Delay-Aware Coding in Multi-Hop Line Networks

Derya Malak*†, Arno Schneuwly*, Muriel Médard*
*Research Laboratory of Electronics, MIT, Cambridge, MA, USA
{deryam, aschneuw, medard}@mit.edu

Edmund Yeh†
†Northeastern University, Boston, MA, USA
{eyeh}@ece.neu.edu

***Abstract*—We propose an adaptive coding technique for delay constrained simple wireless multi-hop line networks. We study the joint optimization of coding and scheduling with one sender and one receiver, with delay sensitive flows consisting of packets. We analyze the trade-off between delay and throughput for single path multi-hop wireless erasure links for different encoding and feedback acknowledgment schemes. To do so, we devise an ℓ_p-norm delay cost metric where p models the sensitivity of the receiver. We show that with adaptively adjusting the coding bucket size based on p and the feedback delay, recoded multi-hop transmissions can increase the throughput of the end-to-end coded transmissions by 30%.**

I. Introduction

We develop encoding and scheduling techniques for optimizing the delay cost of the end user in a wireless mesh network. While capacity-achieving schemes have been well-studied, such as in [1] and [2], and techniques to optimize average delay have been developed in [3] and [4]; to the best of our knowledge, delay-throughput trade-offs in mesh networks have not been fully understood. In reality, the coding protocol has to achieve application-specific communication delays at low cost by dynamically adapting the coding bucket size and routing decisions. We consider different coded transmission techniques for coverage; and feedback acknowledgment schemes to characterize the delay performance of single path multi-hop line networks by taking into account physical layer considerations, and analyze their delay and throughput trade-offs by devising a delay cost function that exploits the sensitivity of the receiver.

We study the joint optimization of coding and scheduling of wireless links in tandem for varying delay sensitivities. To analyze the trade-off between delay and throughput in multi-hop networks, we devise an ℓ_p-norm delay cost metric where p denotes the sensitivity of the receiver. By extending the adaptive point-to-point scheme of Zeng *et al.* in [5], shown in Fig. 1, to multi-hop line networks, we characterize *the trade-off between delay and throughput* for unicast connections, for the following coding and acknowledgment (ACK) schemes:

I. Recoded scheme with end-to-end ACK (not shown);
II. End-to-end coding with end-to-end ACK (Fig. 2-(a));
III. Uncoded scheme with link-by-link ACK (not shown);
IV. Recoded scheme with link-by-link ACK (Fig. 2-(b)).

We numerically demonstrate the benefits of coded schemes.

In the following sections (Sects. II–IV), we describe the adaptive point-to-point coding scheme, and detail the analysis of the different coding schemes for multi-hop line networks.

978-1-5386-4980-0/19/$31.00 ©2019 IEEE

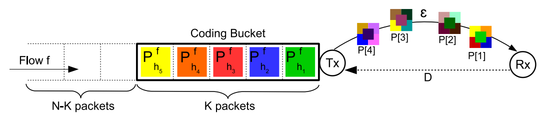

Fig. 1: Single path adaptive linear coding with a sender (Tx) and a receiver (Rx) [5]. Each coded packet $P[t]$ at time slot t is a linear combination of the packets in the bucket, and D is the ACK delay.

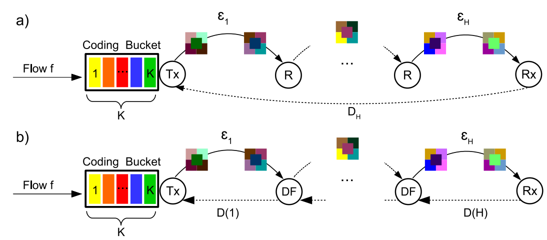

Fig. 2: Multi-hop line network consisting of H hops with different loss probabilities ϵ_i. (a) End-to-end: intermediate nodes act as relays (R). (b) Recoded: intermediate nodes recode-and-forward (RF).

A. Related Work and Motivation

Retransmission of lost packets is a capacity-achieving strategy in point-to-point communication under the assumption of perfect feedback. However, feedback-based approaches are not well-suited to lossy wireless networks. Feedback may be unreliable or delayed in the case of satellite or wireless networks or real-time applications. Furthermore, due to packet congestion, end-to-end retransmissions might be preferred. However, end-to-end retransmissions are not suited for multicast connections because there may be many requests that place an unnecessary load on the network. As an alternative to end-to-end coded schemes, link-by-link retransmissions where packets are routed hop-by-hop toward their destinations, and low link-by-link feedback acknowledgment delay can be used.

In wireless networks, retransmission-based schemes require a large amount of feedback to achieve reliability. However, if the links are not reliable enough, feedback may hurt more. Packet-level coding is capacity-achieving and also resilient against erasures in wireless links [2], which alleviates the need of a great deal of feedback. Coding over packets can also provide cooperation gains because nodes that are not transmitting packets can assist the nodes that are. Network coding for cost optimal multicast has been studied in [1]. A capacity-achieving packet-level coding scheme for unicast and single multicast connections has been proposed in [2].

Mesh networking aims to provide ubiquitous connectivity and Internet access in urban and rural environments, with few gateway points, and a flexible deployment [6]. A multi-radio unification protocol for multi-hop networks has been developed in [7] to optimize local spectrum usage via intelligent channel selection. Recent studies have shown that WiFi routers are a good alternative to long-distance links that require high-gain directional antennas and expensive base stations [8]. Using multi-hop paths with stronger links for long backhaul connections may provide better rates, and be a cost-effective alternative for connectivity. To this end, secure, reliable MP routing protocols have been devised in [9], and energy-aware routing for multi-hop wireless networks have been developed in [10]. These initial attempts motivate us to investigate the feasibility of using multi-hop for backhaul connections. It has been also verified that unlike hierarchical cooperation and distributed MIMO communication in dense networks [11], multi-hop can achieve better capacity scalings [12].

II. Adaptive Coding for Point-to-Point Links

We consider an adaptive coding scheme for a point-to-point wireless erasure channel connecting the sender (Tx) to the receiver (Rx). The sender wants to transmit a flow f consisting of N packets $\{P_1^f,\dots,P_N^f\}$ to the receiver. All data packets are available at the sender prior to any transmission. Each data packet has the same size L. At any time slot, the sender can transmit a packet to the receiver, through the erasure channel.

The adaptive coding scheme was first introduced in [5], as a modification of a generation-based network coding. A coding bucket is created that acts like the head of the line (HOL) generation, and K is the number of packets in the bucket. For encoding, the sender sequentially partitions the N packets into generations $\{G_1^f,\dots,G_M^f\}$ where $M=\lceil N/K\rceil$ is the number of generations. Although the generations may have different number of packets, i.e. the bucket size may change over time, we assume that it remains the same for the flow f. Given the bucket size at a time slot t, the sender reads the HOL generation $G_g^f=\{P_{g_1}^f,\dots,P_{g_K}^f\}$ where g is the generation (bucket) index. The sender generates a coded packet $P[t]$ as a linear combination of all packets in G_g^f: $P[t]=\sum\limits_{k=1}^{K}a_k[t]P_{g_k}^f$, where $a[t]=(a_1[t],\dots,a_K[t])$ is the coding coefficient vector, uniformly chosen at random from the space $\mathbb{F}_q^K$ over some finite field $\mathbb{F}_q$ [13]. The receiver collects coded packets over time. If q is large enough, the receiver can decode the K packets with high probability. The receiver then sends an ACK message via the feedback link to the sender, which is successfully received by the sender after D time slots. Upon receiving the ACK, the sender moves to the new HOL generation by adjusting K adaptively based on the receiver delay constraints. The system describing the adaptive linear network coding-based transmission is shown in Fig. 1.

A. In-Order Delivery Delay

We assume that the receiver buffers the packets to ensure in-order final delivery of N packets of flow f. Consider the transmission of a bucket of K packets $G_g^f=\{P_{g_1}^f,\dots,P_{g_K}^f\}$. Once the receiver obtains K linearly independent coded packets of the bucket, it will decode all K packets together. It then sends an ACK to the sender, which is always received correctly in D time slots. When informed, the sender then empties the coding bucket and moves in K new packets sequentially.

Let T_i be the final in-order delivery time slot in which the i^{th} original packet is decoded at the receiver. For ease of exposition, let T_{g_j} be the final in-order delivery time slot in which the j^{th} packet in the g^{th} bucket, i.e., the $(gK+j)^{\text{th}}$ original data packet, is decoded at the receiver. Since all packets in G_g^f are decoded together, the final in-order inter-arrival time slot ΔT_{g_j} at the sender satisfies $\Delta T_{g_j}=T_{g_j}-T_{g_{j-1}}=0$ for $j=2,\dots,K$. More specifically,

$$\Delta T_{g_j}=\begin{cases}T_{g_K}+D, & \text{if} \quad j=1,\\ 0, & \text{if} \quad j\in\{2,\dots,K\}.\end{cases}.$$

The delay metric only contains the decoding delay. We consider the lightly loaded case, where the arrival rate is lower than the service late. Hence, the queueing delay is negligible. In the heavily loaded regime, the mean waiting time (including queueing delay and service time) can be derived using the Pollaczek–Khinchin formula [14] that depends on the first and second moments of the HOL file completion time [15].

Given the packet erasure probability ϵ of the forward link, the transmission rate of the coded packets is $r=1-\epsilon$. The expected time for receiving K linearly independent coded packets is K/r. Hence, the average value of ordered inter-arrival time ΔT_{g_j} for the j^{th} packet in bucket g is

$$\mathbb{E}[\Delta T_{g_j}]=\begin{cases}\frac{K}{r}+D, & \text{if} \quad j=1,\\ 0, & \text{if} \quad j=\{2,\dots,K\}.\end{cases}, \quad (1)$$

where the expectation is taken over the distribution of packet erasures over the system and all the randomness associated with the coding and scheduling scheme.

Consider the special case when the packet injection process is a renewal process where a renewal occurs with probability $r=1-\epsilon$ at time slot $t\in\{1,2,\dots\}$. Then, the times between the final in-order delivery time slots T_{g_j} for $j=\{1,\dots,K\}$ in coding bucket $g=\{1,\dots,M\}$ are i.i.d. Geometrically distributed random variables with success probability r.

B. Delay Cost Function for the Adaptive Coding Scheme

We devise a delay metric as function of the in-order delivery time of the flow f in order to exploit the delay sensitivity of the receiver and characterize the delay-throughput trade-offs.

We consider the ℓ_p-norm of the sequence of i.i.d Geometric random variables $\mathbf{\Delta T}=(\Delta T_1,\dots,\Delta T_N)$, i.e., $\mathbb{E}[\|\mathbf{\Delta T}\|_p]$, hence define the delay cost function as

$$\tilde{d}(p)=\frac{1}{LN^{\frac{1}{p}}}\mathbb{E}\left[\Big(\sum_{i=1}^{N}\Delta T_i^p\Big)^{1/p}\right], \quad p\in[1,\infty), \quad (2)$$

where p models the delay sensitivity of the receiver. This metric is determined by the type of the applications running on the receiver. For example, if a user is downloading a file, it might be more concerned about shortening the overall completion time. However, if the user is running a real-time video application, then it might be more sensitive to the maximum inter-arrival time between any two packets [5].

When $p=1$, we have $\tilde{d}(1)=\frac{1}{LN}\sum_{i=1}^{N}\mathbb{E}[\Delta T_i]$ that is the *average delay per packet*, normalized by the total size of the received data. Hence, minimizing $\tilde{d}(1)$ is equivalent to maximizing the throughput $\tilde{d}(1)^{-1}$ of the system. When $p=\infty$, we have $\tilde{d}(\infty)=\frac{1}{L}\mathbb{E}[\max_{i=1,\dots,N}\Delta T_i]$, which is the maximum expected inter-arrival time between any two successive packets or the *per-packet delay*. Hence, minimizing $\tilde{d}(\infty)$ is equivalent to minimizing per-packet delay.

While it is possible to show that the moment generating functions of $\mathbf{\Delta T}$ exist, it is nontrivial to derive a closed form expression for the value of $\mathbb{E}[\|\mathbf{\Delta T}\|_p]$ for $p>1$. Therefore, we next find a lower and and upper bound for the term $\tilde{d}(\infty)$.

Proposition 1. *A lower bound for $\tilde{d}(\infty)$ is given as*

$$\tilde{d}(\infty)\geq\frac{1}{L}\max_{i=1,\dots,N}\mathbb{E}[\Delta T_i]=\frac{1}{L}\Big(\frac{K}{r}+D\Big).$$

An upper bound for $\tilde{d}(\infty)$ is given as

$$\tilde{d}(\infty)\leq(Nm!)^{1/m}\Big/(L\log(1/(1-r))),\quad m\in\mathbb{N}. \tag{3}$$

Proof. The lower bound is due to the Jensen's inequality and (1). For the upper bound, see Appendix A. □

Consider the delay cost function which is defined in [5] as

$$d(p)=\frac{1}{L}\Big(\frac{1}{N}\sum_{i=1}^{N}(\mathbb{E}[\Delta T_i])^p\Big)^{1/p},\quad p\in[1,\infty). \tag{4}$$

When $p=1$, $d(1)=\frac{1}{LN}\sum\limits_{i=1}^{N}\mathbb{E}[\Delta T_i]$, which is the same as the *average delay per packet* $\tilde{d}(1)$. When $p=\infty$, $d(\infty)=\frac{1}{L}\max\limits_{i=1,\dots,N}\mathbb{E}[\Delta T_i]$, which is indeed a lower bound to the *per-packet delay* $\tilde{d}(\infty)$ that is due to Jensen's inequality.

Using $d(1)$, the bucket size from (4) is computed as

$$K=Dr/\big(rLd(1)-1\big). \tag{5}$$

If the adaptive scheme chooses a bucket size of K for a flow of N packets, the delay cost in (4) is simplified as [5]:

$$d(p)=\frac{1}{L}\Big(\frac{1}{K}\sum_{j=1}^{K}(\mathbb{E}[\Delta T_{g_j}])^p\Big)^{1/p}=\frac{\frac{K}{r}+D}{LK^{1/p}}, \tag{6}$$

where we assume that K is in the region $[1,K_{\max}]$, $K_{\max}$ being the maximum bucket size. This limiting assumption is justifiable through the maximum computation complexity that can be handled by the target system. Inserting (5) into (6), the trade-off between $d(1)$ and $d(\infty)$ is expressed as

$$d(\infty)=D\Big/\big(L-\frac{1}{d(1)r}\big). \tag{7}$$

The optimal block size K^* that minimizes $d(p)$ for the point-to-point link model is obtained as follows:

$$K^*=(rD/(p-1))\Big|_{[1,K_{\max}]},\quad 0<\epsilon<1, \tag{8}$$

where $(x)_{[1,K_{\max}]}\triangleq\min(\max(1,x),K_{\max})$.

Let $S_g(K,r)=\sum_{j=1}^{K}\Delta T_{g_j}$ be the sum of ordered inter-arrival times of packets in bucket g. This is a negative Binomial random variable when $D=0$, and its average value is $\mathbb{E}[S_g(K,r)]=\frac{K}{r}$, as also given in (1). Let $\hat{S}=\max\limits_{g=1,\dots,M}S_g(K,r)$, which is the maximum of ordered inter-arrival times of all generations in the flow f. To the best of our knowledge, we can only compute bounds on $\mathbb{E}[\hat{S}]$.

We are interested in the regime where $d(\infty)$ can well approximate $\tilde{d}(\infty)$, i.e. $\mathbb{E}\big[\max\limits_{i=1,\dots,N}\Delta T_i\big]\approx\max\limits_{i=1,\dots,N}\mathbb{E}[\Delta T_i]$. We next provide a necessary condition for that:

Proposition 2. *The tail probability of $S_g(K,r)$ satisfies*

$$\mathbb{P}[S_g(K,r)>lK/r]\leq\exp\big(-(1-1/l)^2lK/2\big),\quad l\geq 1.$$

Proof. See Appendix B. □

From Prop. 2, when the bucket size K is large enough, ordered inter-arrival time of a bucket concentrates around its mean. Therefore, the tail probability of $\hat{S}$ satisfies

$$\mathbb{P}\Big[\hat{S}>lK/r\Big]\leq 1-\Big(1-\exp\big(-(1-1/l)^2lK/2\big)\Big)^M,$$

where $l\geq 1$. Similar to $S_g(K,r)$, $\hat{S}$ concentrates around its mean by choosing the bucket size K sufficiently large. Sect. IV validates our approximation.

We next propose an adaptive coding scheme for multi-hop line networks, which is an extension of the point-to-point model in [13]. The sender (Tx) and the receiver (Rx) are connected by wireless erasure channels with independent packet erasure probabilities and perfect feedback channels with constant delay at each hop from the receiver to the sender.

III. Multi-hop Line Networks

In this section, we consider different coded transmission and acknowledgment schemes for modeling multi-hop line networks. We analyze their delay-throughput tradeoffs by exploiting the sensitivity model summarized in Sect. II.

The end-to-end and recoded multi-hop line networks are depicted in Figs. 2-(a) and 2-(b), respectively. The network consists of H links in tandem, and $H+1$ nodes h, $0\leq h\leq H-1$ is connected to node $h+1$ through the $h+1^{\text{th}}$ erasure link, where nodes 0 and H denote the sender Tx and the receiver Rx, respectively. The probability of transmission failure, i.e. the erasure rate, on link h equals ϵ_h. We assume that the erasure of each link is independent of the others.

We assume the average delay per hop is $\bar{D}=\mathbb{E}[D]$. Given the number of hops H, and the delay per hop $D(h)$, $h\in\{1,\dots,H\}$, denote the total delay of the feedback by $D_H=\sum_{h=1}^{H}D(h)$, and its average value by $\bar{D}_H=\mathbb{E}\left[D_H\right]=\bar{D}H$. The bucket size for each transmitter node h, $h\in\{0,\dots,H-1\}$ is also the same, and is denoted by K. We next detail the random linear coding model for multi-hop links.

A. A Primer on Random Linear Coding

Consider a network consisting of H links in tandem. Assume that the edges have capacities $1-\epsilon_h$, $h\in\{1,\dots,H\}$ as shown in Fig. 2-(b). The achievable rate for a coding scheme is bounded by the cut-set bound. Although the cut-set bounds are not achievable in general[1], the tandem network

[1]For general multi-terminal networks, the converse on the rate of information flow across any cut-set is proven [16, Ch. 15.10].

admits a simple max-flow min-cut interpretation. From the Ford-Fulkerson theorem [17], this capacity can be achieved.

The coding scheme is run for a total duration such that a flow f of N packets, i.e. $\mathcal{V} = \{v_1, v_2, \dots v_N\}$, is received by node 1 within this time. Any received packet $V[t]$ in the network is a linear combination of the packets in $\mathcal{V}$. Therefore, $V[t] = \sum_{j=1}^{N} b_j[t]v_j$, where $b_j[t]$ associated with $V[t]$ is the auxiliary encoding vector of $V[t]$ [2]. Since v_j is a random linear combination of the message packets $\{P_{g_1}^f, P_{g_2}^f, \dots, P_{g_K}^f\}$, we have $v_j = \sum_{k=1}^{K} a_{jk}[t]P_{g_k}^f$ for $j = 1, \dots, N$, where each $a_{jk}[t]$ is uniformly chosen at random over $\mathbb{F}_q$. Hence,

$$V[t] = \sum_{k=1}^{K} \Big(\sum_{j=1}^{N} b_j[t]a_{jk}[t] \Big) P_{g_k}^f, \tag{9}$$

and the k^{th} component of the global encoding vector of $V[t]$ is given by $\gamma_k[t] = \sum_{j=1}^{M} b_j[t]a_{jk}[t]$.

When a node receives $\lceil K(1+\delta) \rceil$, for small $\delta > 0$, with linearly independent auxiliary encoding vectors, the received packets along with the encoding vectors form a random $\lceil K(1+\delta) \rceil \times K$ matrix uniformly over $\mathbb{F}_q$. The message packets can be recovered if this matrix has rank K. Hence, node 2 can recover the message packets, as long as it receives $\lceil K(1+\delta) \rceil$ packets with linearly independent encoding vectors. This is possible by taking K arbitrarily large.

Packets are innovative if they carry new information about $\mathcal{V}$. It has been proven in [2] that the propagation of innovative packets through a network follows the propagation of jobs through a queuing network, for which fluid models give good approximations. The innovative packets arriving at node 1 at a rate of r_1 can be approximated by fluid flowing in at rate r_1. Similarly, r_{h+1} denotes the rate of the fluid flowing out of node h to $h+1$. From this fluid analogy, the maximum flow from Tx and Rx can be computed from the Ford-Fulkerson theorem. Hence, a connection of rate arbitrarily close to R packets per unit time from Tx to Rx can be established provided that

$$R \leq \min_{1 \leq h \leq H} r_h. \tag{10}$$

The time required for node 2 to receive $\lceil K(1+\delta) \rceil$ packets can be approximated as $K(1+\delta)/\min\{r_1, r_2\}$. Mapping the 2-hop model to a network consisting of H hops (see Fig. 2-(b)), the time required for node H (Rx) to receive $\lceil K(1+\delta) \rceil$ packets is approximated as $K(1+\delta)/\min_{1 \leq h \leq H} r_h$. Hence, a rate of R from Tx to Rx satisfying (10) is achievable.

For the multi-hop line network scenario and onward we assume that each node has the same coding bucket size K. For a given receiver sensitivity p, nodes jointly decide the optimal bucket size K^* that minimizes the delay cost function $d(p)$ of the multi-hop line network. This is detailed for different encoding and feedback schemes in Sects. III-B-III-D. For large K, this scheme achieves the capacity [2]. While it is possible to generalize the recoding scheme by associating each intermediate node h with a distinct bucket size that optimizes the delay sensitivity of node $h+1$, this generalization is indeed non-trivial and is not considered in the current paper.

We next detail different encoding and feedback schemes for multi-hop networks and outline the trade-offs between them.

B. Recoded Scheme with End-to-End ACK

In this scheme, since the sender codes and each intermediate node recodes, from the min-cut max-flow theorem, the rate at which the coded packets is received equals $\mathrm{r}_{\mathrm{c}}^{\mathrm{e2e}} = \min_{h=1,\dots,H} (1-\epsilon_h)$. In this recoded scheme, intermediate nodes do not decode in between. Instead, they recode-and-forward, i.e. mix packets and forward them. Effectively, due to the random linear coding scheme, any new packet each hop receives adds some new information, and each hop mixes the K packets it has together. This implies recoding on the fly.

The value of $d(p)$ for the recoded scheme is computed as

$$d(p) = \frac{1}{LK^{1/p}} \left(\frac{K}{\mathrm{r}_{\mathrm{c}}^{\mathrm{e2e}}} + \bar{D}_H \right). \tag{11}$$

From (5), the bucket size is given as

$$K = \bar{D}_H \Big/ \Big(Ld(1) - \frac{1}{\mathrm{r}_{\mathrm{c}}^{\mathrm{e2e}}} \Big). \tag{12}$$

From (11) and (12), the maximum per-packet delay is

$$d(\infty) = \bar{D}_H \Big/ \Big(L - \frac{1}{d(1)\,\mathrm{r}_{\mathrm{c}}^{\mathrm{e2e}}} \Big), \tag{13}$$

from which we observe that the maximum per-packet delay scales with the number of hops, and the tradeoff between $d(\infty)$ and $d(1)$ is sharper than the point-to-point case in (7).

C. End-to-End Coded Scheme with End-to-End ACK

In this scheme the sender codes and no recoding is involved at intermediate nodes. Therefore, the links can be treated independently of each other. The network may be considered as a point-to-point link with an effective rate of $\mathrm{r}^{\mathrm{e2e}} = \prod_{h=1}^{H} (1-\epsilon_h)$, which is clearly bounded above by the cut-set bound. Hence, the scheme may not achieve the capacity. From (5), the bucket size is given as

$$K = \bar{D}_H \Big/ \Big(Ld(1) - \frac{1}{\mathrm{r}^{\mathrm{e2e}}} \Big). \tag{14}$$

Given the total delay of the feedback D_H, by exploiting the trade-off between $d(1)$ and $d(\infty)$ given in (7), the maximum per-packet delay is determined as

$$d(\infty) = \bar{D}_H \Big/ \Big(L - \frac{1}{d(1)\,\mathrm{r}^{\mathrm{e2e}}} \Big). \tag{15}$$

Comparing (15) with (13), since the effective rate is much smaller for the end-to-end coded case, the trade-off between $d(\infty)$ and $d(1)$ is sharper for the end-to-end coded model.

D. (Totally) Uncoded Scheme with Link-by-Link ACK

In this uncoded scheme, intermediate nodes forward the incoming packets one by one, and every packet received by any intermediate node or the receiver is innovative. Since the propagation of innovative packets are well approximated by fluid flow models, from the max-flow min-cut theorem, the rate at which the packets are received is $\mathrm{r}_{\mathrm{u}}^{\mathrm{l2l}} = \min_{h=1,\dots,H} (1-\epsilon_h)$.

The given scheme is uncoded. Hence, the value of the bucket size K does not play a role in the value of $d(1)$.

The average ordered inter-arrival time of the i^{th} packet is

$$\mathbb{E}[\Delta T_i] = 1/\,\mathrm{r}_{\mathrm{u}}^{\mathrm{l2l}} + \bar{D}_H, \quad \forall i = 1, \dots, N.$$

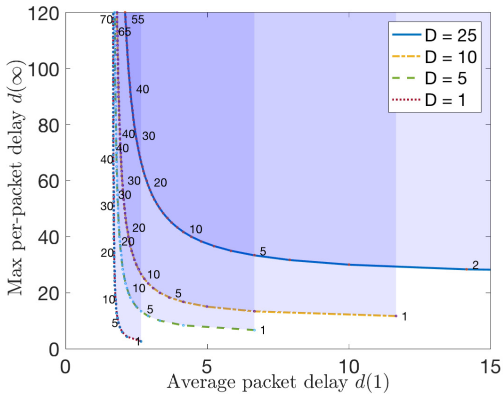

Fig. 3: $d(\infty)$ vs $d(1)$ with varying D in Fig. 1. Single hop with coding where $\epsilon = 0.4$. Optimal bucket size K^* is also marked for several achievable delay pairs $(d(1), d(\infty))$.

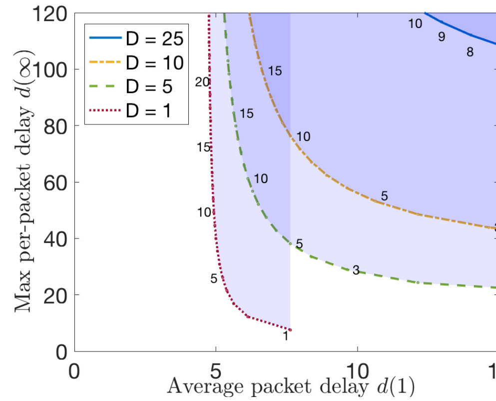

Fig. 4: $d(\infty)$ vs $d(1)$ with varying D in Fig. 2-(a). Three hops with end-to-end coding with end-to-end ACK, where $\boldsymbol{\epsilon} = [0.4,\, 0.6,\, 0.1]^\intercal$. The values K^* are marked for several $(d(1), d(\infty))$.

The value of $d(p)$ for the uncoded scheme with link-by-link acknowledgment is computed as

$$d(p) = \frac{1}{LN^{1/p}} \left(\frac{N}{K} (K/\, \mathrm{r}_{\mathrm{u}}^{\mathrm{l2l}} + K\bar{D}_H)^p \right)^{1/p}. \qquad (16)$$

The average packet delay is $d(1) = \frac{1}{L}\left(1/\,\mathrm{r}_{\mathrm{u}}^{\mathrm{l2l}} + \bar{D}_H\right)$. Evaluating (16) at $p = \infty$, the bucket size is given as

$$K = Ld(\infty) \Big/ \left(1/\,\mathrm{r}_{\mathrm{u}}^{\mathrm{l2l}} + \bar{D}_H\right).$$

Hence, the maximum per-packet delay scales linearly with K:

$$d(\infty) = \frac{K}{L}(1/\,\mathrm{r}_{\mathrm{u}}^{\mathrm{l2l}} + \bar{D}_H). \qquad (17)$$

Since this scheme does not involve coding, its maximum delay can be minimized by choosing $K = 1$. Unlike coded schemes, we cannot observe a trade-off between $d(\infty)$ and $d(1)$.

IV. Numerical Results

We numerically demonstrate the performance of the different multi-hop schemes proposed in Sect. III, and contrast with the uncoded scheme of [5]. In Figs. 3-6, we investigate the trade-off between the average packet delay $d(1)$ (inverse of throughput) versus maximum per-packet delay $d(\infty)$ given the optimal bucket size K^*, for various values of D and the erasure probabilities $\boldsymbol{\epsilon} = [0.4,\, 0.6,\, 0.1]^\intercal$.

The region lower bounded by each curve is the area of all achievable delay pairs $(d(1), d(\infty))$ normalized by L, for the specific feedback ACK delay D. With small D, low values of $(d(1),\ d(\infty))$ can be achieved. However, as D increases, the trade-off becomes increasingly stronger. This is evident from (7), where D is in the numerator. In the plots, we explicitly show the overlaps for the jointly achievable delay pairs for different D. As the trade-off becomes stronger with increasing D, the overlap amount decreases. We also mark the optimal bucket size K^* for several achievable delay pairs $(d(1), d(\infty))$. Note that K^* increases with $d(\infty)$, which follows from (6). As provided in Prop. 2, when K is large enough, i.e $d(1)$ is small, $\tilde{d}(\infty) \approx d(\infty)$. Given a per-packet delay $d(\infty)$, a multi-hop scheme performs better than the others in terms of throughput if the achievable $d(1)$ is smaller.

In Figs. 3 and 4, we illustrate the trade-off of $d(1)$ vs $d(\infty)$ with varying D, for the single hop of scheme of [5], and the *end-to-end coded scheme with end-to-end acknowledgment* with $H = 3$ hops, respectively. In this model, the sender codes but there is no coding at the intermediate nodes.

In Fig. 5, we illustrate the trade-off of $d(1)$ vs $d(\infty)$ with varying D for the *recoded scheme with end-to-end ACK* (Sect. III-B). We can observe that with recoding using multiple hops, it is possible to have significant reductions in $d(\infty)$ at the cost of increasing $d(1)$. In Fig. 6, we compare the performance of the single hop scheme and two different multi-hop schemes for $D = 5$ with $H = 3$. In the *recoded scheme with end-to-end ACK*, the sender codes and each node recodes. In the *end-to-end coded scheme with end-to-end ACK* of Sect. III-C, the sender codes and the intermediate nodes are relays and do not recode. Therefore, we can observe that the recoded scheme provides a better trade-off than end-to-end coded scheme. By taking into account the additional feedback delay due to 3 hops, we can observe that for a given $d(1)$, the recoded scheme provides a better scaling in $d(\infty)$ than the single hop model.

We observe from Figs. 3-6 that with coding, the trade-off between throughput and average delay can be improved, and the extra delay caused by multiple hops can be compensated.

Comparing the schemes in Sect. III, we can observe from Figs. 3, 4, 5, and 6 that with recoding, the trade-off between throughput and average delay can be improved, and the additional delay caused by multiple hops can be compensated.

V. Conclusion

We study throughput and delay trade-offs in multi-hop erasure channels, with different encoding and feedback schemes. To do so, we devise an ℓ_p-norm delay metric that exploits the sensitivity of the receiver. We show that with adaptively adjusting the coding bucket size based on receiver's sensitivity and the feedback delay, recoded multi-hop transmissions can increase the throughput of the end-to-end coded transmissions by 30%. Our numerical results demonstrate that through the feedback acknowledgment scheme, the adaptive coding and scheduling scheme is very robust to the variations of the delay

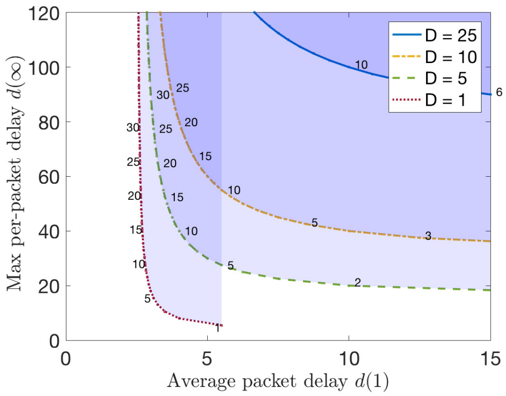

Fig. 5: $d(\infty)$ vs $d(1)$ with varying D in Fig. 2-(b). Three hops with recoding and end-to-end ACK $\boldsymbol{\epsilon} = [0.4,\ 0.6,\ 0.1]^\intercal$. The values K^* are marked for several $(d(1), d(\infty))$.

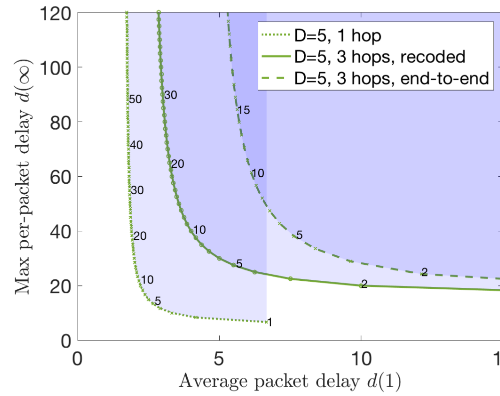

Fig. 6: $d(\infty)$ vs $d(1)$ for $D = 5$. Comparison of end-to-end schemes in Figs. 1 and 2-(a) and recoded scheme in Fig. 2-(b), where $\boldsymbol{\epsilon} = [0.4,\ 0.6,\ 0.1]^\intercal$. The values K^* are marked for several $(d(1), d(\infty))$.

sensitivity of the receiver. We believe that the current work paves the way for modeling and analyzing the throughput and delay trade-offs for the full mesh network with a diverse set of connections, and more general directed acyclic networks. Interesting research directions are to consider the queueing delay for the heavy traffic regime, and exploit cross-path coding to reduce delay. Possible future directions also include the characterization of how the performance scales with the network size, and the design of multi-path packet scheduling algorithms to minimize the in-order delivery time.

Appendix

A. Proof of Proposition 1

Lemma 1. *If X is geometrically distributed with parameter r, its moments satisfy $\mathbb{E}[X^m] \le \frac{m!}{(\log(1/(1-r)))^m}$ for $m \in \mathbb{N}$.*

Proof. If Y is exponentially distributed with parameter λ, then $X = \lfloor Y \rfloor$ is a geometrically distributed random variable with parameter $1 - \exp(-\lambda)$. Since $X \le Y$, $\mathbb{E}[X^m] \le \mathbb{E}[Y^m] = m!/\lambda^m$. Letting $r = 1 - \exp(-\lambda)$ we obtain the bound. □

We can upper bound $\tilde{d}(\infty)$ as

$$\tilde{d}(\infty) \overset{(a)}{\le} \frac{1}{L}\mathbb{E}\left[\Big(\sum_{i=1}^{N} \Delta T_i^m\Big)^{1/m}\right] \overset{(b)}{\le} \frac{1}{L}\Big(\sum_{i=1}^{N} \mathbb{E}[\Delta T_i^m]\Big)^{1/m},$$

where (a) follows from that $\mathbb{E}[\|\mathbf{x}\|_\infty] \le \mathbb{E}\Big[\Big(\sum_{i=1}^N x_i^m\Big)^{1/m}\Big]$ because $\|\mathbf{x}\|_m \ge \|\mathbf{x}\|_p$ when $0 < m < p$, and $x_i > 0$; and (b) from that $f(X) = X^{1/m}$ is concave when $m > 1$ and $X > 0$. Hence, inserting $X = \sum_{i=1}^N \Delta T_i^m$ and from Jensen's inequality, we have $\mathbb{E}[f(X)] \le f(\mathbb{E}[X])$.

Using this upper bound and Lemma 1, without computing the higher order moments of ΔT_i, we can upper bound $\tilde{d}(\infty)$ as in (3), which is easily computed for $m > 2$.

B. Proof of Proposition 2

Lemma 2. *Let $X = \sum_{i \in [n]} X_i$ where X_i, $i \in [n]$ are independently distributed in $[0, 1]$. Then for $0 < \epsilon < 1$, we have the following Chernoff-Hoeffding bound:*

$$\mathbb{P}[X < (1-\epsilon)\mathbb{E}[X]] \le \exp\Big(-\epsilon^2 \mathbb{E}[X]/2\Big).$$

Proof. See [18, Theorem 1.1]. □

Lemma 3. *Let $B(n, r)$ be a Binomial random variable with success probability r. We then have the following relation:*

$$\mathbb{P}[B(n,r) \ge K] = \mathbb{P}[S_g(K,r) \le n].$$

Proof. See [18, Chapter 3.3]. □

From Lemmas 2-3, and letting $\epsilon = 1 - 1/l$ for $l \ge 1$, we have that $\mathbb{P}[S_g(K,r) > lK/r] = \mathbb{P}[B(lK/r, r) < K] \le \exp\big(-(1-1/l)^2 lK/2\big)$.

References

[1] D. S. Lun *et al.*, "Achieving minimum-cost multicast: A decentralized approach based on network coding," in *Proc. IEEE Infocom*, vol. 3, Mar. 2005, pp. 1607–17.

[2] ——, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 1, pp. 3–20, Mar. 2008.

[3] A. Eryilmaz, A. Ozdaglar, and M. Médard, "On delay performance gains from network coding," in *Proc., IEEE Annu. Conf. Info. Sciences and Systems*, Mar. 2006, pp. 864–870.

[4] D. Lun *et al.*, "An analysis of finite-memory random linear coding on packet streams," in *Proc., IEEE WiOpt*, Apr. 2006.

[5] W. Zeng, C. T. K. Ng, and M. Médard, "Joint coding and scheduling optimization in wireless systems with varying delay sensitivities." in *Proc., IEEE Commun. Soc. Conf. on Sensor, Mesh and Ad Hoc Commun. and Netw. (SECON)*, Jun. 2012, pp. 416–424.

[6] J. Ishmael *et al.*, "Deploying rural community wireless mesh networks," *IEEE Internet Computing*, vol. 12, no. 4, Jul. 2008.

[7] A. Adya *et al.*, "A multi-radio unification protocol for IEEE 802.11 wireless networks," in *Proc., IEEE Intl. Conf. Broadband Networks*, Oct. 2004, pp. 344–354.

[8] Z. Zaidi and K. C. Lan, "Wireless multihop backhauls for rural areas: A preliminary study," *PloS one.*, vol. 12, no. 4, Apr. 2017.

[9] S. Othmen *et al.*, "Secure and reliable multi-path routing protocol for multi-hop wireless networks," *Adhoc and Sensor Wireless Networks*, vol. 36, 2017.

[10] M. M. Mahmoud, X. Lin, and X. S. Shen, "Secure and reliable routing protocols for heterogeneous multihop wireless networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 26, no. 4, pp. 1140–53, Apr. 2015.

[11] A. Özgür, O. Lévêque, and D. Tse, "Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks," *IEEE Transactions on information theory*, vol. 53, no. 10, pp. 3549–3572, 2007.

[12] J. Ghaderi, L. L. Xie, and X. Shen, "Hierarchical cooperation in ad hoc networks: Optimal clustering and achievable throughput," *IEEE Trans. Info. Theory*, vol. 55, no. 8, pp. 3425–3436, Aug. 2009.

[13] T. Ho *et al.*, "A random linear network coding approach to multicast," *IEEE Trans. Info. Theory*, vol. 52, no. 10, pp. 4413–30, Oct. 2006.

[14] D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[15] A. Eryilmaz, A. Ozdaglar, M. Médard, and E. Ahmed, "On the delay and throughput gains of coding in unreliable networks," *IEEE Trans. Inf. Theory*, vol. 54, no. 12, pp. 5511–24, Dec. 2008.

[16] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY: John Wiley & Sons: John Wiley & Sons, 2006.

[17] L. R. Ford and D. R. Fulkerson, *Flows in Networks*. Princeton, NJ: Princeton University Press, 1962.

[18] D. Dubhashi and A. Panconesi, *Concentration of measure for the analysis of randomized algorithms*. Cambridge University Press, 2009.