performance metrics resulted from novel network applications should be sought for better network architectures and operations. In particular, because date files are separated into blocks with finite number of packets in the BitTorrent [1], which contributes roughly 35% of the total Internet traffic [2], the *data-block delay* should be one of the key performance metrics for network designers. It follows that the central question addressed in this paper is how to optimize network operations based on this emerging performance metric (i.e., the data block delay).

Technically, the delay performance of content delivery can be improved by exploiting multiple connections in mesh networks. The multi-path routing scheme [3, 4], where a source node allocates its data packets over multiple disjoint paths through the network to a destination node, presents improved performance compared to the traditional single-path routing. Indeed, it has been considered in various contexts. Previously, the problem of two nodes communicating over multiple paths has been considered extensively in wireline networks [5, 6]. Lately, with advent of wireless networks such as the Roof-Net [7], Low-Earth-Orbit satellite network, and ad-hoc networks, there is a rising interest in multi-path routing research [6-7] for wireless networks. Specifically, the multi-path routing scheme is sought for various operational benefits, for example, traffic balancing, higher aggregate capacity (or reduced packet delay) and path diversity for higher reliability. Multi-path routing, due to its diverse routing path, becomes an efficient means in mitigating the unreliable channels due to fading and interference, and thus provides an improved error performance as demonstrated in [4]. In [8], the authors propose models to analyze and compare single-path and multi-path routing protocols in terms of overheads, traffic distribution and connection throughput in a mobile ad-hoc network. In addition, a good packet delay performance can be achieved by exploiting the flexibility inherent to multi-path routing. For example, in [9], the authors develop a framework for optimal rate allocation among multiple routing paths in multi-hop wireless networks, where analytical results for optimal rate allocation for Poisson arrivals at each node are derived. In this research, we address the problem of how to best exploit the available multi-paths to reduce the delay of sending a data block of finite size, for example, a file block (~16KB) in the BitTorrent transaction, over heterogeneous mesh networks.

In this research, we start with the two-connected case, where the source node and the destination node are connected by two disjointed routing paths. Practically, two-connected source-destination pairs are ubiquitous in current network infrastructures. One example is the commercial handset that can talk to both Wifi network and Cellular network [10], which route the packets through different infrastructures. Another example is the bandwidth sharing scheme for two residential neighbors, developed by Mushroom Networks [11] and WiBoost Inc [12]. Although this is a limited case, our investigation reveals deep insights to understanding data block delivery over multiple-connected source-destination pairs, as illustrated later.

In particular, we consider the problem of minimizing the average delay of sending a data block consisting of n packets through two disjointed paths in heterogeneous mesh networks. For each path available for transmission, the time that takes for each packet to traverse the path and reach its destination is modeled as an independent random variable with some known distribution (e.g., the exponential distribution). The data-block delay is then defined as the time interval from the transmission of the first packet to the time that all of the n packets are received on the destination side. Our objective is to identify the optimal number of packets to be allocated to each path so as to minimize the average data-block delay.

The results obtained in this research complement and generalize conventional results on the multi-path routing scheme. For the extreme case of infinite number of packets, previous researchers have shown that using more paths can reduce individual *packet delay* and the rate-based allocation policy [9, 13] is optimal, based solely on delay mean of each path. In our research, we generalize the multi-path routing scheme to accommodate data blocks of finite size and focus on the *data-block delay*. Under the assumption of exponential packet delay for each path, we first establish an upper bound for the average block delay, contributed by both the delay mean and the delay variance of each path. In addition, we solve a non-linear optimization problem to obtain an optimal packet allocation policy, which either allocates all the packets to the faster path for data blocks of small size or allocate all the packets to both paths in proportion to their service rates for data blocks of large size. Contrary to conventional results, our analysis suggests that using additional slower routing path could increase the average data-block delay in some cases, resulting from an *empty queue phenomenon* where the finished path has to wait for the other unfinished path. We further characterize the whole spectrum of optimal packet allocation policies in regarding to different block sizes, and conclude that the existing rate-based allocation policy [9, 13] is a special case for large data block.

This paper is organized as follows. In Section II, the network model is introduced for the packet allocation problem over the two-connected source-destination pair. In Section III, the lower and upper bounds on the average data-block delay are established. In Section IV, we solve the packet allocation problem for the two-connected source-destination pair and characterize the whole spectrum of optimal packet allocation policies in regarding to different block sizes. Section V concludes this paper.

II. Data Block Transport Over Two-Connected Mesh Networks

A. Network Delay Model

We model each routing path as an FIFO (first-input-first-output) queue. The service time of each FIFO queue can be modeled as a random variable with a generalized distribution. Following the widely adopted exponential delay model in network research community, we assume that the packet delay along routing path i is modeled as an exponential random variable with rate of λ_i, $i=1,2$. In addition, we assume that delays experienced by different packets on the same routing path are identically and independently distributed, and delays experienced by different packets on different routing paths are independent. Although it is a limited case to assume an exponential delay distribution, it provides deep insights in here understanding of more practical delay distributions and suggests useful operating rules for practical networks.

We also assume that the source node has no access to instant channel state information, but the long-term average channel statistics, for example, the service rate vector $\boldsymbol{\lambda}=(\lambda_1,\lambda_2)$, are available to the source node. In practical networks, the real-time channel state information can be too expensive for the source node to acquire; or in some cases even unavailable to the source node in a timely fashion due to physical constraints such as long round-trip delay in satellite networks.

B. Finite-Size Data Block Transport over Two-Connected Source-Destination Pairs

For a heterogeneous mesh network, a data block containing n packets of equal length, is to be transported between a two-connected source-destination pair.

The inefficiency of the finite-size data block transport over two-connected source-destination pairs results from *an empty queue phenomenon*, that is, one path could have finished its own packets and be waiting for the other path to finish its own packets. This happens with a non-zero probability because there are not always packets to transport. The empty queue situation means a waste of network resource and thus results in a prolonged block delay. Two alternative strategies can combat the empty queue phenomenon and improve the delay performance, as illustrated next.

One approach is to design optimal packet allocation policies to minimize the occurrence frequency of an empty queue on either path. For the extreme case of infinite number of packets per block, previous researchers have shown that the rate-based allocation policy [9, 13-14] is optimal. We would like to generalize the optimal packet allocation polices to the whole spectrum of different block sizes.

The other approach is based on an inter-packet encoding technique to fill the empty queue with redundant packets. At the source node, the set of n original packets are encoded into a set of l packets via some erasure channel codes such as the Digital Fountain code [15] to add some redundancy (i.e., $l \geq n$). The coded packets are then allocated to the two disjointed paths based on their service rates. At the destination node, the original n data packets can be decoded reliably after receiving more than n packets. The block delay improvement comes from the fact that the destination can decode the original data block without waiting for all the packets, which could arrive much later due to the delay variance. However, injecting redundant packets into the network would increase the network congestion level and thus result in a prolonged packet delay. This increase packet delay will mitigate the benefit of redundancy packets. It follows that the trade-off between the average block delay and the code rate (i.e., $R=n/l$) should be characterized.

Due to the limited space in this paper, we focus on characterizing the minimum-delay packet allocation policy without coding, and will present the inter-packet coding approach in a sequel paper.

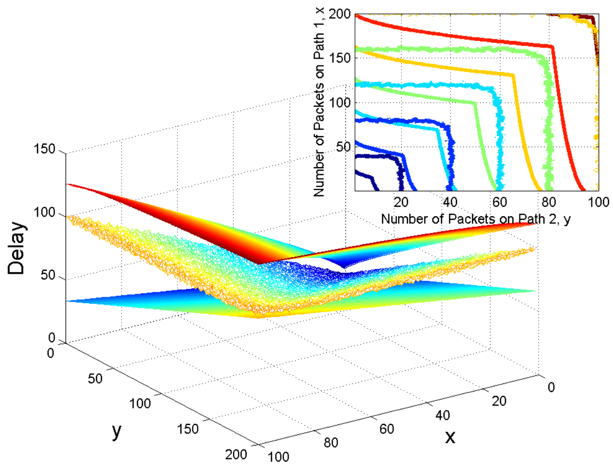

Fig. 1. Both delay lower bound and upper bound are compared with the simulated file delay with $\lambda_1 = 2$ and $\lambda_2 = 1$.

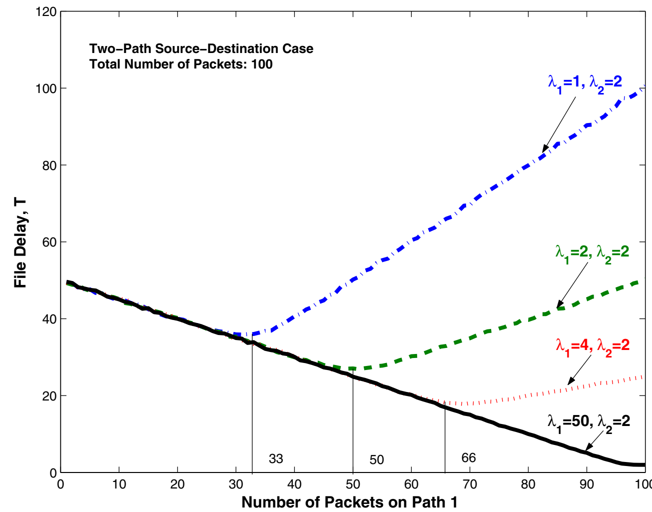

Fig.2. Illustration of the optimal packet allocation policies for 100 packets and different service rates.

III. Lower and Upper Bounds on Average Block Delay

In this section, lower and upper bounds on the average block delay are developed for any fixed packet allocation policy over two disjoint routing paths.

Without loss of generality, we assume the fixed packet allocation policy with x packets allocated to path 1 of service rate λ_1 and y packets for path 2 of service rate λ_2 . We denote D_1 and D_2 as the delays of transmitting all the allocated packets on path 1 and 2, respectively. The data-block delay is then defined as the total time for all the $x+y$ packets to arrive at the destination, i.e.,

$$T = \max\{D_1, D_2\}, \tag{1}$$

where D_1 and D_2 are two Erlang random variables of order x and y , respectively.

First, the upper bound can be derived through an ideal scenario. If the source node knows queue states of the two routing paths, it can allocate a packet to a routing path whenever the routing path has finished serving the previous one. Under such an allocation policy, the average block delay can be derived as

$$T^{LB} = n/(\lambda_1 + \lambda_2), \tag{2}$$

which is a lower bound on the minimum average block delay with or without coding. Notice that this lower bound can be achieved with a simple inter-packet code. If we encode the original n packets into $2n$ data packets (i.e., the code rate $R = 1/2$) and allocate n packets along each routing path, the average block delay is equal to the lower bound.

Second, using the Chernoff bound (*omitted due to limited space*), we can derive an upper bound on the average block delay for the destination to receive all the packets as

$$E[T] \le \max\left\{\frac{x}{\lambda_1}, \frac{y}{\lambda_2}\right\} + \sqrt{2\pi}\left(\sqrt{\frac{x}{\lambda_1^2}} + \sqrt{\frac{y}{\lambda_2^2}}\right) \triangleq T^{UB}, \tag{3}$$

where x/λ_1 and y/λ_2 are average delays for sending x packets over path 1 and y packets over path 2, respectively; and x/λ_1^2 and y/λ_2^2 are delay variances of sending x packets over path 1 and y packets over path 2, respectively. It suggests that the delay upper bound is equal to the sum of the maximum of the delay mean along each path and the delay standard deviation bound timed by a constant.

In Fig. 1, we plot the lower bound (2) and the upper bound (3) of the average block delay, compared with the simulated average block delay. We observe that the shape of the upper bound is similar to the shape of the simulated average block delay. The contours of the upper bound and the simulated average delay in Fig. 1 also verify the curvature similarity between them. This suggests that the delay upper bound can be used to investigate the optimal packet allocation policy for the two-connected source-destination pair, and the resulted optimal policies would be close to the one obtained by using the exact average block delay, which is hard to derive.

IV. Optimal Packet Allocation Policies

A. Minimum-Delay Packet Allocation Policies

In this subsection, we characterize the optimal policy of allocating any data block of finite size to two disjoint paths via a non-linear programming approach.

Using the delay upper bound (3) as the cost metric, we can identify the optimal packet allocation policy by solving the optimization problem,

$$\begin{aligned} \min \quad & \max\left\{\frac{x}{\lambda_1}, \frac{y}{\lambda_2}\right\} + \frac{\sqrt{2x\pi}}{\lambda_1} + \frac{\sqrt{2y\pi}}{\lambda_2}. \\ s.t \quad & x + y = n \end{aligned} \tag{4}$$

In the following, we first assume that $\lambda_1 \ge \lambda_2$ and all the results can be easily modified for the case of $\lambda_1 \le \lambda_2$. Under such a simplified assumption, we define two allocation policies as follows:

Proportional allocation policy (PA): each path is allocated with a number of packets in proportion to its service rate, i.e.,

$$\left(x^*, y^*\right) = \left(\frac{\lambda_1}{\lambda_1+\lambda_2}n, \frac{\lambda_2}{\lambda_1+\lambda_2}n\right). \tag{5}$$

Dominant allocation policy (DA): all the packets is allocated to the path with the dominant service rate,

$$\left(x^*, y^*\right) = (n, 0). \tag{6}$$

Solving the nonlinear optimization problem (4) in *Appendix A*, we obtain the following theorem for the optimal packet allocation policy.

Theorem 1 *To minimize the average delay of sending an uncoded data block of n packets between two disjointed paths with service rates of $\lambda_1 \geq \lambda_2$, the optimal allocation policy is either the proportional allocation policy (5) or the dominant allocation policy (6).*

As a sanity check, we plot the average block delay versus the number of packets allocated to path 1 for the case of 100 packets and various settings of service rates in Fig. 2. We observe that the optimal packet allocation policy is either the proportional allocation policy or the dominant allocation policy.

Theorem 1 has several important implications. First, it suggests that complexity of the optimal packet allocation problem can be reduced significantly. Since the optimal packet allocation policy can be identified by comparing the two candidate policies, the complexity to identify the optimal packet allocation policy is only $O(1)$. On the other hand, if we apply the brute-force searching algorithm by comparing all the $n+1$ candidate policies as in Fig. 2, the complexity is $O(n\log n)$. Second, it also suggests that the rate-based allocation policy [9, 13-14] is a special case of our minimum-delay packet allocation solution for data blocks of large size. As to be shown in next sub-section, the optimal packet allocation policy evolves from the dominant allocation policy when n is small and turns into the proportional allocation policy when n is large.

B. Characterization of Optimal Packet Allocation Policies

In this sub-section, we characterize the optimality conditions for both candidate packet allocation policies, suggested in Theorem 1. This investigation discloses the whole spectrum of optimal allocation policies as a function of data block size.

First, as shown in *Appendix A*, the dominant allocation policy (6) is optimal if the following condition is satisfied,

$$\left|\sqrt{\frac{\pi n}{2\lambda_1^2}}\frac{1}{\sqrt{\alpha}} - \sqrt{\frac{\pi n}{2\lambda_2^2}}\frac{1}{\sqrt{1-\alpha}}\right| \geq \frac{n}{\lambda_1}, \tag{7}$$

where $\alpha = \lambda_1/(\lambda_1+\lambda_2)$. Using the fact that $\lambda_1 \geq \lambda_2$, we can bound the left hand side of (7) as

$$\left|\sqrt{\frac{\pi n}{2\lambda_1^2}}\frac{1}{\sqrt{\alpha}} - \sqrt{\frac{\pi n}{2\lambda_2^2}}\frac{1}{\sqrt{1-\alpha}}\right| \geq \frac{\sqrt{\pi n}}{\lambda_2} - \frac{\sqrt{\pi n}}{\lambda_1}. \tag{8}$$

To satisfy the condition of (7), it is sufficient to have

$$\frac{\sqrt{\pi n}}{\lambda_2} - \frac{\sqrt{\pi n}}{\lambda_1} \geq \frac{n}{\lambda_1}, \tag{9}$$

which suggests that

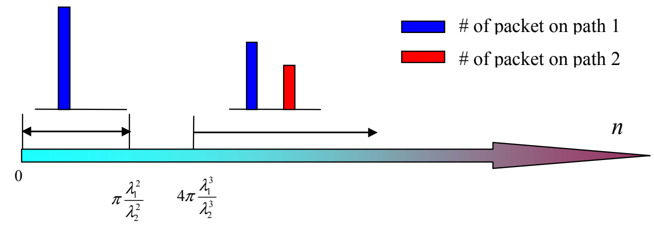

Fig. 3. The whole spectrum of the optimal packet allocation policy is plot as a function of block size. As the block size increases, the optimal packet allocation policy evolves from the dominant allocation policy to the proportional allocation policy.

$$\lambda_1/\lambda_2 \geq \sqrt{n/\pi} + 1. \tag{10}$$

When the number of packets n is large enough, (10) can be approximated as

$$n \leq \pi\left(\lambda_1/\lambda_2\right)^2, \tag{11}$$

which indicates that, when the number of packets in the data block is smaller than the threshold of $\pi\lambda_1^2/\lambda_2^2$, we should assign all the packet to the path with the higher service rate.

The optimality condition of the dominant allocation policy can be understood intuitively as follows. Using the O-notation, we can re-arrange (11) as

$$n/\lambda_1^2 = O\left(1/\lambda_2^2\right), \tag{12}$$

where $1/\lambda_2^2$ is the delay variance of allocating one packet to path 2 with slower service rate and n/λ_1^2 is the delay variance of assigning all the packets to path 1 with higher service rate. It suggests that, when the delay variance of assigning one packet to the slower path is comparable to or larger than the delay variance of assignment all the packets to the faster path, it is optimal to assign all the packets to the faster path. Accordingly, the intuition behind the dominant allocation policy is to avoid the empty queue situation where all the packets on the faster path have arrived but the destination has to wait for the packets from the slower path, which could arrive much later due to its large delay variance.

Second, the optimality condition for the proportional allocation policy can be characterized by comparing the performance of the two candidate policies. On one hand, if we apply the proportional allocation policy, the average block delay is bounded above by

$$T_{PA}^{UB} = \frac{n}{\lambda_1+\lambda_2} + \sqrt{\frac{2\pi n}{\lambda_1(\lambda_1+\lambda_2)}} + \sqrt{\frac{2\pi n}{\lambda_2(\lambda_1+\lambda_2)}}. \tag{13}$$

On the other hand, if we apply the dominant allocation policy, the average block delay is bounded above by

$$T_{DA}^{UB} = \frac{n}{\lambda_1} + \sqrt{\frac{2\pi n}{\lambda_1^2}}. \tag{14}$$

Denoting $c = \lambda_1/\lambda_2$ ($c \geq 1$ by assumption), the delay difference between these two policies can be bounded from below as

$$T_{DA}^{UB} - T_{PA}^{UB} \geq \frac{\sqrt{n}}{\lambda_1\sqrt{1+c}}\left(\sqrt{\frac{n}{1+c}} - c\sqrt{2\pi}\right). \tag{15}$$

Specifically, the assumption of $1 \leq c \leq \sqrt[3]{n/4\pi}$ can make the right hand side of (15) positive, indicating that the proportional allocation policy outperforms the dominant allocation policy. It suggests that when the number of packets in the data block satisfies

$$n \geq 4\pi\left(\lambda_1/\lambda_2\right)^3 , \tag{16}$$

the proportional allocation policy is the optimal.

Combining (11) and (16), we conclude the following *rules of thumb* to decide the optimal packet allocation policy for the two-connected source-destination pair with $\lambda_1 \geq \lambda_2$:

1)When $n \leq \pi \lambda_1^2/\lambda_2^2$, the optimal policy is the dominant allocation policy which allocates all the packets to path 1;

2)When $n \geq 4\pi \lambda_1^3/\lambda_2^3$, the optimal policy is the proportional allocation policy which assigns all the packets to both paths according to their service rates;

3)When $\pi \lambda_1^2/\lambda_2^2 < n < 4\pi \lambda_1^3/\lambda_2^3$, the optimal policy can be identified by comparing (13) with (14).

Practically, this set of rules of thumb can be considered as the set of sufficient optimality conditions for optimal packet allocation policies. As illustrated in Fig. 3, for a given ratio of λ_1/λ_2 , the optimal allocation policy evolves from the dominant allocation policy to the proportional allocation policy as the number of packets in a data block increases. When $n \leq \pi \lambda_1^2/\lambda_2^2$, the dominant allocation policy prevails; when $n \geq 4\pi \lambda_1^3/\lambda_2^3$, the proportional allocation policy outperforms; otherwise, the optimal allocation policy is determined by comparing (13) and (14).

V. Conclusion

In this paper we considered the problem of minimizing the average delay of sending a data block of fixed size through any two-connected source-destination pair in heterogeneous mesh networks.

Previous research suggested that sending packets over two disjoint routing paths results in a smaller *packet delay* than the single path routing approach due to the utilization of more network resources. Generalizing the multi-path routing scheme to accommodate data block transport, we observed that the empty queue phenomenon, where the finish path has to wait for the unfinished path, is the critical issue in the optimal design. We first established that the average *data-block delay* takes contributions from the delay mean and the delay variance of each path, and then identified the minimum-delay packet allocation policy as a result of balancing the two components of delay contributions. On one hand, when the number of packets n in a data block is larger than a threshold, which depends on the ratio of two service rates, the optimal allocation policy is to assign all packets to both paths in proportion their service rates. On the other hand, when the number of packets n in a data block is less than a threshold such that the delay variance for one packet through the slower path is comparable to that for all the packets through the faster path, the optimal allocation policy is to assign all packets to the path with higher service rate. We also established the whole spectrum of optimal packet allocation policies as a function of the block size and concluded that the previous rate-based packet allocation policy as a special case of the general two-path transmission packet allocation problem for large block size.

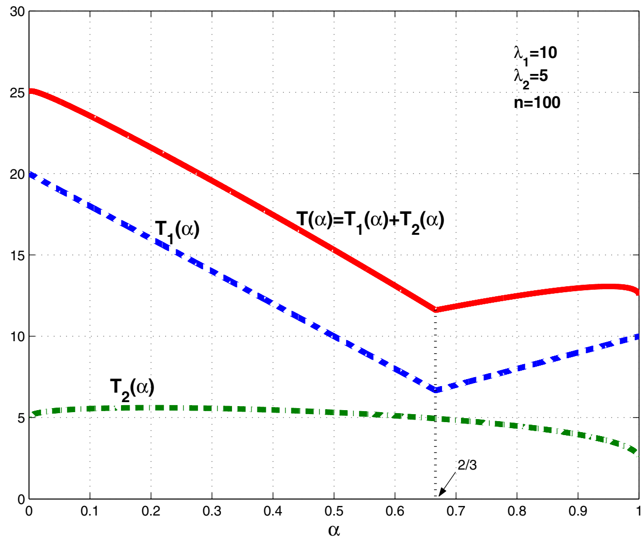

Fig. 4. An illustration of the piece-wise linear V shape of function $T_1(\alpha)$ and the concavity of function $T_2(\alpha)$ with $\lambda_1 = 10$, $\lambda_2 = 5$ and $n = 100$.

In addition, we derived a lower bound on the minimum average data-block delay. This lower bound can be achieved by sending n coded packets along each routing path (i.e., total $2n$ coded packets) such that the destination can decode the original n packets as long as it receives n coded data packets. This simple observation motivates us to exploit the inter-packet coding strategy and identify the optimal coding strategy to minimize the average data block delay in a sequel paper.

Appendix

A. Proof of Theorem 1

The optimal packet allocation policy can be obtained by solving the following nonlinear optimization problem,

$$\begin{aligned} \min \quad & \max\left\{\frac{x}{\lambda_1}, \frac{y}{\lambda_2}\right\} + \frac{\sqrt{2x\pi}}{\lambda_1} + \frac{\sqrt{2y\pi}}{\lambda_2} . \\ s.t \quad & x + y = n \end{aligned} \tag{A.1}$$

Alternatively, the optimization problem can be made simpler by the following transformation. If we denote $x = \alpha n$, $0 \leq \alpha \leq 1$, we must have $y = (1-\alpha)n$. It follows that the optimization problem (A.1) can be rewritten as

$$\min_{0\leq\alpha\leq1} \bar{T}(\alpha) = \max\left\{\frac{n}{\lambda_1}\alpha, \frac{n}{\lambda_2}(1-\alpha)\right\} + \sqrt{2\pi n}\left(\frac{\sqrt{\alpha}}{\lambda_1} + \frac{\sqrt{(1-\alpha)}}{\lambda_2}\right) . \tag{A.2}$$

For the special case of $\lambda_1 = \lambda_2$, the optimum is $\alpha^* = 1/2$ by symmetry, which is a special case of the proportional allocation policy. In the following, we assume that $\lambda_1 > \lambda_2$. We observe that the objective function in (A.2) can be decomposed into two parts. The first term is denoted as

$$T_1(\alpha) = \max\left\{\frac{n}{\lambda_1}\alpha, \frac{n}{\lambda_2}(1-\alpha)\right\}, \quad \text{(A.3)}$$

and the sum of the second and third term is denoted as

$$T_2(\alpha) = \frac{\sqrt{2\pi n\alpha}}{\lambda_1} + \frac{\sqrt{2\pi n(1-\alpha)}}{\lambda_2}. \quad \text{(A.4)}$$

To solve the optimization problem (A.1), we first characterize the properties of (A.3) and (A.4).

First, it can be verified that $T_1(\alpha)$ has a V shape within its support of $[0,1]$, and a global minimum at $\alpha^* = \lambda_1/(\lambda_1+\lambda_2)$. Within the range of $[\alpha^*,1]$, $T_1(\alpha)$ increases linearly with slope of n/λ_1 as α increases; at the same time, within the range of $[0,\alpha^*]$, $T_1(\alpha)$ increases linearly with slope of n/λ_2 as α decreases. As an illustration, Fig. 4 plots function $T_1(\alpha)$ with $\lambda_1 = 10$, $\lambda_2 = 5$ and $n = 100$.

Second, the first derivative of $T_2(\alpha)$ is given by

$$T_2^{'}(\alpha) = \sqrt{\frac{\pi n}{2\lambda_1^2}}\frac{1}{\sqrt{\alpha}} - \sqrt{\frac{\pi n}{2\lambda_2^2}}\frac{1}{\sqrt{1-\alpha}}, \quad \text{(A.5)}$$

and its second derivative is given by

$$T_2^{''}(\alpha) = -\sqrt{\frac{\pi n}{8\lambda_1^2}}\frac{1}{\sqrt{\alpha^3}} - \sqrt{\frac{\pi n}{8\lambda_2^2}}\frac{1}{\sqrt{(1-\alpha)^3}}. \quad \text{(A.6)}$$

We observe that $T_2^{''}(\alpha)$ must be negative since both terms in (A.6) have negative signs. It follows that $T_2(\alpha)$ is a strictly concave function in the range of $[0,1]$, as shown in Fig. 4.

At the point of $\alpha^* = \lambda_1/(\lambda_1+\lambda_2)$, it can be verified that $T_2^{'}(\alpha^*) < 0$. It follows that, within the neighborhood of α^*, $T_2(\alpha)$ is a strictly decreasing function. Moreover, within the range of $(\alpha^*,1]$, $T_2(\alpha)$ is a strictly decreasing function as α increases and the speed of decreasing increases due to the concave property. At the same time, within the range of $[\alpha^*-\delta,\alpha^*)$ for some small $\delta > 0$, $T_2(\alpha)$ increases as α decreases.

Using the properties of $T_1(\alpha)$ and $T_2(\alpha)$, we can make the following observations. On one hand, if

$$\left|T_2^{'}(\alpha^*)\right| \geq \frac{n}{\lambda_1}, \quad \text{(A.7)}$$

$T_2(\alpha)$ decreases faster than $T_1(\alpha)$ increases as α increases in the range of $(\alpha^*,1]$. In this case, $\overline{T}(\alpha) = T_1(\alpha) + T_2(\alpha)$ decreases as α increases and is minimized at $\alpha^\dagger = 1$. Therefore, the optimal allocation policy is to allocate all the packets to path 1, i.e., $(x^*, y^*) = (n,0)$. On the other hand, if

$$\left|T_2^{'}(\alpha^*)\right| < \frac{n}{\lambda_1}, \quad \text{(A.8)}$$

$T_2(\alpha)$ decreases slower than $T_1(\alpha)$ increases as α increases in the range of $(\alpha^*,1]$. It follows that α^* is a local minimum of the block delay upper bound. In this case, as shown in Fig.4, another possible local minimum is $\alpha^\dagger = 1$. Therefore, the set of local minimums are $\{\alpha^*,\alpha^\dagger\}$, which are possible candidates for the global minimum.

In summary, only two solution candidates of optimal allocation policies exist for the optimization problem (A.1). One is the service-rate proportional allocation policy, i.e.,

$$(x^*, y^*) = \left(\frac{\lambda_1 n}{\lambda_1+\lambda_2}, \frac{\lambda_2 n}{\lambda_1+\lambda_2}\right). \quad \text{(A.9)}$$

The other is the dominant assignment policy which allocates all the packets to the path with higher service rate, i.e.,

$$(x^*, y^*) = (n,0). \quad \text{(A.10)}$$

To identify the optimal assignment policy, one can simply compare the delay bounds for both policies and take the one with lower average block delay bound as the optimal policy.

ACKNOWLEDGMENT

The authors wish to thank Prof. Vincent W.S. Chan, Dr. Terrence McGarty, Prof. Eytan Modiano, Prof. Lizhong Zheng and Prof. Devarat Shah at MIT for helpful discussions.

REFERENCES

[1] http://www.bittorrent.com/.

[2] http://in.tech.yahoo.com/041103/137/2ho4i.html.

[3] N. F. Maxemchuk, "Dispersity Routing," Proceedings of ICC '75, San Francisco CA, June 1975, pg. 41-10, 41-13.

[4] V. W. S. Chan and A. H. Chan, "Reliable Message Delivery via Unreliable Networks," *1997 IEEE International Symposium on Information Theory*, Ulm, Germany, 29 June - 4 July 1997.

[5] E. Gustafsson and G. Karlsson, "A Literature Survey on Traffic Dispersion," *IEEE Network*, Vol. 11, No. 2, March 1997, pp. 28-36.

[6] R. Krishan and J.A Silvester, "Choice of allocation granularity in multi-path source routing schemes," in *IEEE INFOCOM'93*. IEEE, 1993, vol. 1, pp. 322–29.

[7] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level Measurements from an 802.11b Mesh Network," *SIGCOMM 2004*, Aug 2004.

[8] P.P. Pham and S. Perreau, "Performance Analysis of Reactive Shortest Path and Multipath Routing Mechanism with Load Balance", *INFOCOM 2003*, March 2003.

[9] R. Agarwal and A. Goldsmith, "Joint Rate Allocation and Routing for Multihop Wireless Networks with Delay-Constrained Data", *Globecom* 2004.

[10] http://featured.gigaom.com/2006/08/16/t-mobile-wifi-cellular-launch-in-september/.

[11] http://www.mushroomnetworks.com/v7/company.htm/.

[12] http://wiboost.com/.

[13] J. Li and J. Brassil, "On the performance of traffic equalizers on heterogeneous communication links.," In *The Third International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks* (QShine 2006), Waterloo, Canada, August 2006.

[14] M. Luby, "LT Codes," in Proc. 43rd Annual *IEEE Symposium on Foundations of Computer Science*, November 2002, pp. 271-280.

[15] S. Mao, S. S. Panwar and Y. T Hou, "On Optimal Partitioning of Realtime Traffic over Multiple Paths," in Proceedings of *IEEE INFOCOM'05*.