Optimal Rate–Delay Tradeoffs and Delay Mitigating Codes for Multipath Routed and Network Coded Networks

John MacLaren Walsh, *Member, IEEE*, Steven Weber, *Member, IEEE*, and Ciira wa Maina, *Student Member, IEEE*

Abstract—Via multiterminal information theory, a framework is presented for deriving fundamental rate–delay tradeoffs that delay mitigating codes must have when utilized over multipath routed and random linear network coded networks. The rate–delay tradeoff is formulated as a calculus problem on a capacity region of a related abstracted broadcast channel. Given this general framework for studying such rate–delay tradeoffs, the extreme case of uniform networks, in which each possible received packet arrival order is equally likely, is considered. For these networks, the rate–delay calculus problem is simplified to an integer programming problem, which for small numbers of packets may be solved explicitly, or for larger numbers of packets, may be accurately approximated through the calculus of variations by appropriate relaxation of an integer constraint. Explicit expressions for the rate–delay tradeoff in uniform networks are presented in the special cases of i) constant packet inter-arrival times, and ii) exponential independent and identically distributed (i.i.d.) packet arrival times. Finally, the delay mitigating codes achieving these rate–delay tradeoffs are discussed.

Index Terms—Connection-oriented service, delay-mitigating codes, multipath routing, network coding, rate–delay tradeoffs.

I. INTRODUCTION

A. Problem Statement

CONSIDER a multipath routed network in which a particular source hands M packets $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M$ (each K bits long) to the lower network layers for transmission at time 0. These M packets are all headed to the same destination, but, because of queueing and propagation delays along the different routes, the packets $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M)$ arrive at the sink at different time instants $(t_1, \ldots, t_M)$, possibly giving a different order. We will model the source-to-destination effects of the multipath routed network transmission, then, as selecting these arrival times $t_1, \ldots, t_M$ randomly according to some joint distribution $\mathsf{p}(t_1, \ldots, t_M)$. We assume packets are not lost in order to focus our attention on the impact of packet reordering and delay (see Section VI).

Denote by $\pi(i)$ the index of the transmitted packet that is the ith received packet, so that the packets arriving at the receiver in order are $\boldsymbol{x}_{\pi(1)}, \boldsymbol{x}_{\pi(2)}, \ldots, \boldsymbol{x}_{\pi(M)}$, with[1] $t_{\pi(1)} < t_{\pi(2)} < \cdots < t_{\pi(M)}$, which we denote by $\tau_i = t_{\pi(i)}$. We can then think of the probability distribution $\mathsf{p}(t_1, \ldots, t_M)$ on the arrival times as being composed of a probability distribution $\mathsf{p}(\pi)$ on the arrival order and a probability distribution on the ordered arrival instants given the arrival order $\mathsf{p}(\tau_1, \ldots, \tau_M | \pi)$. This paper investigates in particular the case where the ordered arrival instants are independent of the arrival order, so that $\mathsf{p}(\tau_1, \ldots, \tau_M | \pi) = \mathsf{p}(\tau_1, \ldots, \tau_M)$. Such independence occurs, for example, when the packet arrival times $t_1, \ldots, t_M$ are independent and identically distributed (i.i.d.).

Suppose further that the source node wishes to use the packets $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M$ to convey some temporally ordered data to the receiver. For instance, the source node may wish to convey a digital multimedia file that is the output of a multimedia source encoder which, after compressing the source, returns data organized into temporally ordered frames $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$ corresponding to successive chunks of time from the multimedia signal. Assuming that the source encoder has done a good job of compressing, the data in $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$ will be independent. Alternatively, the source node may be controlling the destination node remotely through the network, with a sequence of temporally ordered instruction frames $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$ that must be executed in order. As another possibility, the source node may be engaged in a voice over Internet Protocol (VoIP) conversation with the receiver. Indeed, almost any connection-oriented network source application could be considered as relevant for what we are about to discuss.

All of these temporally ordered source signals share the common characteristic that later data (i.e., with higher frame indices) is not useful until earlier data (i.e., with lower frame indices) has been received. However, since the effective point-to-point channel that the multipath routed network creates reorders packets, were we simply to transmit the source data directly as is over the network channel, we would have to wait at the receiver, storing packets in a buffer, and reordering them as we played them out. Alternatively, one could consider a single-source multicast network coded network

[1]Since we are using continuous time, and thus continuously distributed arrival instants, the probability that two or more of the arrival instants are equal must be zero, so we ignore that case in our discussion.
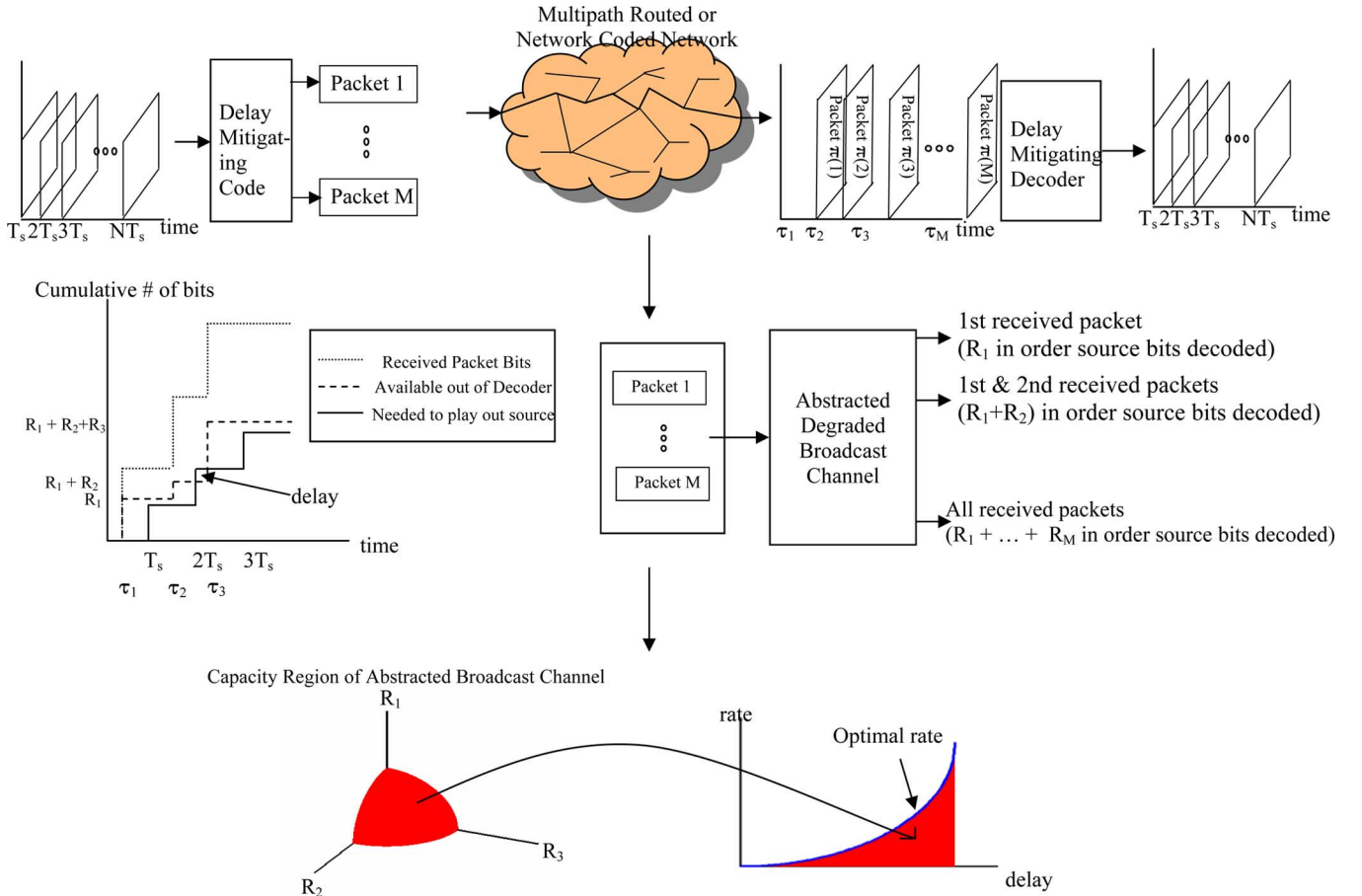
Fig. 1.   The problem and the approach we will use to solve it.

employing random linear network coding [1], [2]. In this case $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M)$ arriving at the sink will be random linear combinations of original transmitted packets (letting M be the dimension of the global encoding vector), and we will have to wait, buffering the received packets, until all M innovative (i.e., with linearly independent encoding vectors) packets have been received before we can start decoding even the earliest source frame $\boldsymbol{s}_1$. Here, we will model the innovative network encoded packets as arriving at times $t_1, \ldots, t_M$, giving again ordered arrival times $\tau_1 < \cdots < \tau_M$.

While buffering has been extensively studied as a technique for connection-oriented transmission over current networks such as the Internet, we note that routing data from the same temporally ordered flow simultaneously along multiple paths and/or employing network coding for the sake of greater reliability and speed opens up a host of packet and information reordering phenomena. Buffering techniques designed for networks such as the present internet which generally maintain a first-in first-out (FIFO) delivery policy to the same packet flow [3], [4] may thus not be as appropriate for these situations. Thus, for these multipath routed and network coded networks we wish to consider an alternative to buffering: an approach based on coding. In particular, we propose to use a linear *delay mitigating code* to create the data in the transmitted packets from the source packets in such a way as to minimize the delay incurred while playing the source out at the receiver. This approach is to be contrasted with classical techniques that code

for packet erasures with erasure correcting codes, where the focus is on delivering reliable transmission of a large block after a large delay. Our focus, instead, is to study the tradeoffs between rate and delay on these channels in a more concrete manner than allowed by, e.g., an error exponent based study.

B. Solution Techniques

To achieve the goal of exhaustively characterizing and designing the best such delay mitigating codes, we use the following **four-step technique**, depicted in Fig. 1.

Step 1) Find the capacity region of the associated abstracted broadcast channel. Define an abstracted broadcast channel matching the packet reordering and combining characteristics of the network in question, in which different receivers correspond to the cumulative set of packet observations after different successive packet arrivals at the sink. Recognize this channel as a degraded broadcast channel, and calculate its capacity region $\mathcal{R}$, the general form of which is known [5]–[7]. This capacity region dictates the set of possible rate vectors $\boldsymbol{r} = (R_1, \ldots, R_M)$ over all possible delay mitigating codes. Here, each R_i indicates the amount of new information that may be decoded upon each successive packet arrival at the sink (Section III).

Step 2) Define the rate and delay. Define the overall rate $\rho(\boldsymbol{r})$ and overall delay $\delta(\boldsymbol{r})$ for a particular delay mitigating coding scheme in terms of the rate vector $\boldsymbol{r}$ (Section IV-A). The overall rate will reflect the number of bits per second that will

be delivered by the connection. The connection will provide this constant bitrate data in the form of a fixed number of new bits $\rho(\boldsymbol{r})T_s$ delivered every T_s seconds for NT_s seconds. The chosen delay metric should reflect any deviation from this information delivery schedule.

Step 3) Determine the fundamental rate–delay tradeoff. Find the optimal tradeoff between rate and delay by maximizing the overall rate $\rho(\boldsymbol{r})$ subject to a delay constraint $\delta(\boldsymbol{r}) \leq d$ over all feasible rate vectors $\boldsymbol{r}$ in the capacity region $\mathcal{R}$

$$\rho_d^* := \max_{\boldsymbol{r} \in \mathcal{R}} \{\rho(\boldsymbol{r}) | \delta(\boldsymbol{r}) \leq d\}. \tag{1}$$

An optimal delay-mitigating code is found as a capacity-achieving code from the previous step with rate vector

$$\boldsymbol{r}_d^* \in \arg \max_{\boldsymbol{r} \in \mathcal{R}} \{\rho(\boldsymbol{r}) | \delta(\boldsymbol{r}) \leq d\}. \tag{2}$$

Our work also investigates a variational calculus approach for calculating in closed form the solutions to the rate–delay tradeoff problem (1) in the limit of large numbers of packets. This amounts to approximating the rate vector, after appropriate scaling in the limit, as a continuous function mapping $[0,1]$ to $[0,1]$. The rate and delay are then normalized to each lie within $[0,1]$ and are approximated as functionals. In some cases, this approximation can be aided by the theory of order statistics. The rate–delay tradeoff programming problem then becomes a calculus of variations problem that we have solved in some special cases (Sections IV-C, IV-D, IV-E).

Step 4) Determine capacity-achieving codes for the abstracted degraded broadcast channel. These codes have rate vectors on the boundary of the capacity region $\mathcal{R}$. Despite the asymptotic block lengths employed in proving the degraded broadcast channel capacity region (which defeat the whole notion of bounding delay), surprisingly there exist codes block length of one (i.e., a single broadcast channel use) exhibiting the desired rate–delay tradeoffs (Section V) in the uniform case which is studied in this paper. In Section V, capacity-achieving codes in the uniform case are shown to be priority encoded transmission (PET) [8], [9] codes with an optimized collection of component code rates selected through the introduced framework. In other nonuniform cases, classes of codes other than PET codes are capacity achieving, but the rate–delay tradeoffs we provide still hold as bounds.

C. Overview of Main Results

Section II discusses related work on error correction, fountain codes, priority encoded transmission, and error exponents. Section III gives capacity regions for two degraded broadcast channels modeling packet reordering with delays and network coded networks, respectively. Theorem 2 gives the rate region for the uniform permutation channel, i.e., when all packet reordering permutations are equally likely. Theorem 3 gives the analogous rate region for the uniform network coded channel, i.e., when the encoding matrix is selected uniformly at random over all matrices of given rank. In both cases, the rate region is those vectors $\boldsymbol{r} = (R_1, \ldots, R_M)$ such that $\sum_{i=1}^{M} R_i/i \leq K$. Section IV presents the fundamental rate–delay tradeoffs. While the presented rate–delay tradeoff framework holds for more general situations, the remainder of the paper focusses on the uni-

form permutation and uniform network coded channel cases, which may be considered to be extreme cases of the more general problem. Theorem 4 gives the structural properties of the rate–delay tradeoff in this uniform case, emphasizing that the problem can be parameterized in the decoding deadlines indicating the number of packets that must be received before decoding each frame. Because the decoding deadlines are integer valued, this result transforms a continuous nonlinear program into an integer program, making the problem amenable to exact solution for small M, N. Then, Theorem 5 discusses a relaxation of the rate–delay tradeoff problem via variational calculus. The integer-valued decoding deadline sequence is relaxed to a continuous function mapping $[0,1] \rightarrow [0,1]$. This relaxation gives a tractable solution which is a reasonable approximation of the optimal rate–delay tradeoff for large M, N. Theorem 6 utilizes the approximation from the previous theorem to give an algebraic expression for the rate–delay tradeoff in the special case of constant inter packet arrival times. Similarly, Theorem 7 gives an algebraic expression for the special case of i.i.d. exponential packet arrival times. In order to demonstrate the feasibility of the results, Section V discusses practical small block-length codes achieving the rate–delay tradeoffs. Section VI shows that although the rate–delay tradeoffs are obtained using forward error correction (FEC), they are also optimal in a certain sense with respect to schemes incorporating feedback, e.g., TCP. Section VII contains a brief conclusion. Appendix A summarizes the key mathematical notation. Proofs of key theorems are presented in Appendices B through G.

II. RELATED WORK

Now that we have introduced the problem we wish to solve, and the technique we will use to solve it, we presently review some of the related literature.

A. Erasure and Error Correction for Network Coded Networks

One of the most important breakthroughs in the information theory community within the past decade has been the concept of network coding [1], [10]–[12] and its necessary utility in achieving the capacity of multicast networks. There is further a recent interest in erasure and error correction over network coded networks [13]–[15]. While some of this work can be used in delay mitigating codes research, e.g., [9], [16], it solves a different problem. The focus in network erasure and error-correcting codes is typically designing block codes that can handle a particular (expected or exact) fraction of erasures or errors. Our work attempts to design codes which have unequal erasure protection (UEP), as in [9], [17], with a specific novel focus on choosing the UEP in such a way as to optimally trade transmission rate for delay. This builds into our model interest not just in the fraction (or expected fraction) of packets which are lost, but rather the way in which the channel combines, reorders, and delivers packets. Classic basic block erasure and error-correcting codes do not incorporate these channel aspects.

Delay of random linear network coding is also addressed in [1], [18], where the delay properties of a particular random linear network coding scheme are assessed. Our scheme, in contrast, attempts to mitigate delay to be lower than that offered by unmodified random linear network coding by trading some of

the rate with a concatenated (outer) delay mitigating code at the source. Alternatively, one could consider a fixed known network topology, and try to design the network code itself to accommodate both full-rate transmission and zero delay, as was attempted in [19]. Designing such a network code requires global centralized coordination and therefore may not be feasible in practical networks. Our scheme, in contrast, is more likely to be feasible since it does not require any such global coordination.

B. Fountain Codes

Fountain codes [20], [21] are another related recent breakthrough in reliable broadcasting/multicasting over networks with users suffering different amounts of packet losses. Fountain codes can be seen as infinite block-length (zero-rate) analogs of irregular low-density parity-check (LDPC) codes [22]–[26], exploiting the same iterative low complexity belief propagation (BP) decoder, and the same sparse graph code construction. An unbounded number of packets are transmitted for a fixed block of source information. Their major innovation is that source information can be recovered with high probability and low computational complexity as long as a sufficient number of transmitted packets have been received by a particular receiver, no matter which subset of the original transmitted packets they were. Thus, different users can come in and out of the network, and can receive different subsets of transmitted packets, but as soon as they have received a number of bits that is $1 + \epsilon$ times the number of source bits they can decode all of the source information with high probability.

Our delay-mitigating codes are fundamentally different from fountain codes in a number of ways. First, our delay mitigating codes are not block codes at the decoder, unlike fountain codes which are. Since we are looking at a situation in which different information is decoded *from all of the previously received packets* upon *each* new packet arrival, block codes, and thus fountain codes do not apply. There have been some studies as to the fraction of bits that can be decoded in fountain codes before the necessary number of packets have been received to decode all of the source information [21], [27], [28]. Delay-mitigating codes require not just that a certain *fraction*, say k/n, of the bits are recoverable, but in fact require the much stronger property that the recovered bits are the first k of n bits of the source. Even more, delay-mitigating codes require a sequence of decoding deadlines be enforced, e.g., decode the first k_1 bits after receiving the first packet, decode the first $k_1 + k_2$ packets after receiving the first two packets, etc. This requirement of enforcing a sequence of decoding constraints precludes the use of Raptor codes.

A second fundamental difference between delay-mitigating codes and fountain codes, turbo codes [29]–[32], or erasure-correcting sparse graph codes, is that delay-mitigating codes are for a different channel than these codes. The most common channel model for erasure-correcting codes is the binary erasure channel (BEC) or packet erasure channel, which erases each bit (or packet) with a given probability p. Our codes, by contrast, are for an abstracted degraded broadcast channel relevant for trading rate for delay over multipath routed/network coded networks.

C. Priority Encoded Transmission

The above discussion of UEP brings up another body of older related work called PET codes, or codes for priority encoded transmission [8], [33] and their rank metric counterparts [17], [19]. The basic setup in PET and the related work [34], [35] involves a source comprised of different parts of information with different priorities, such as results from the use of a successive refinements encoder (e.g., as in the MPEG video standard [36]–[38]). Under successively decreasing prescribed erasure rates, source information of successively lower priority may be decoded. The PET construction is specified such that reception of any fraction $0 < \rho_i \leq 1$ of the encoded packets guarantees that the data with priority greater than or equal to i is all capable of being decoded ($\rho_i > \rho_{i-1}$). To allow this, PET codes time share maximum distance separable (MDS) codes of different rates, and the time sharing construction can be thought as achieving points on a capacity region of an associated broadcast channel [33].

Our work is different from classic PET coding due to the focus on quantifying delay. In delay-mitigating codes, the different classes of source information correspond to different frames or time instants in the source data, with the requirement that earlier source data must be played out at the sink before later source data can be used. In a particular context of uniform network coded network and uniform multipath routed networks discussed later in the paper, our code construction amounts to optimally selecting the component rates ρ_i for a PET code, and so PET codes are critical to our work. Selecting these component rates ρ_i to correctly ensure specified delay constraints is in fact nontrivial and part of our work in this paper addresses this problem, see Section V.

D. Error Exponents as Delay Metrics

A classic metric used as a surrogate for delay in information-theoretic analysis of channels is the random coding error exponent [39]. The error exponent is the asymptotic exponential rate at which the sequence of error probabilities decays as a function of the block length in the limit of infinite block lengths when a random code is used. Because this metric is closely related to the necessary block length to achieve a certain bound on the probability of error for a given code rate, albeit when using a random code, it is often considered to be a reasonable metric related to the delay associated with the channel [40]. However, error exponents are inappropriate for our problem for two reasons.

First, the error exponents of codes we encounter in the context of this paper are infinite. In channels such as the BEC, the error exponent is finite, reflecting the fact that the probability of error when using random codes, although decreasing to zero with increasing block lengths, is not zero for any finite block. The channels we are investigating, by contrast, do not share this characteristic. From the perspective of code design they are multiterminal analogs of fixed erasure fraction channels, which have capacity-achieving (MDS) codes with zero probability of error even for finite block lengths.

Second, the relationship between block length and temporal delay is tenuous in a network context. Rate measured in bits per channel use does not map smoothly to rate measured in bits per
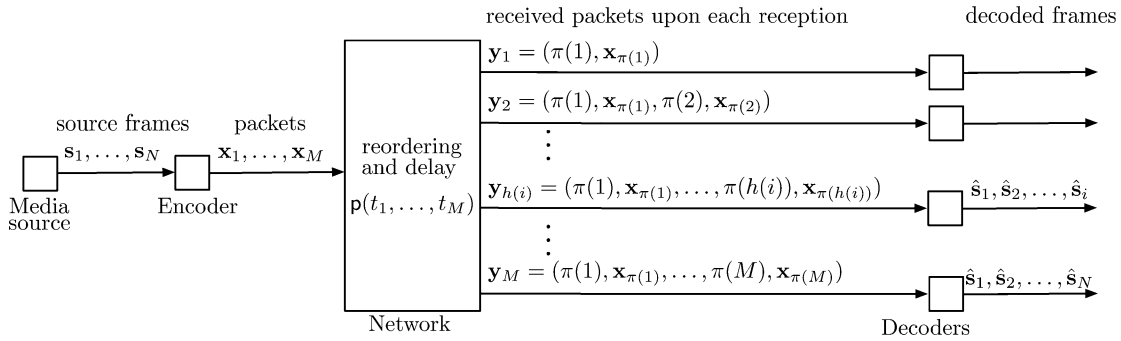
Fig. 2. The permutation channel reinterpreted as an abstracted degraded broadcast channel for the purpose of analyzing rate–delay tradeoffs. Upon reception of the first $h(i)$ packets, the sink may decode the first i frames. The decoding deadlines $\boldsymbol{h} = (h(1), \ldots, h(N))$ are illustrated in Fig. 7.

second when a channel use is defined as simultaneously transmitting a block of packets. This is because the packets take different amounts of time to traverse the network, owing to time spent waiting in different queues, and thus arrive at the sink for decoding at (unevenly spaced) random time instants. Using this channel repeatedly in a classic block code manner, the temporal rate at which information may be played out at the destination in bits per second is only indirectly related to the rate of the code in bits per channel use.

III. CAPACITY REGION OF SOME RELEVANT DEGRADED BROADCAST CHANNELS

In this section, we introduce two abstracted broadcast channels in which different receivers correspond to successive packet arrivals at a common sink in the network. These two channels model the case of packets reordering with delays (Section III-A) and network coded networks (Section III-B), in a way that we will clarify in greater detail in (Section III-C).

A. Permutation Channel Capacity Region

Recall π is the random permutation determining the order of packet arrivals at the destination, and $\mathsf{p}(\tilde{\pi})$ is the probability of each possible permutation $\tilde{\pi} \in \Pi$. We are interested in the region $\mathcal{R}$ of possible amounts of information that may be decoded upon each successive packet arrival at the sink. In particular, denote by R_j the amount of new information that can be decoded after j packets have arrived at the sink per collection of M packets transmitted at a common time at the source. Collect these rates R_j into a vector $\boldsymbol{r} = (R_1, \ldots, R_M)$. We can determine the region $\mathcal{R}$ of feasible (mutually satisfiable) rate vectors $\boldsymbol{r}$ as the capacity region of an abstracted degraded broadcast channel shown in Fig. 2. Different receivers in the abstracted broadcast channel correspond to the cumulative packets received upon each successive packet reception at the sink. The packets are assumed to be labeled so that upon receiving the ith packet, $\boldsymbol{x}_{\pi(j)}$, the destination knows both the payload $\boldsymbol{x}_{\pi(j)}$ and the packet index $\pi(j)$ (through the packet header). The observed values at the jth receiver are $\boldsymbol{y}_j := (\pi(1), \boldsymbol{x}_{\pi(1)}, \pi(2), \boldsymbol{x}_{\pi(2)}, \ldots, \pi(j), \boldsymbol{x}_{\pi(j)})$. This broadcast channel is degraded; this is seen from the fact that the time reversed observations form a Markov chain through erasure operations $\boldsymbol{y}_M \to \boldsymbol{y}_{M-1} \to \cdots \to \boldsymbol{y}_1$.

Theorem (Capacity Region of Degraded Broadcast Channel (DBC) [6], [5], [7], [41]): The capacity region of the degraded broadcast channel (DBC) is the closure of the convex hull of the region $\mathcal{R}$ of rates satisfying

$$R_1 \leq \mathcal{I}(\boldsymbol{y}_1; \boldsymbol{u}_1) \tag{3}$$
$$R_2 \leq \mathcal{I}(\boldsymbol{y}_2; \boldsymbol{u}_2 | \boldsymbol{u}_1), \ldots \tag{4}$$
$$R_M \leq \mathcal{I}(\boldsymbol{y}_M; \boldsymbol{x} | \boldsymbol{u}_{M-1}) \tag{5}$$

for a sequence of dummy discrete random variables $\boldsymbol{u}_1 \to \cdots \to \boldsymbol{u}_{M-1} \to \boldsymbol{x}$ with bounded support, where $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M)$ is the channel input.[2]

Note that the auxiliary random variables in this theorem play the part of message variables in a random code [6], [5], [7], i.e., the decoder at receiver j can determine the messages associated with the random variables $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_j$, for each $j \in \{1, \ldots, M\}$.

Proposition 1 (Permutation Channel Capacity Region): In the particular case of the permutation channel, the generic DBC capacity region expression can be simplified to

$$R_1 \leq \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \left(\mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)} \right) - \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)} | \boldsymbol{u}_1 \right) \right), \ldots \tag{6}$$
$$R_k \leq \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \left(\mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_{k-1} \right) \right.$$
$$\left. - \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_k \right) \right), \ldots \tag{7}$$
$$R_M \leq \mathsf{H}(\boldsymbol{x} | \boldsymbol{u}_{M-1}). \tag{8}$$

The proof of this proposition is provided in Appendix B. From this expression we can view the capacity region as the image under a linear transformation whose coefficients are determined by the permutation distribution of the set of all possible entropy vectors created from joint entropies of subsets of the dummy random variables $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{M-1}, \boldsymbol{x}$. In the two special cases of interest we presently discuss, the expression for the rate region can be simplified even further.

Proposition 2 (Special Case: Single Permutation): When only one permutation is possible, the amount of information received upon the jth packet arrival is limited by the cumulative number of received bits. In particular, the rate region $\mathcal{R}$ is the

[2]References [6], [5], [7], [41] mostly focus on the two-user case, however there is no difficulty in extending the coding scheme [7] and converse proofs [41] to the presented M-user case.
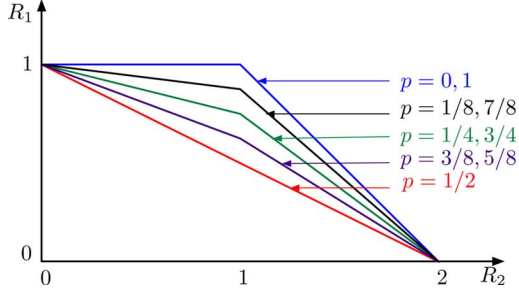
Fig. 3. The permutation channel's capacity region for $M = 2$. The two extreme rate regions (known single permutation ($p \in \{0, 1\}$) and uniform permutation selection ($p = 1/2$)) are shown here, with p indicating the probability that the packet numbered 1 when placed in the stack at the source node arrives first at the receiver. Note that the rate region boundaries for the intermediate cases lie in between those for these two extreme cases.

collection $\boldsymbol{r} = (R_1, \ldots, R_M)$, $\boldsymbol{r} \geq \boldsymbol{0}$ specified by the M inequalities

$$\sum_{j=1}^{m} R_j \leq mK, \quad \forall m \in \{1, \ldots, M\}.$$

This rate region for $M = 2$ is shown as the $p \in \{0, 1\}$ region in Fig. 3. An additional interesting special case occurs when the distribution on the reordering permutation $\tilde{\pi}$ is uniform. Such a uniform distribution for π occurs, for instance, via symmetry when the unordered arrival times $\{t_j\}$ are i.i.d.

Theorem 2 (Uniform Permutation Channel): When all permutations are equally likely ($\mathsf{p}(\tilde{\pi}) = 1/|\Pi|$ for each $\tilde{\pi} \in \Pi$) the capacity region is specified by the inequality

$$\mathcal{R} = \left\{ \boldsymbol{r} = (R_1, \ldots, R_M) : \boldsymbol{r} \geq \boldsymbol{0}, \sum_{j=1}^{M} \frac{R_j}{j} \leq K \right\}. \quad (9)$$

The proof is found in Appendix C. This rate region for $M = 2$ is shown as the $p = 1/2$ region in Fig. 3.

B. Network Coded Case

In the random linear network coded case where the packets that arrive at time instants $\{\tau_j\}$ are linear combinations of the transmitted packets, we are again interested in a similar abstracted DBC, depicted in Fig. 4. The different receivers correspond to successive innovative packet arrivals and the vectors $\boldsymbol{g}_k$ correspond to the global encoding vector of the kth received packet. Because we are considering random linear network coding from a source-to-sink viewpoint, we model a random linear network coded network as a device that selects the random global encoding matrices $\boldsymbol{G}_k = [\boldsymbol{g}_1^\top, \boldsymbol{g}_2^\top, \ldots, \boldsymbol{g}_k^\top]^T$ according to a probability distribution $\mathsf{p}(\tilde{\boldsymbol{G}}_k)$ on $\mathcal{G}_k$ the set of $k \times M$ matrices with rank k. The rate region corresponding to the amounts of new information that can be decoded upon successive innovative packet arrivals at the sink is given by

$$R_1 \leq \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1)(\mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x}) - \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x} | \boldsymbol{u}_1)) \quad (10)$$

$$R_k \leq \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k)\Big(\mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1}) - \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_k)\Big) \quad (11)$$

$$R_M \leq \mathsf{H}(\boldsymbol{x} | \boldsymbol{u}_{M-1}) \quad (12)$$

where (11) ranges over $k \in \{2, \ldots, M-1\}$. This region follows from substituting the definition for the M received signals in the network coded context into (3), (4), and (5), and mimicking the steps detailed in Appendix B.

We consider next the specific case where we can model the source-to-destination channel created by network coding as sampling the random encoding matrices $\boldsymbol{G}_k$ uniformly from the set $\mathcal{G}_k$ of all encoding matrices in GF $(q)^{k \times M}$ of rank k (henceforth, the uniform network coded channel). In this instance, the capacity region of the abstracted broadcast channel determining the amounts of information decodable upon each successive packet arrival is given by the following theorem.

Theorem 3 (Uniform Network Coded Channel Capacity Region): For the uniform network coded channel ($\mathsf{p}(\tilde{\boldsymbol{G}})$ is uniform) the capacity region is specified by the inequality

$$\mathcal{R} = \left\{ \boldsymbol{r} = (R_1, \ldots, R_M) : \boldsymbol{r} \geq \boldsymbol{0}, \sum_{j=1}^{M} \frac{R_j}{j} \leq K \right\}.$$

The proof is found in Appendix D. Note that this rate region is the same as that obtained for the uniform packet reordering channel in Theorem 2.

C. Relating the Abstracted Broadcast Channel to Delay Mitigating Codes

Having introduced the two abstracted broadcast channels (Theorems 2 and 3), we now discuss capacity-achieving codes for these channels, their associated fundamental limits, and their connection with the problem of connection-oriented transmission for multipath routed and network coded networks.

First, it is important to point out that Figs. 2 and 4 detail a *single use* of the discrete memoryless channel under consideration. A block code with block length L for this discrete memoryless channel would have codewords which individually are a block of L collections of M packets (LM packets all together), labeled $\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l, \ldots, \boldsymbol{x}^L$ with $\boldsymbol{x}^l := [\boldsymbol{x}_1^l, \ldots, \boldsymbol{x}_M^l]$. For the permutation channel, the jth receiver receives a sequence of L collections of j packets

$$\boldsymbol{y}_j^L = \left[\Big(\pi_l(1), \boldsymbol{x}_{\pi_l(1)}^l, \ldots, \pi_l(j), \boldsymbol{x}_{\pi_l(j)}^l\Big) | l \in \{1, \ldots, L\} \right].$$

Due to the memoryless nature of the channel there are L i.i.d. permutations governing the arrival order of each of the L collections of M packets $\pi_1 := (\pi_1(1), \ldots, \pi_1(M))$, $\pi_2 := (\pi_2(1), \ldots, \pi_2(M))$, through $\pi_L := (\pi_L(1), \ldots, \pi_L(M))$. Because this is clearly a discrete memoryless DBC, the capacity region theorem can be applied, and a capacity region $\mathcal{R}$ of achievable rate vectors $\boldsymbol{r} = (R_1, \ldots, R_M)$ with R_j signifying the number of bits deliverable per channel use to receiver j. The collective amount of error-free information decoded by receiver j will be LR_j. Similarly, for the network coded channel, Fig. 4 details a single use of the associated discrete broadcast channel, while a block code would code over multiple such channel uses.

Second, we emphasize that the capacity regions in Theorems 2 and 3 hold under an assumed uniform distribution: equally likely packet arrival orders (permutations) for Theorem 2 and equally likely encoding matrices of each rank
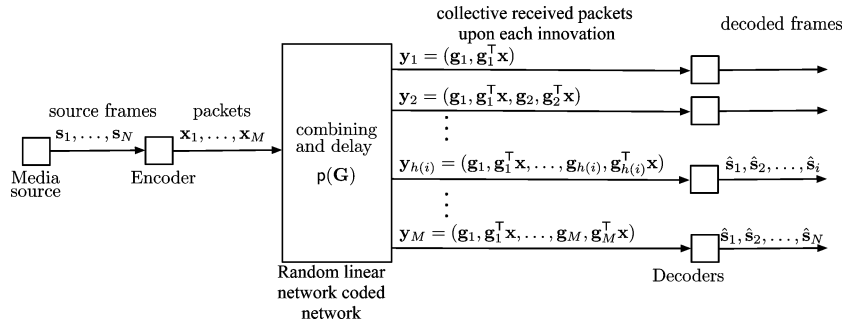
Fig. 4. Point-to-point network coded channel as a degraded broadcast channel.

for Theorem 3. A more general problem is to determine the capacity of the channels under an arbitrary distribution on permutations and encoding matrices, respectively, but still assuming the ordered arrival instants $\tau_j := t_{\pi(j)}, j \in \{1, \ldots, M\}$ are independent of the arrival order π. An even more general problem is to relax this assumption and allow dependence between the arrival order and the ordered arrival times. A uniform distribution for π and independence between τ and π happen, for instance, if the unordered arrival times, $t_1, \ldots, t_M$, are i.i.d. This assumption is valid in a network with M independent paths through the network where each path has an identical delay distribution. Naturally, we would like to understand with some level of rigor this statistically symmetric case before delving into the more complex, yet more realistic, irregular cases.

We want to use a multipath network like this to provide a constant bit rate connection between the source and destination, in the sense of providing a new frame containing a constant number of bits every T_s seconds, and we want these frames to be played out in a specified order without any being omitted. The reordering phenomenon caused by the random arrival order makes this difficult to do without introducing extra delay if uncoded transmission is to be used, because we must buffer the received packets so we can play them out in order. Alternatively, we can encode the contents of the M transmitted packets. This would correspond to a block code (as discussed above) for the introduced abstracted degraded broadcast channel with block length $L = 1$. The number of bits R_j that can be guaranteed (with probability 1) to be delivered by this block code after j packets have been received *must* yield a rate vector $\boldsymbol{r} = (R_1, \ldots, R_M)$ lying inside the capacity region $\mathcal{R}$ for the associated degraded broadcast channel. To see this (in spite of the infinite block lengths used in the typical DBC capacity region proof), simply note we can make a block length L code with rate (per channel use) vector $\boldsymbol{r}$, by concatenating L of our block length 1 codes end to end. Since we are only considering error-free block-length 1 codes, this code cannot be any better than the fundamental limits for reliable communication as $L \to \infty$ provided by the degraded discrete memoryless broadcast channel capacity theorems. For many discrete memoryless DBCs, not all rate vectors $\boldsymbol{r}$ may be achieved with a block length $L = 1$, and thus $\mathcal{R}$ forms a strict outer bound for the set of block length $L = 1$ code rate vectors that we are interested in. In these cases, the calculus on the associated rate region to trade aggregate bit rate for aggregate delay that we discuss in the next section would yield bounds for the associated rate–delay tradeoff

which could not be reached with practical codes. However, as we shall show in Section V, the channel we are studying has nontrivial practical codes which achieve points on the boundary of the capacity region with block length $L = 1$, and thus the rate–delay tradeoffs we present are achievable with practical and low-complexity decodable codes.

IV. FUNDAMENTAL RATE–DELAY TRADEOFFS

In this section, the capacity regions obtained in Section III are used to describe fundamental rate–delay tradeoffs in uniform multipath routed and network coded networks. Rate and delay are defined in Section IV-A relative to a constant bit rate source, along with the optimization problem whose solution is the rate–delay tradeoff. This section (Section IV-A) is general, and can be applied out of the scope of the limited uniform multipath routed and network coded cases. In Section IV-B, we identify the structural properties of this optimization problem in the uniform cases, showing that it may be simplified to an integer programming problem, amenable to exact exhaustive solution for small M, N. In order to make this problem more tractable for large M, N, Section IV-C introduces a variational relaxation of the integer programming problem. Algebraic expressions for the rate–delay tradeoff are obtained by solving this variational calculus problem for some special packet arrival time statistics in Section IV-D (constant inter arrival times) and Section IV-E (exponential i.i.d. arrival times).

A. Rate–Delay Calculus Problem Formulation

Theorems 2 and 3 give the region $\mathcal{R}$ describing the amounts of new information available after each successive packet arrival at the sink, provided the permutations and encoding matrices, respectively, are selected uniformly. We can incorporate the inter-arrival statistics with these results to determine the information transmission rate (in bits per second) and delay (in seconds) associated with each point $\boldsymbol{r}$ in the capacity region. With the connection oriented context in mind, suppose that the source decoder at the sink node needs to decode a new frame $\boldsymbol{s}_i$ every T_s seconds. Then the overall rate at which data is decoded at the source decoder at the sink is

$$\rho(\boldsymbol{r}) := \frac{1}{NT_s} \sum_{j=1}^{M} R_j \qquad (13)$$

bits per second. This means we are trying to decode $\frac{1}{N} \sum_{j=1}^{M} R_j$ bits every T_s seconds.

We next define a class of delay metrics, parameterized by $p \in [1, \infty)$, for each rate vector $\boldsymbol{r}$ that penalizes the (random) delay incurred by every frame, according to the expression

$$D_p(\boldsymbol{r}) = \sum_{i=1}^{N} \left[\left(\tau_{g(i,\boldsymbol{r})} - iT_s \right)^+ \right]^p \qquad (14)$$

where $(x)^+ := \max\{x, 0\}$ and

$$g(i, \boldsymbol{r}) := \inf \left\{ n \Big| \sum_{j=1}^{n} R_j \geq iT_s \rho(\boldsymbol{r}) \right\}. \qquad (15)$$

Here, $g(i, \boldsymbol{r})$ is the number of packets that must be received before the sink can decode the first i frames of the source. The delay is a random variable as it is a function of the arrival instants $\{\tau_j\}$. Note that the assumed functional form (14) allows a wide variety of delay metrics, reflecting the fact that the overall delay will be likely to be a function of the delay $(\tau_{g(i,\boldsymbol{r})} - iT_s)^+$ of ith frame, which is the difference between the time iT_s it is to be played out, and the (random) time it is guaranteed to be available $\tau_{g(i,\boldsymbol{r})}$. Due to properties of the p norms, the family of functionals (14) can thus be stretched all the way from being proportional to the sum of the frame delays, to being proportional to the worst frame delay by using larger and larger powers p.

At the heart of the idea of rate–delay calculus is the intent of finding the point $\boldsymbol{r}$ in the coding achievable rate region $\mathcal{R}$ which maximizes the overall rate $\rho(\boldsymbol{r}) \leq \rho_d^*$ among all codes with expected delay $\delta(\boldsymbol{r}) = \boldsymbol{E}[D_p(\boldsymbol{r})]$ less than or equal to some bound d. The optimal rate–delay tradeoff is

$$\rho_d^* := \max_{\boldsymbol{r} \in \mathcal{R}} \{ \rho(\boldsymbol{r}) | \boldsymbol{E}[D_p(\boldsymbol{r})] \leq d \}. \qquad (16)$$

B. Structural Properties of the Rate–Delay Tradeoff in Uniform Case

Because the rate regions in Theorems 2 and 3 are the same, calculating the rate–delay tradeoffs for these two channels amounts to the same calculus problem. Thus, for the remainder of this paper, we handle these two cases simultaneously, focussing entirely on the uniform cases. The next theorem shows that in these uniform cases, the rate–delay tradeoff, although correctly posed as a continuous optimization, is found as the solution to an integer programming problem (with a finite set of feasible solutions).

Theorem 4 (Structural Properties of Rate–Delay Tradeoff): When the channel selects the permutation uniformly from the set of all permutations, the optimal rate–delay tradeoff (16) may be obtained by solving the optimization problem

$$\rho_d^* = \max_{\boldsymbol{h}} \left\{ \rho(\boldsymbol{h}) = \frac{K}{T_s} \left(\sum_{m=1}^{n(\boldsymbol{h})} \frac{\Delta(\boldsymbol{h})_m}{l(\boldsymbol{h})_m} \right)^{-1} \Bigg| \mathbb{E}[D_p(\boldsymbol{h})] \leq d \right\} \qquad (17)$$

over the decoding deadlines $\boldsymbol{h} = (h(i), i = 1, \ldots, N)$ which are nondecreasing integer-valued functions mapping $\{1, \ldots, N\} \to \{1, \ldots, M\}$. Here $h(i)$ indicates the number of packets that must be received before decoding the first i frames,

$n(\boldsymbol{h})$ is the number of distinct values of $\boldsymbol{h}$, $l(\boldsymbol{h})_m$ is the mth distinct value of $\boldsymbol{h}$, and $\Delta(\boldsymbol{h})_m$ is the number of frame indices associated with the mth distinct level of $\boldsymbol{h}$. Finally, $D_p(\boldsymbol{h})$ equals $D_p(\boldsymbol{r})$ as defined in (14) but with $g(i, \boldsymbol{r}) = h(i)$ in (15) for each $i = 1, \ldots, N$.

The proof is found in Appendix E. This structural transformation to an integer programming problem over $\boldsymbol{h}$ allows direct calculation of the rate–delay tradeoffs by explicit enumeration over all nondecreasing sequences satisfying the delay bound.

C. Variational Approximation Based Rate–Delay Tradeoff for Large M, N

For large M, N, the solution by explicit enumeration in Theorem 4 is computationally infeasible, necessitating an approximate solution. We obtain an approximation by transforming the integer programming problem into a variational calculus problem over a family of nondecreasing continuous functions, whose solution, via Lagrangian techniques, can be found as a solution to an algebraic equation. In order to avoid unbounded rates and delays as M, N grow large, we first normalize the rate and delay metrics to lie in between 0 and 1.

Since $\sum_{j=1}^{M} R_j \leq MK$ it follows that $\rho(\boldsymbol{r})$ in (13) and hence $\rho(\boldsymbol{h})$ in (17) is upper-bounded by $\frac{MK}{NT_s}$. The normalized rate is then

$$\bar{\rho}(\boldsymbol{h}) = \frac{NT_s}{MK} \rho(\boldsymbol{h}) = \frac{N}{M} \left(\sum_{m=1}^{n(\boldsymbol{h})} \frac{\Delta(\boldsymbol{h})_m}{l(\boldsymbol{h})_m} \right)^{-1}. \qquad (18)$$

The normalized delay is obtained by noting that our delay metric is monotone nondecreasing in $h(i)$, and thus the greatest delay is obtained when the decoding deadlines are $h(i) = M$ for each $i = 1, \ldots, N$. Thus, to obtain a normalized delay lying in between 0 and 1, we need only to divide the expected nonnormalized random delay metric by its expected value at the extreme decoding deadlines

$$\bar{\delta}(\boldsymbol{h}) := \frac{\mathbb{E}[D_p(\boldsymbol{h})]}{\mathbb{E}[D_p(\{h(i) = M, \quad \forall i \in \{1, \ldots, N\}\})]}.$$

In order to approximate the solution to the integer programming problem by the solution to a variational calculus problem, first define a zero-order hold piecewise continuous function $\mathsf{z} : [0, 1] \to [0, 1]$ such that $\mathsf{z}(\frac{i}{N}) = \frac{h(i)}{M}$

$$\mathsf{z}(t) = \sum_{i=1}^{N} \frac{h(i)}{M} \mathbf{1}_{\frac{i-1}{N} \leq t < \frac{i}{N}}, \quad t \in [0, 1].$$

The normalized rate $\rho(\boldsymbol{h})$ may be expressed in terms of z as the functional $\bar{\rho}[\mathsf{z}]$ according to

$$\bar{\rho}(\boldsymbol{h})^{-1} = \sum_{m=1}^{n(\boldsymbol{h})} \frac{1}{\frac{l(\boldsymbol{h})_m}{M}} \frac{\Delta(\boldsymbol{h})_m}{N} = \bar{\rho}[\mathsf{z}]^{-1} = \int_0^1 \frac{1}{\mathsf{z}(t)} \mathrm{d}t.$$

The normalized delay is then approximated by a functional $\bar{\delta}[\mathsf{z}]$ of z as well

$$\bar{\delta}[\mathsf{z}] = \frac{\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}\left[\left[\left(\tau_{Mz(i/N)} - \frac{i}{N} NT_s \right)^+ \right]^p \right]}{\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}\left[\left[\left(\tau_M - \frac{i}{N} NT_s \right)^+ \right]^p \right]}$$

$$\rightarrow \frac{\int_0^1 \mathbb{E}\left[\left[\left(\tau_{Mz(t)} - tNT_s\right)^+\right]^p\right] dt}{\int_0^1 \mathbb{E}\left[\left[(\tau_M - tNT_s)^+\right]^p\right] dt} \quad (19)$$

for N large.

The rate–delay tradeoff for large M is obtained by solving the calculus of variations [42], [43] problem for a normalized delay bound $\bar{d} \in [0, 1]$

$$\bar{\rho}_{\bar{d}}^* := \max_{z:[0,1]\to[0,1]} \left\{ \bar{\rho}[z] \mid \bar{\delta}[z] \le \bar{d}, \ \frac{\partial z(t)}{\partial t} \ge 0 \right\}. \quad (20)$$

Here, z is no longer constrained to be a piecewise constant function, and, in fact, is now considered among the class of continuously differentiable nondecreasing functions. For large M (where explicit enumeration of $\boldsymbol{h}$ is too difficult), the intervals on which z was to be piecewise constant should be very small, so one would expect to get a good approximation for these cases. Using monotonicity of the function $\frac{1}{x}$, the problem (20) may be cast in standard variational calculus form

$$\frac{1}{\bar{\rho}_{\bar{d}}^*} := \min_{z:[0,1]\to[0,1]} \left\{ \int_0^1 \frac{1}{z(t)} dt \mid \bar{\delta}[z] \le \bar{d}, \ \frac{\partial z(t)}{\partial t} \ge 0 \right\}. \quad (21)$$

The following theorem gives a general form for the optimal decoding deadline function z under an even more general family of delay metrics than (19).

Theorem 5 (Variational Approximation to Rate–Delay Tradeoff_: Suppose the normalized delay function may be written as an integral of a function $Q(t, z(t))$, i.e.,

$$\bar{\delta}[z] = \int_0^1 Q(t, z(t)) dt. \quad (22)$$

Then, for those t_0 such that $\frac{\partial z^*}{\partial t}|_{t=t_0} \ne 0$, the optimal decoding deadline function $z_{\bar{d}}^*$ to solve (20) is the solution z to the equation

$$z(t_0) = \left(\mu \frac{\partial Q(t_0, z(t_0))}{\partial z(t_0)} \right)^{-\frac{1}{2}}$$

with $\mu = \mu_{\bar{d}}$ selected so that $\bar{\delta}[z_{\bar{d}}^*] \le \bar{d}$. The rate–delay tradeoff is then given by $\bar{\rho}_{\bar{d}}^* = \bar{\rho}[z_{\bar{d}}^*]$.

Proof: The Lagrangian $\mathcal{L}(z, \mu, a)$ is

$$\int_0^1 \frac{1}{z(t)} dt + \mu \left(\int_0^1 Q(t, z(t)) dt - \bar{d} \right) + \int_0^1 a(t) \frac{\partial z(t)}{\partial t} dt$$

where $a(t)$ is the Lagrange multiplier on the constraint $\frac{\partial z(t)}{\partial t} \ge 0$ at t, for each $t \in [0, 1]$. The solution to the variational problem (20) must satisfy the algebraic equation obtained by differentiating $\mathcal{L}(z, \mu, a)$ at each $t = t_0$

$$-\frac{1}{(z^*(t_0))^2} + \mu \cdot \frac{\partial Q(t_0, x)}{\partial x}\bigg|_{x=z^*(t_0)} + \alpha(t_0) \mathbf{1}_{\{\frac{\partial z^*(t_0)}{\partial t}\}} = 0$$

for each $t_0 \in [0, 1]$, with $\mu, z^*(0)$ such that $\bar{\delta}[z^*] \le \bar{d}$ and $\alpha(t)$ selected so as to make z^* constant on intervals where it is constant (and thus it is not necessary to actually solve for it). $\square$

Note the family of functionals (19) is a particularly useful form of the more general class of functionals (22). We next

specialize Theorem 5 to the special cases of constant packet inter-arrival times (Section IV-D) and i.i.d. packet inter-arrival times (Section IV-E).

D. Rate–Delay Tradeoffs for Constant Packet Inter-Arrival Times

Consider the case when $\tau_j = jT_s$ and $N = M$, so that time is slotted on packet arrivals and the only remaining random element in the system is the uniform permutation selection. The normalized delay functional becomes

$$\bar{\delta}(\boldsymbol{h}) = \frac{\sum_{j=1}^M (h(j) - j)^+}{\sum_{j=1}^M (M - j)} = \frac{2}{M(M-1)} \sum_{j=1}^M (h(j) - j)^+$$

$$= \frac{2M}{M-1} \sum_{j=1}^M \left(\frac{h(j)}{M} - \frac{j}{M} \right)^+ \frac{1}{M} \quad (23)$$

$$\bar{\delta}[z] = \frac{2M}{M-1} \int_0^1 (z(t) - t)^+ dt. \quad (24)$$

Theorem 6 below specializes Theorem 5 to the case of constant packet inter-arrival times, yielding approximate but accurate closed form rate–delay tradeoffs for large M.

Theorem 6 (Approximate Rate–Delay Tradeoff for Constant Inter Packet Arrival Times): The approximate (asymptotically accurate) rate–delay tradeoff solving (20) for a normalized delay bound $\bar{d} \in [0, 1]$ is given by

$$\bar{\rho}_{\bar{d}}^* = \left(1 - \log \left(\sqrt{\frac{M-1}{M} \bar{d}} \right) \right)^{-1}.$$

Alternatively, the rate–delay tradeoff for a normalized rate requirement $\bar{\rho} \in [0, 1]$ is given by

$$\bar{d}_{\bar{\rho}}^* = \frac{M}{M-1} \exp\left(2 - \frac{2}{\bar{\rho}} \right).$$

The optimal decoding deadline function z^* takes the form

$$z^*(t) = \begin{cases} \zeta, & 0 \le t \le \zeta^* \\ t, & \zeta^* < t \le 1 \end{cases}$$

with $\zeta^* = \sqrt{\frac{M-1}{M} \bar{d}}$.

The proof is found in Appendix F. The approximate and exact rate–delay tradeoffs are plotted together in Fig. 5.

E. Trading Rate for Average Sum Delay With i.i.d. Arrival Times

The second special case we consider is that of i.i.d. packet arrival times $(t_1, \ldots, t_M)$ with density f, cumulative distribution F, and inverse function to the cumulative distribution F^{-1}. The asymptotic theory of order statistics ([44, Theorem 10.3, p. 288]) asserts that $\sqrt{M}(\tau_{Mz} - \mathsf{F}^{-1}(z))$ has an asymptotic normal distribution as $M \to \infty$. That is, provided $0 < \mathsf{f}(\mathsf{F}^{-1}(z)) < \infty$ (an easily satisfied condition), then

$$\sqrt{M}(\tau_{Mz} - \mathsf{F}^{-1}(z)) \xrightarrow{M\to\infty} \mathcal{N}\left(0, \frac{z(1-z)}{(\mathsf{f}(\mathsf{F}^{-1}(z)))^2} \right). \quad (25)$$
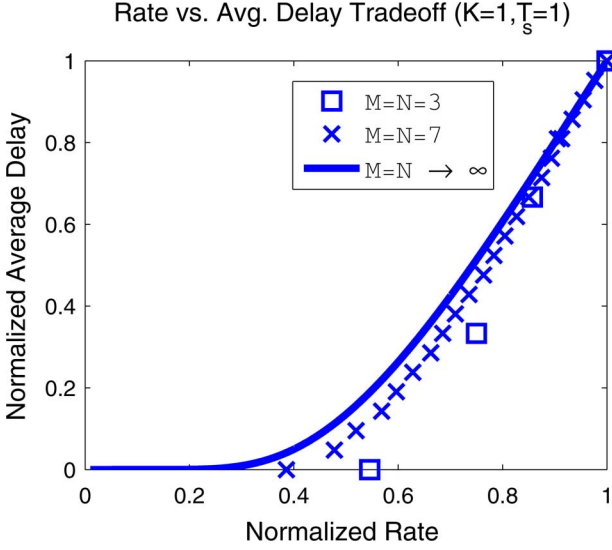
Fig. 5. Exact and approximate rate–delay tradeoffs when the (reordered) packets arrive in a constant periodic manner.

It follows that for large $M, \mathbb{E}[\tau_{Mz(t)}] \approx \mathsf{F}^{-1}(\mathsf{z}(t))$ for each $t \in (0,1)$, and that the random variable $\tau_{Mz(t)}$ is increasingly concentrated around its mean, and therefore

$$\mathbb{E}[(\tau_{Mz(t)} - tNT_s)^+] \approx (\mathsf{F}^{-1}(\mathsf{z}(t)) - tNT_s)^+.$$

This implies that the large M approximate form of the delay functional for the mean sum delay metric, i.e., $p = 1$ in (19), is

$$\bar{\delta}[\mathsf{z}] \approx \frac{\int_0^1 (\mathsf{F}^{-1}(\mathsf{z}(t)) - tNT_s)^+ \mathrm{d}t}{\int_0^1 \mathbb{E}[(\tau_M - tNT_s)^+] \mathrm{d}t}. \qquad (26)$$

Here, the denominator may also be approximated through the asymptotic distribution of the extreme values ([44, Ch. 11]), which after scaling can take only one of three forms.

To get a closed-form expression for the rate–delay tradeoff, we now specialize to the case where the arrival times $(t_1, \ldots, t_M)$ are i.i.d. exponential random variables with parameter λ. Symmetry suggests that the permutation distribution will be uniform, thereby selecting the capacity region solved in Theorem 2. We first develop an expression for the expected sum delay for fixed decoding deadlines $\boldsymbol{h}$, i.e., the expected value of (14) for $p = 1$ and i.i.d. exponentials. We then develop an approximation for the normalized expected sum-delay expressed (19) in terms of the functional z appropriate for large M.

For finite M, the ordered packet arrival times $(\tau_1, \ldots, \tau_M)$ are order statistics, having marginal distributions

$$\mathsf{f}(\tau_j) = j\lambda \binom{M}{j} \sum_{k=0}^{j-1} \binom{j-1}{k} (-1)^{j-k-1} \mathrm{e}^{-(M-k)\lambda\tau_j}$$

for $\tau_j \geq 0$ and $j \in \{1, \ldots, M\}$.

Under frame decoding deadline constraints $\boldsymbol{h}$, the expected decoding delay of frame i is then given by applying the defi-

nition of expectation with respect to this distribution, making a change of variables, and integrating by parts to get

$$\mathbb{E}\left[(\tau_{h(i)} - iT_s)^+\right] = h(i) \binom{M}{h(i)} \sum_{k=0}^{h(i)-1} \binom{h(i)-1}{k} \times \frac{(-1)^{h(i)-k-1}\mathrm{e}^{-(M-k)\lambda i T_s}}{\lambda(M-k)^2}$$

for $i \in \{1, \ldots, N\}$. Summing these expected decoding delays over each frame yields an expected overall delay under $\boldsymbol{h}$ of

$$\mathbb{E}[D_1(\boldsymbol{h})]$$
$$= \sum_{i=1}^{N} h(i) \binom{M}{h(i)} \sum_{k=0}^{h(i)-1} \binom{h(i)-1}{k} (-1)^{h(i)-k-1}$$
$$\times \frac{\mathrm{e}^{-(M-k)\lambda i T_s}}{\lambda(M-k)^2}.$$

The optimal rate-delay tradeoff for small or moderate M, N can be determined by search over all decoding deadline sequences $\boldsymbol{h}$ as in Theorem 4. Fig. 6 plots some sample normalized rate–delay tradeoffs.

For the exponential cumulative distribution function (CDF) $\mathsf{F}(x) = 1 - \mathrm{e}^{-\lambda x}$, the inverse CDF is $\mathsf{F}^{-1}(z) = -\lambda^{-1}\log(1-z)$. Substituting this inverse CDF into (26) yields

$$\bar{\delta}[\mathsf{z}] \approx \frac{\int_0^1 (-\lambda^{-1}\log(1-\mathsf{z}(t)) - tNT_s)^+ \mathrm{d}t}{\int_0^1 \mathbb{E}[(\tau_M - tNT_s)^+] \mathrm{d}t} \qquad (27)$$

which may be used to obtain the following rate–delay tradeoff.

Theorem 7 (Rate–Delay Tradeoff for i.i.d. Exponential Packet Arrival Times): The approximate (asymptotically accurate) rate–delay tradeoff solving (20) for a normalized delay bound $\bar{d} \in [0,1]$ is given by

$$\bar{\rho}_{\bar{d}}^* \approx \lambda NT_s \left(\frac{(\gamma + \log M)\sqrt{\bar{d}}}{1 - \mathrm{e}^{-\gamma\sqrt{\bar{d}}}M^{-\sqrt{\bar{d}}}} + \log \frac{\mathrm{e}^{\lambda NT_s} - 1}{\mathrm{e}^{\gamma\sqrt{\bar{d}}}M^{\sqrt{\bar{d}}} - 1} \right)^{-1}$$

when $N > \frac{\gamma + \log M}{\lambda T_s}$ and

$$\bar{\rho}_{\bar{d}}^* \approx \lambda NT_s \left[\frac{\sqrt{(2\gamma + 2\log M - \lambda NT_s)\lambda NT_s \bar{d}}}{1 - \exp(-\sqrt{(2\gamma + 2\log M - \lambda NT_s)\lambda NT_s \bar{d}})} \right.$$
$$\left. + \log \frac{\exp(\lambda NT_s) - 1}{\exp(\sqrt{(2\gamma + 2\log M - \lambda NT_s)\lambda NT_s \bar{d}}) - 1} \right]^{-1}$$

when $N \leq \frac{\gamma + \log M}{\lambda T_s}$. Here in both cases $\gamma \approx 0.5772156649$ is the Euler–Mascheroni constant. The associated optimal decoding deadline function takes the form

$$\mathsf{z}^*(t) = \begin{cases} 1 - \mathrm{e}^{-\lambda NT_s\zeta^*}, & 0 \leq t \leq \zeta^* \\ 1 - \mathrm{e}^{-\lambda NT_s t}, & \zeta^* < t \leq 1 \end{cases}$$

with

$$\zeta^* = \begin{cases} \frac{(\gamma + \log M)\sqrt{\bar{d}}}{\lambda NT_s}, & N < \frac{\gamma + \log M}{\lambda T_s} \\ \sqrt{\left(\frac{2(\gamma + \log M)}{\lambda NT_s} - 1\right)\bar{d}}, & N \geq \frac{\gamma + \log M}{\lambda T_s}. \end{cases}$$

Rate vs. Avg. Delay Tradeoff (K=1,T$_s$=1,λ=1)

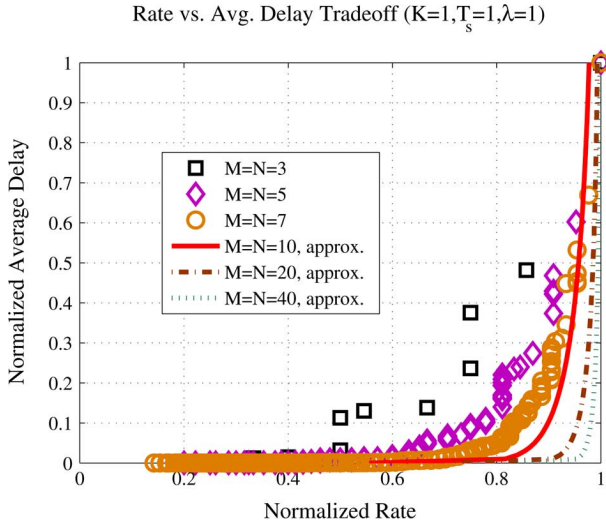Fig. 6. Exact normalized rate–delay tradeoffs for some small M, N for the i.i.d. exponential arrival time case for $M = N = 3, 5, 7$, and the approximated rate–delay tradeoff via the calculus of variations for $M = N = 10, 20, 40$.

The proof is found in Appendix G. Fig. 6 shows some example approximate rate–delay tradeoffs by plotting these expressions. The figure illustrates that significant reductions in normalized delay are achievable with only small or moderate reductions in normalized rate.

V. PRACTICAL DELAY MITIGATING CODES

This paper has addressed rate–delay tradeoffs for multipath routed channels and random linear network coded channels. For multipath routed channels we have focused on the case where the packet reordering permutation is uniform over the set of all permutations. For random linear network coded channels we have focused on the case where the encoding matrices are selected uniformly over all matrices of a given rank. The relevant abstracted DBC capacity region for both cases is presented in Section III, where we show the capacity regions of the two channels to be the same. This region is characterized by one linear constraint, and thus all capacity-achieving strategies may be formed by time sharing (across "users" of the DBC) codes which transmit to only one of the users at maximal rate for that user. In particular, we may time-share codes with rate vectors $\boldsymbol{r}_j = jK\boldsymbol{e}_j$ bits per channel use, with $\boldsymbol{e}_j$ the jth column of the identity matrix to get any rate vector $\boldsymbol{r}$ in the capacity region of the uniform permutation channel or the uniform network coded channel. Capacity-achieving codes with rate vectors $\boldsymbol{r}_j = jK\boldsymbol{e}_j$ exist for finite block lengths and appropriate K, and have the property in the multipath routed case that they encode j packets of K source bits to form at total of MK encoded bits, from any subset of jK-bit packets of which the original source bits can be determined. Such codes and are called MDS erasure codes. Alternatively, in the uniform network coded case, codes with rate vectors $\boldsymbol{r}_j = jK\boldsymbol{e}_j$ bits per channel use encode jK source bits into M packets of K encoded bits, is such a way that the source bits can be determined from any rank j linear combination of the M packets. Such codes are called rank metric MDS codes [16]. The act of time sharing such MDS codes is commonly referred

to as PET after [8], [33], and their rank metric counterparts have recently been investigated [9], [16]. Thus, in the uniform case studied in detail in this paper, codes exist which achieve the introduced rate–delay tradeoff, and they are PET codes in the permutation channel case, and rank metric PET codes in the network coded case. In the PET construction, it is left up to the user what rates the component MDS codes should have, so that the PET name is given to any construction which time-shares MDS codes, regardless of the component rates. This paper provides the necessary analysis for an end user who needs a particular bounded delay to determine the component code rates of a time shared collection of MDS codes in order to satisfy this delay. Because the PET construction in [8] is slightly different from the one presented here, namely, in that it allows the number of packets M to be dictated by other problem variables, while M is fixed here, and we allow the number of bits per packet K to be dictated by other problem variables, we will provide a brief discussion of how to construct practical PET codes in this section.

In particular, let us discuss how to obtain a code achieving a point on the rate–delay tradeoff. Suppose we have selected a delay bound d, and the rate vector $\boldsymbol{r}_d^*$ bits per channel use for the degraded broadcast channel attains the maximal bit rate for this delay, as in (1) and (2). First, we encode the (temporally) first $R_1(d)$ bits of the source data into $R_1(d)M$ bits with an MDS code. These bits are then divided up into M equally sized contiguous consecutive blocks of length $R_1(d)$, and one such block is placed at the beginning of each of the M packets. Second, we encode the next $R_2(d)$ bits of the source data into $\frac{R_2(d)M}{2}$ encoded bits using an MDS code. These bits are then divided up equally (and contiguously and consecutively) into blocks of size $\frac{R_2}{2}$ bits and one such block is placed in each of the M packets. The construction continues, i.e., at the jth step, the next $R_j(d)$ bits of source data are encoded into $\frac{R_j(d)M}{j}$ encoded bits using an MDS code, and these are broken up into M contiguous blocks of $\frac{R_j(d)}{j}$ bits, with one such block placed in one of each of the M packets.

The preceding discussion glosses over the quantization of rates that must occur and the selection of an appropriate packet length K. If we are using Reed–Solomon codes over GF(q) with $q = 2^m$ as the MDS codes, for instance, then the number of source bits $R_j(d)$ must equal $(2^m - 1 - 2t)m$ for some integer t_j, and the number of encoded bits $\frac{R_j(d)M}{j}$ must equal $(2^m - 1)m$ [45]. Furthermore, $\frac{(2^m - 1)m}{M}$ must be an integer multiple of m, so that the blocks of information to go into the different packets break up at boundaries between different elements of the finite field. One can always select the packet length K, t_j, and the field size 2^m to obey these conditions and arbitrarily closely approximate the rate–delay tradeoff. In particular, due to the form of the rate regions of the degraded broadcast channels (6)–(8), choosing a different K' for the same delay constraint simply scales the optimal rate vector associated with this delay to $\boldsymbol{r}'(d) = \frac{K'}{K}\boldsymbol{r}(d)$, and thus the total rate $\rho'(d) = \frac{K'}{K}\rho(d)$. Thus, first select any point on the *normalized* rate–delay tradeoff curve, e.g., by taking the associated K to be $K = 1$. Take the rate vector $\boldsymbol{r}$ associated with this (or a sufficiently close approximation to it that has all elements as rational numbers if some

of its elements are not rational numbers), and scale it to $\boldsymbol{r}K'$, with K' a natural number selected so that the jth element of the vector $\boldsymbol{r}K'$ is a natural number divisible by jm. For instance, given a rational $\boldsymbol{r}$ it suffices (but is not necessary) to take $K' = GmM$ where G is the least common multiple of the denominators of the rational numbers to achieve this. Then, for the jth component code with specified rate $R_j(d)$, select t_j to be

$$t_j = \left\lceil \frac{(2^m - 1)\left(1 - \frac{j}{M}\right)}{2} \right\rceil.$$

This will give a normalized rate and delay arbitrarily close to the one originally selected as m gets large, as well as an associated rate vector that is in the achievable rate region for any m.

Finally, it is important to note that, for some other, nonuniform cases, the rate regions of the degraded broadcast channels introduced in this paper are different, and PET codes do not achieve the rate–delay tradeoffs, because they are not capacity-achieving strategies for these degraded broadcast channels. The rate–delay tradeoff framework in this paper, however, still provides an upper bound on the achievable bit rate given a delay in these cases.

VI. INCORPORATING FEEDBACK AND PACKET LOSS

This paper has employed forward erasure correction based delay mitigation. However, many modern communications networks are bidirectional. One could then ask if better rate–delay tradeoffs could be achieved by the use of automatic response request (ARQ)-like schemes incorporating feedback. Surprisingly, the rate–delay tradeoffs presented are the same, even if feedback is used. Fundamentally, this is because feedback does not change the capacity region of the discrete memoryless DBC [46]. Applying this fact to our case, feedback could only affect the rate–delay tradeoffs through the associated abstracted DBC's capacity region. Thus, while the codes we have discussed have been forward erasure correction based, a scheme incorporating feedback is subject to the same rate–delay tradeoffs.

An additional natural extension points out the fact that real networks also exhibit (nonnegligible) packet loss in addition to packet reorderings and delays. Packet losses could be incorporated in the present model by simply having any lost packets arrive at a very large time $\tau \approx \infty$. A small delay bound would generate rate–delay tradeoff that would simply not require all packets to be received before all of the source frames can be decoded. Indeed, even without incorporating such large delays into the model directly, they happen in the i.i.d. $\{t_j\}$ case in the limit as $M \to \infty$ (which is the limit we investigated) due to the growth of the maximal order statistic. The optimal decoding deadlines found via the calculus of variations continous approximation in this case already do not require all of the encoded packets to arrive before all of the source frames may be decoded (effectively already handling packet loss).

An alternative approach at including packet loss by incorporating random numbers of packet losses would allow for an opportunity to discuss a more classic information-theoretic way of studying rate–delay tradeoffs: through error exponents. Note that for the abstracted degraded broadcast channels presented

in this paper, the error exponents are infinite (more precisely: after a critical block length of one channel use it is possible to create capacity-achieving codes that have zero probability of error), and thus a classic study of rate–delay tradeoffs through the use of error exponents would not have made sense. However, as we have seen, even with infinite error exponents for the associated abstracted degraded broadcast channel, there is still a rigorous and important notion of a rate–delay tradeoff. If random numbers of packet losses were incorporated, the error exponents would no longer be infinite, and a rate–delay tradeoff analysis through error exponents should become feasible. It would be interesting to see how the present results could be melded into such a setup, and is part of our plans for future work.

VII. CONCLUSION

This paper has addressed the optimal rate–delay tradeoff in multipath routed and network coded networks. In both cases, we have focused on the uniform distribution case: uniformly selected packet reordering for multipath routed networks, and uniformly selected encoding matrices for network coded networks. Both cases are modeled as an abstracted DBC, their capacity regions are shown to be the same, and the region is specified by a single linear inequality. The rate–delay tradeoff is formulated as a nonlinear continuous optimization problem, but we demonstrate the problem is in fact equivalent to an integer optimization problem, suitable for solution via enumeration for moderate M, N. For large M, N we propose an approximate solution using the calculus of variations. We solve this variational problem and obtain explicit expressions for rate–delay tradeoffs for the two special cases of constant packet inter-arrival times and i.i.d. exponentially distributed arrival times. We propose the time-shared use of practical finite block-length capacity achieving codes: PET codes for multipath routed networks, and rank-metric MDS component codes for linear network coded networks. The rates of these codes are dictated by the fundamental rate–delay tradeoffs we present, and are determined by the specified delay bound.

APPENDIX A
MATHEMATICAL NOTATION

See Table I on top of the following page.

APPENDIX B
PROOF OF PROPOSITION 1

We begin with the general expression for receiver k

$$
\begin{aligned}
\mathcal{I}(\boldsymbol{y}_k; \boldsymbol{u}_k | \boldsymbol{u}_{k-1}) & \\
&= \mathcal{I}\left((\pi(1), \boldsymbol{x}_{\pi(1)}, \dots, \pi(k), \boldsymbol{x}_{\pi(k)}); \boldsymbol{u}_k | \boldsymbol{u}_{k-1}\right) \quad (28) \\
&= \mathsf{H}\left(\pi(1), \boldsymbol{x}_{\pi(1)}, \dots, \pi(k), \boldsymbol{x}_{\pi(k)} | \boldsymbol{u}_{k-1}\right) \\
&\quad - \mathsf{H}\left(\pi(1), \boldsymbol{x}_{\pi(1)}, \dots, \pi(k), \boldsymbol{x}_{\pi(k)} | \boldsymbol{u}_k\right) \quad (29) \\
&= \mathsf{H}(\pi(1), \dots, \pi(k) | \boldsymbol{u}_{k-1}) - \mathsf{H}(\pi(1), \dots, \pi(k) | \boldsymbol{u}_k) \\
&\quad + \mathsf{H}\left(\boldsymbol{x}_{\pi(1)}, \dots, \boldsymbol{x}_{\pi(k)} | \pi(1), \dots, \pi(k), \boldsymbol{u}_{k-1}\right) \\
&\quad - \mathsf{H}\left(\boldsymbol{x}_{\pi(1)}, \dots, \boldsymbol{x}_{\pi(k)} | \pi(1), \dots, \pi(k), \boldsymbol{u}_k\right) \quad (30) \\
&= \mathsf{H}(\pi(1), \dots, \pi(k)) - \mathsf{H}(\pi(1), \dots, \pi(k)) \\
&\quad + \mathsf{H}\left(\boldsymbol{x}_{\pi(1)}, \dots, \boldsymbol{x}_{\pi(k)} | \pi(1), \dots, \pi(k), \boldsymbol{u}_{k-1}\right) \\
&\quad - \mathsf{H}\left(\boldsymbol{x}_{\pi(1)}, \dots, \boldsymbol{x}_{\pi(k)} | \pi(1), \dots, \pi(k), \boldsymbol{u}_k\right) \quad (31)
\end{aligned}
$$

TABLE I
MATHEMATICAL NOTATION

N	number of source frames
M	number of packets sent over network
K	length in bits of each packet
$\mathbf{s}_1, \ldots, \mathbf{s}_N$	temporally ordered frames from a multimedia source encoder
$\mathbf{x}_1, \ldots, \mathbf{x}_M$	packets sent over network
$t_1, \ldots, t_M$	packet arrival times
$\pi(j)$	index of the transmitted packet that is the jth received packet
$\pi = (\pi(1), \ldots, \pi(M))$	permutation of the packets capturing packet reordering
$\tau_j = t_{\pi(j)}$	time of jth received packet
$\tau_1 < \tau_2 < \ldots < \tau_M$	packet arrival times in order of reception
$\mathsf{p}(t_1, \ldots, t_M)$	joint distribution on packet arrival times
$\mathsf{p}(\tilde{\pi}),\ \tilde{\pi} \in \Pi$	distribution on the packet reordering permutation
$\mathbf{x}_{\pi(1)}, \mathbf{x}_{\pi(2)}, \ldots, \mathbf{x}_{\pi(M)}$	packets in order of reception
$\mathsf{p}(\tau_1, \ldots, \tau_M \vert \pi)$	joint distribution on receiver–ordered packet arrival times
$\mathcal{R}$	capacity region of degraded broadcast channel
$\mathbf{r} = (R_1, \ldots, R_M),\ \mathbf{r} \in \mathcal{R}$	a rate vector in the capacity region
R_j	amount of new information (in bits) available upon the jth received packet
T_s	desired time between successive playback of the temporally ordered frames
$\rho(\mathbf{r}) = \frac{1}{NT_s} \sum_{j=1}^{M} R_j$	overall rate at which data is decoded at the source decoder
$D_p(\mathbf{r}) = \sum_{i=1}^{N} \left((\tau_{g(i,\mathbf{r})} - iT_s)^+ \right)^p$	overall random delay for rate vector $\mathbf{r}$
$\delta(\mathbf{r}) = \mathbb{E}[D_p(\mathbf{r})]$	overall expected delay associated with rate vector $\mathbf{r}$
$\mathbf{y}_j = (\pi(1), \mathbf{x}_{\pi(1)}, \pi(2), \mathbf{x}_{\pi(2)}, \ldots, \pi(j), \mathbf{x}_{\pi(j)})$	cumulative information available at destination upon the jth received packet
$\mathbf{u}_1 \to \cdots \to \mathbf{u}_{M-1} \to \mathbf{x}$	dummy discrete random variables characterizing the DBC capacity region $\mathcal{R}$
$\mathbf{g}_k$	random global encoding vector (using network coding) of the kth received packet
$\mathbf{G}_k = [\mathbf{g}_1^T, \mathbf{g}_2^T, \ldots, \mathbf{g}_k^T]$	random global encoding matrix
$\mathcal{G}_k$	set of global encoding matrices of rank k from $\mathrm{GF}(q)^{k \times M}$
$p_k(\tilde{\mathbf{G}}_k),\ \tilde{\mathbf{G}}_k \in \mathcal{G}_k$	probability distribution on the set of global encoding matrices
$g(i, \mathbf{r}) = \inf \left\{ n \left\vert \sum_{j=1}^{n} R_j \geq iT_s \rho(\mathbf{r}) \right. \right\}$	# of packets that must be received before the first i frames may be decoded
$\mathbf{h} = (h(1), \ldots, h(N))$	decoding deadline for each $g(i, \mathbf{r})$; *i.e.*, each $g(i, \mathbf{r}) \leq h(i)$
$\rho_d^* = \max_{\mathbf{r} \in \mathcal{R}} \{ \rho(\mathbf{r}) \,\vert\, \delta(\mathbf{r}) \leq d \}$	max. feasible overall rate subject to a bound d on expected overall delay
$\bar{\rho}(\mathbf{h}) = \frac{NT_s}{MK} \rho(\mathbf{h})$	overall rate ρ normalized to lie in $[0, 1]$ for decoding deadlines $\mathbf{h}$
$\bar{\delta}(\mathbf{h}) = \frac{\mathbb{E}[D_p(\mathbf{h})]}{\mathbb{E}[D(\{h(i)=M\ \forall i \in \{1, \ldots, N\}\})]}$	overall delay normalized to lie in $[0, 1]$ for decoding deadlines $\mathbf{h}$
$\mathbf{z}(t) = \sum_{i=1}^{N} \frac{h(i)}{M} \mathbf{1}_{\frac{i-1}{N} \leq t < \frac{i}{N}}$	zero order hold piecewise continuous fcn. $\mathbf{z} : [0, 1] \to [0, 1]$ s.t. $h(i) = M\mathbf{z}(i/N)$
$\bar{\rho}[\mathbf{z}] = \left(\int_0^1 \frac{1}{\mathbf{z}(t)} \mathrm{d}t \right)^{-1}$	overall rate ρ normalized to lie in $[0, 1]$ for approximate decoding deadlines $\mathbf{z}$
$\bar{d}[\mathbf{z}] = \int_0^1 \mathsf{Q}(t, \mathbf{z}(t)) \mathrm{d}t$	overall norm. delay for approx. decoding deadlines $\mathbf{z}(\cdot)$ with integrand $\mathsf{Q}(t, \mathbf{z}(t))$
$\bar{\rho}_d^* = \max_{\mathbf{z}:[0,1] \to [0,1]} \left\{ \bar{\rho}[\mathbf{z}] \left\vert \bar{\delta}[\mathbf{z}] < \bar{d},\ \frac{\partial \mathbf{z}(t)}{\partial t} \geq 0 \right. \right\}$	optimal overall normalized rate subject to overall normalized delay requirement $\bar{d}$.

$$= \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \left(\mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} \vert \boldsymbol{u}_{k-1} \right) \right.$$
$$\left. - \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} \vert \boldsymbol{u}_k \right) \right). \tag{32}$$

Here, (28) substitutes in the definition of $\boldsymbol{y}_k$ and (29) is the definition of condition mutual information together with the fact that $\boldsymbol{u}_1 \to \boldsymbol{u}_2 \to \boldsymbol{u}_{M-1} \to \boldsymbol{x}$ form a Markov chain. Next, (30) expresses a joint entropy as a marginal entropy plus a conditional entropy (e.g., $\mathsf{H}(U, V) = \mathsf{H}(U) + \mathsf{H}(V \vert U)$). The process selecting the random reordering variables $(\pi(1), \ldots, \pi(M))$, which comes from the discrete memoryless channel, has nothing to do with the contents of the input block of packets $\boldsymbol{x}$ or the message random variables, so these are independent of one another, giving equation (31). Substituting in the definition of conditional entropy and recognizing that the contents of the received packets only depend on the arrival order through their index then gives the desired expression (32). We emphasize the distinction in the last step between the random variables π and a particular realization (permutation) $\tilde{\pi} \in \Pi$, for Π the set of permutations of the numbers $1, \ldots, M$. The equivalences for the other two inequalities (6) and (8) follow from essentially the same chain of arguments.

APPENDIX C
PROOF OF THEOREM 2: UNIFORM PERMUTATION CHANNEL
CAPACITY REGION

Han's inequality ([47, Theorem 17.6.1, p. 668], citing [48]) states

$$\frac{1}{k \binom{M}{k}} \sum_{S : \vert S \vert = k} \mathsf{H}(X_S)$$
$$\leq \frac{1}{(k-1) \binom{M}{k-1}} \sum_{S : \vert S \vert = k-1} \mathsf{H}(X_S) \tag{33}$$

for random variables $X_1, \ldots, X_M$ and for each $k = 2, \ldots, M$ and where X_S is shorthand for $X_i, i \in S$. Proposition 3 specializes Han's inequality to the uniform permutation channel.

Proposition 3: For the uniform permutation channel

$$\frac{1}{k} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} \vert \boldsymbol{u}_{k-1} \right)$$
$$\leq \frac{1}{k-1} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} \vert \boldsymbol{u}_{k-1} \right) \tag{34}$$

for each $k \in \{2, \ldots, M\}$.

Consider the left-hand side of (34). By assumption, the distribution on the permutations is uniform: $\mathsf{p}(\tilde{\pi}) = \frac{1}{|\Pi|}$ for each $\tilde{\pi} \in \Pi$. There are $M!$ different permutations, and hence $\mathsf{p}(\tilde{\pi}) = \frac{1}{M!}$. However, the entropy depends only upon the unordered set of the first k elements of each permutation, $\{\tilde{\pi}(1), \ldots, \tilde{\pi}(k)\}$. Define C_k to be the set of all subsets from $\{1, \ldots, M\}$ of size k. We may then group together all $\tilde{\pi}$ with a common unordered set of first k elements. There are $\binom{M}{k}$ distinct such sets, and thus each group of terms has $M!/\binom{M}{k}$ elements. This allows

$$\frac{1}{k} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_{k-1}\right)$$
$$= \frac{1}{k \binom{M}{k}} \sum_{\{\tilde{\pi}(1), \ldots, \tilde{\pi}(k)\} \in \mathsf{C}_k} \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_{k-1}\right).$$

Repeating this procedure for $k-1$ (with the entropy still conditioned on $\boldsymbol{u}_{k-1}$, as above) gives

$$\frac{1}{k-1} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} | \boldsymbol{u}_{k-1}\right)$$
$$= \sum_{\{\tilde{\pi}(1), \ldots, \tilde{\pi}(k-1)\} \in \mathsf{C}_{k-1}} \frac{\mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} | \boldsymbol{u}_{k-1}\right)}{(k-1) \binom{M}{k-1}}.$$

The proposition follows by Han's inequality (33).

Proposition 4: For the uniform permutation channel

$$\sum_{j=1}^{k} \frac{R_j}{j} \leq \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}\right) \tag{35}$$
$$- \frac{1}{k} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_k\right) \tag{36}$$

for $k \in \{1, \ldots, M\}$.

The proof is by induction on k. The base case $k = 1$ is true by (6) in Proposition 1

$$R_1 \leq \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}\right) - \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)} | \boldsymbol{u}_1\right). \tag{37}$$

Suppose the proposition is true for $k-1$

$$\sum_{j=1}^{k-1} \frac{R_j}{j} \leq \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}\right)$$
$$- \frac{1}{k-1} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} | \boldsymbol{u}_{k-1}\right).$$

The proposition follows by successive application of the induction hypothesis, (7) in Propositions 1 and 3:

$$\sum_{j=1}^{k} \frac{R_j}{j} = \frac{R_k}{k} + \sum_{j=1}^{k-1} \frac{R_j}{j} \leq \frac{R_k}{k} + \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}\right)$$
$$- \frac{1}{k-1} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} | \boldsymbol{u}_{k-1}\right)$$
$$\leq \frac{1}{k} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \left(\mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_{k-1}\right)\right.$$
$$\left. - \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_k\right)\right) + \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}\right)$$

$$- \frac{1}{k-1} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} | \boldsymbol{u}_{k-1}\right)$$
$$\leq \frac{1}{k-1} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} | \boldsymbol{u}_{k-1}\right)$$
$$- \frac{1}{k} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_k\right)$$
$$+ \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}(\boldsymbol{x}_{\tilde{\pi}(1)})$$
$$- \frac{1}{k-1} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k-1)} | \boldsymbol{u}_{k-1}\right)$$
$$= \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}\right)$$
$$- \frac{1}{k} \sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}\left(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(k)} | \boldsymbol{u}_k\right)$$

This proves Proposition 4. The converse in Theorem 2 follows from Proposition 4 by setting $k = M$ and by noting that

$$\sum_{\tilde{\pi} \in \Pi} \mathsf{p}(\tilde{\pi}) \mathsf{H}(\boldsymbol{x}_{\tilde{\pi}(1)}) \leq K$$

and that $\mathsf{H}(\boldsymbol{x}_{\tilde{\pi}(1)}, \ldots, \boldsymbol{x}_{\tilde{\pi}(M)} | \boldsymbol{u}_M) = 0$. The remainder of the proof must show that these rates are achievable, i.e., that the inequality is tight. This may be established by noting that the capacity region is described by a single inequality. Furthermore, the corner points $\boldsymbol{r}_j = jK\boldsymbol{e}_j$ of this single inequality described rate region can be achieved using capacity-achieving codes for the point-to-point channel between the source and the ith reciever alone. These codes may then be time shared to yield all points on the boundary of the capacity region. The code for the jth receiver alone (which has rate point $\boldsymbol{r}_j = jK\boldsymbol{e}_j$) exists by the standard proof of the channel capacity for a discrete memoryless point-to-point channel, e.g., by employing random codes and jointly typical sequence decoding.

APPENDIX D
PROOF OF THEOREM 3: UNIFORM NETWORK CODED CHANNEL CAPACITY REGION

The proof of the theorem depends upon the following proposition. Note that all matrices discussed below have elements in the finite field GF(q).

Proposition 5: For the uniform network coded DBC

$$\frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$
$$\leq \frac{1}{k-1} \sum_{\tilde{\boldsymbol{G}}_{k-1} \in \mathcal{G}_{k-1}} \mathsf{p}(\tilde{\boldsymbol{G}}_{k-1}) \mathsf{H}(\tilde{\boldsymbol{G}}_{k-1} \boldsymbol{x} | \boldsymbol{u}_{k-1}) \tag{38}$$

for $k \in \{2, \ldots, M\}$.

Consider the left-hand side of (38). We decompose $\mathsf{p}(\tilde{\boldsymbol{G}}_k)$ as

$$\mathsf{p}(\tilde{\boldsymbol{G}}_k) = \sum_{\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M} \mathsf{p}(\tilde{\boldsymbol{G}}_k | \tilde{\boldsymbol{G}}_M) \mathsf{p}(\tilde{\boldsymbol{G}}_M).$$

By the uniform assumption $\mathsf{p}(\tilde{\boldsymbol{G}}_M) = \frac{1}{|\mathcal{G}_M|}$ for each $\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M$. Substitution and exchanging the order of summation yields

$$\frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$= \frac{1}{|\mathcal{G}_M|} \frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \left(\sum_{\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M} \mathsf{p}(\tilde{\boldsymbol{G}}_k | \tilde{\boldsymbol{G}}_M) \right) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$= \frac{1}{|\mathcal{G}_M|} \sum_{\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M} \frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k | \tilde{\boldsymbol{G}}_M) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

The conditional distribution $\mathsf{p}(\tilde{\boldsymbol{G}}_k | \tilde{\boldsymbol{G}}_M)$ is the probability of randomly selecting (up to a row reordering) the $k \times M$ submatrix $\tilde{\boldsymbol{G}}_k$ from the $M \times M$ matrix $\tilde{\boldsymbol{G}}_M$

$$\mathsf{p}(\tilde{\boldsymbol{G}}_k | \tilde{\boldsymbol{G}}_M) = \begin{cases} \frac{1}{\binom{M}{k}}, & \tilde{\boldsymbol{G}}_k \subseteq \tilde{\boldsymbol{G}}_M \\ 0, & \text{else.} \end{cases} \quad (39)$$

This allows

$$\frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$= \frac{1}{|\mathcal{G}_M|} \sum_{\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M} \frac{1}{k \binom{M}{k}} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k : \tilde{\boldsymbol{G}}_k \subseteq \tilde{\boldsymbol{G}}_M} \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

The inner sum over all $\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k : \tilde{\boldsymbol{G}}_k \subseteq \tilde{\boldsymbol{G}}_M$ is the same as summing over all subsets of $\{1, \ldots, M\}$ of size k and selecting the corresponding rows from $\tilde{\boldsymbol{G}}_M$. That is, define $[\tilde{\boldsymbol{G}}_M]_{\mathcal{S}}$ to be the rows from $\tilde{\boldsymbol{G}}_M$ with indices in $\mathcal{S}$, where $\mathcal{S} \subseteq \{1, \ldots, M\}$ and $|\mathcal{S}| = k$. This yields

$$\frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$= \frac{1}{|\mathcal{G}_M|} \sum_{\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M} \frac{1}{k \binom{M}{k}} \sum_{\mathcal{S} \subset \{1,\ldots,M\} : |\mathcal{S}|=k} \mathsf{H}([\tilde{\boldsymbol{G}}_M]_{\mathcal{S}} \boldsymbol{x} | \boldsymbol{u}_{k-1}).$$

Finally, for each $\boldsymbol{x}$ and $\tilde{\boldsymbol{G}}_M$ define the M random variables $\boldsymbol{q} = \tilde{\boldsymbol{G}}_M \boldsymbol{x}$ and the corresponding subset $\boldsymbol{q}_{\mathcal{S}} = [\tilde{\boldsymbol{G}}_M \boldsymbol{x}]_{\mathcal{S}}$ formed by selecting elements from $\boldsymbol{q}$ with indices in $\mathcal{S}$. This allows

$$\frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$= \frac{1}{|\mathcal{G}_M|} \sum_{\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M} \frac{1}{k \binom{M}{k}} \sum_{\mathcal{S} \subset \{1,\ldots,M\} : |\mathcal{S}|=k} \mathsf{H}(\boldsymbol{q}_{\mathcal{S}} | \boldsymbol{u}_{k-1}).$$

Performing the same sequence of steps for $k - 1$ (still conditioning on $\boldsymbol{u}_{k-1}$, as above) gives

$$\frac{1}{k-1} \sum_{\tilde{\boldsymbol{G}}_{k-1} \in \mathcal{G}_{k-1}} \mathsf{p}(\tilde{\boldsymbol{G}}_{k-1}) \mathsf{H}(\tilde{\boldsymbol{G}}_{k-1} \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$= \frac{1}{|\mathcal{G}_M|} \sum_{\tilde{\boldsymbol{G}}_M \in \mathcal{G}_M} \frac{1}{(k-1) \binom{M}{k-1}} \sum_{\mathcal{S} \subset \{1,\ldots,M\} : |\mathcal{S}|=k-1} \mathsf{H}(\boldsymbol{q}_{\mathcal{S}} | \boldsymbol{u}_{k-1}).$$

Applying Han's inequality (33) for each $\tilde{\boldsymbol{G}}_M$ completes the proof.

Proposition 6: For the uniform network coded DBC

$$\sum_{j=1}^{k} \frac{R_j}{j} \leq \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x}) - \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \frac{\mathsf{p}(\tilde{\boldsymbol{G}}_k)}{k} \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_k) \quad (40)$$

for $k \in \{1, \ldots, M\}$.

The proof is by induction on k and is identical to the proof of Proposition 4 (*mutatis mutandis*), but is included here for completeness. The base case $k = 1$ is true by (10)

$$R_1 \leq \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x}) - \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x} | \boldsymbol{u}_1). \quad (41)$$

Suppose the proposition is true for $k - 1$, then the proposition follows by successive application of the induction hypothesis, (11), and Proposition 5

$$\sum_{j=1}^{k} \frac{R_j}{j} = \frac{R_k}{k} + \sum_{j=1}^{k-1} \frac{R_j}{j}$$

$$\leq \frac{R_k}{k} + \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x})$$

$$- \frac{1}{k-1} \sum_{\tilde{\boldsymbol{G}}_{k-1} \in \mathcal{G}_{k-1}} \mathsf{p}(\tilde{\boldsymbol{G}}_{k-1}) \mathsf{H}(\tilde{\boldsymbol{G}}_{k-1} \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$\leq \frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) (\mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_{k-1}) - \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_k))$$

$$+ \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x})$$

$$- \frac{1}{k-1} \sum_{\tilde{\boldsymbol{G}}_{k-1} \in \mathcal{G}_{k-1}} \mathsf{p}(\tilde{\boldsymbol{G}}_{k-1}) \mathsf{H}(\tilde{\boldsymbol{G}}_{k-1} \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$\leq \frac{1}{k-1} \sum_{\tilde{\boldsymbol{G}}_{k-1} \in \mathcal{G}_{k-1}} \mathsf{p}(\tilde{\boldsymbol{G}}_{k-1}) \mathsf{H}(\tilde{\boldsymbol{G}}_{k-1} \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$- \frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_k) + \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x})$$

$$- \frac{1}{k-1} \sum_{\tilde{\boldsymbol{G}}_{k-1} \in \mathcal{G}_{k-1}} \mathsf{p}(\tilde{\boldsymbol{G}}_{k-1}) \mathsf{H}(\tilde{\boldsymbol{G}}_{k-1} \boldsymbol{x} | \boldsymbol{u}_{k-1})$$

$$= \sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x}) - \frac{1}{k} \sum_{\tilde{\boldsymbol{G}}_k \in \mathcal{G}_k} \mathsf{p}(\tilde{\boldsymbol{G}}_k) \mathsf{H}(\tilde{\boldsymbol{G}}_k \boldsymbol{x} | \boldsymbol{u}_k).$$

This proves Proposition 6. The converse in Theorem 3 follows from Proposition 6 by setting $k = M$ and by noting that $\sum_{\tilde{\boldsymbol{G}}_1 \in \mathcal{G}_1} \mathsf{p}(\tilde{\boldsymbol{G}}_1) \mathsf{H}(\tilde{\boldsymbol{G}}_1 \boldsymbol{x}) \leq K$ and that $\mathsf{H}(\tilde{\boldsymbol{G}}_M \boldsymbol{x} | \boldsymbol{u}_M) = 0$. The proof of achievability is exactly the same as the achievability proof found at the end of Appendix C.

APPENDIX E
PROOF OF THEOREM 4: STRUCTURAL PROPERTIES OF RATE DELAY TRADEOFF

The decoding deadline sequence $\boldsymbol{h} = (h(i), i = 1, \ldots, N)$ is the minimum number of packets, say j, that must be received before the receiver is guaranteed to be able to decode the first

i frames. Thus, $h : \{1, \ldots, N\} \rightarrow \{1, \ldots, M\}$ maps frame indices to packet reception counts (deadlines). The deadlines are by necessity nondecreasing: $h(i) \leq h(i+1)$. Because the delay metric $D_p(\boldsymbol{r})$ depends on $\boldsymbol{r}$ only through $g(i, \boldsymbol{r})$, we can break the optimization of $\boldsymbol{r}$ up into two coupled optimization problems, so that the optimal rate vector $\boldsymbol{r}^*$ is the solution to

$$\rho(\boldsymbol{r}^*) = \max_{\boldsymbol{h}|\mathbb{E}[D(\boldsymbol{h})] \leq d} \quad \max_{\boldsymbol{r} \in \mathcal{R}|g(i,\boldsymbol{r})=h(i), i=1,\ldots,N} \rho(\boldsymbol{r}). \quad (42)$$

We rearrange the definition of $g(i, \boldsymbol{r})$ (15) using the definition of $\rho(\boldsymbol{r})$ (13) as

$$g(i, \boldsymbol{r}) = \inf \left\{ n \Big| \frac{\sum_{j=1}^{n} R_j}{\sum_{j=1}^{M} R_j} \geq \frac{i}{N} \right\}.$$

We can see that $\rho(\boldsymbol{r})$ depends only on the magnitude ($\| \cdot \|_1$) of $\boldsymbol{r}$, while $g(i, \boldsymbol{r})$ depends only on the direction in which $\boldsymbol{r}$ points. The constraint (9), on the other hand, depends on both its magnitude and direction. This observation motivates a change of variables from $\boldsymbol{r}$ to $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_M)$ and ρ such that $\gamma_j \geq 0, \sum_{j=1}^{M} \gamma_j = 1$ and $\rho = \rho(\boldsymbol{r})$. In particular, set

$$\rho = \rho(\boldsymbol{r}), \gamma_j = \frac{R_j}{\sum_{k=1}^{M} R_k} = \frac{R_j}{\rho N T_s}, \quad j = 1, \ldots, M. \quad (43)$$

The constraints $\boldsymbol{r} \in \mathcal{R}$ and $g(i, \boldsymbol{r}) = h(i)$ expressed in terms of $(\boldsymbol{\gamma}, \rho)$ become

$$\boldsymbol{r} \in \mathcal{R} \rightarrow \sum_{j=1}^{M} \frac{\gamma_j}{j} \leq \frac{K}{N T_s \rho}$$

$$g(i, \boldsymbol{r}) = h(i) \rightarrow \inf \left\{ n \Big| \sum_{j=1}^{n} \gamma_j \geq \frac{i}{N} \right\} = h(i)$$

for $i \in \{1, \ldots, N\}$. The inner maximization in (44) expressed in terms of $(\boldsymbol{\gamma}, \rho)$ becomes

$$\max_{\boldsymbol{\gamma}, \rho} \left\{ \rho \Big| \begin{array}{c} \sum_{j=1}^{M} \gamma_j = 1, \boldsymbol{\gamma} \geq \boldsymbol{0}, \sum_{j=1}^{M} \frac{\gamma_j}{j} \leq \frac{K}{N T_s \rho}, \\ \min \left\{ n \Big| \sum_{j=1}^{n} \gamma_j \geq \frac{i}{N} \right\} = h(i), i = 1, \ldots, N \end{array} \right\}. \quad (44)$$

For fixed $\boldsymbol{\gamma}$, the optimal ρ in (44) is

$$\rho = \frac{K}{N T_s} \left(\sum_{j=1}^{M} \frac{\gamma_j}{j} \right)^{-1}. \quad (45)$$

Note that we may undo the change of variables via

$$R_j = \rho N T_s \gamma_j = K \frac{\gamma_j}{\sum_{k=1}^{M} \frac{\gamma_k}{k}}, \quad j = 1, \ldots, M. \quad (46)$$

Each decoding constraint can be decomposed into a pair of equations

$$\inf \left\{ n \Big| \sum_{j=1}^{n} \gamma_j \geq \frac{i}{N} \right\} = h(i) \Leftrightarrow$$

$$\left(\sum_{j=1}^{h(i)-1} \gamma_j < \frac{i}{N} \text{and} \sum_{j=1}^{h(i)} \gamma_j \geq \frac{i}{N} \right)$$

for all $i \in \{1, \ldots, N\}$. Applying this decomposition and substituting the optimal ρ transforms the inner maximization in (42) to a minimization of $\sum_{j=1}^{M} \frac{\gamma_j}{j}$

$$\min_{\boldsymbol{\gamma}} \left\{ \sum_{j=1}^{M} \frac{\gamma_j}{j} \Bigg| \begin{array}{c} \sum_{j=1}^{M} \gamma_j = 1, \boldsymbol{\gamma} \geq \boldsymbol{0}, \sum_{j=1}^{h(i)-1} \gamma_j < \frac{i}{N}, \\ \sum_{j=1}^{h(i)} \gamma_j \geq \frac{i}{N}, i = 1, \ldots, N \end{array} \right\}. \quad (47)$$

In the absence of any decoding constraints, the objective $\sum_{j=1}^{M} \frac{\gamma_j}{j}$ is minimized over all nonnegative $\boldsymbol{\gamma}$ summing to one by setting $\gamma_j = 0$ for $j = 1, \ldots, M-1$, and $\gamma_M = 1$. The optimal solution is thus to make each γ_j as small as possible while obeying the decoding constraints $\boldsymbol{h}$. To handle these decoding constraints it is convenient to define several quantities associated with $\boldsymbol{h}$

$$\begin{aligned} n &= \#\{j : h(i) = j, i = 1, \ldots, N\} \\ \boldsymbol{l} &= (l_1 < l_2 < \cdots < l_n) \\ &= \{j : h(i) = j, i = 1, \ldots, N\} \\ h^{-1}(j) &= \max\{i : h(i) = j\}, j \in \boldsymbol{l} \\ \boldsymbol{k} &= (0 = k_0 < k_1 < \cdots < k_n = N) \\ &= \{h^{-1}(j), j \in \boldsymbol{l}\} \cup \{0\} \\ &= \{i : h(i) < h(i+1), i = 1, \ldots, N\} \cup \{0, N\} \\ \boldsymbol{\Delta} &= (\Delta_1 = k_1 - 0, \Delta_2 = k_2 - k_1, \ldots, \Delta_n = N - k_{n-1}). \end{aligned}$$

In words, n is the number of distinct deadlines, $\boldsymbol{l}$ is the vector of distinct deadlines, $h^{-1}(j)$ is the last frame index associated with each deadline $j \in \boldsymbol{l}$, $\boldsymbol{k}$ is the vector of last frame indices for each deadline augmented with $k_0 = 0$, and $\boldsymbol{\Delta}$ is the vector of differences between the elements in $\boldsymbol{k}$. These quantities are illustrated in Fig. 7. Note that $n \leq \min\{M, N\}$ for all $\boldsymbol{h}$. To minimize $\sum_{j=1}^{M} \frac{\gamma_j}{j}$ subject to decoding constraints, the optimal choice for $\boldsymbol{\gamma}$ is to set

$$\gamma_{h(k_m)}^* = \frac{k_m - k_{m-1}}{N} = \frac{\Delta_m}{N}, m = 1, \ldots, n \quad (48)$$

and set all other γ_j^* to zero. This assignment satisfies the decoding constraints while delaying assigning the mass in γ to as high an index as possible. Note that this assignment satisfies the constraint that $\boldsymbol{\gamma} \geq \boldsymbol{0}$ and the sum to one constraint

$$\sum_{j=1}^{M} \gamma_j^* = \sum_{m=1}^{n} \gamma_{h(k_m)}^* = \sum_{m=1}^{n} \frac{k_m - k_{m-1}}{N} = \frac{k_n - k_0}{N} = 1. \quad (49)$$

Further, the objective in (47) becomes

$$\sum_{j=1}^{M} \frac{\gamma_j^*}{j} = \sum_{m=1}^{n} \frac{\gamma_{h(k_m)}^*}{h(k_m)} = \frac{1}{N} \sum_{m=1}^{n} \frac{\Delta_m}{l_m} = \frac{1}{N} \sum_{m=1}^{n(\boldsymbol{h})} \frac{\Delta(\boldsymbol{h})_m}{l(\boldsymbol{h})_m} \quad (50)$$

where the last expression makes explicit that n, Δ_m, l_m are each computed from the decoding deadlines $\boldsymbol{h}$. The associated maximum sum rate under $\boldsymbol{h}$ is then

$$\rho(\boldsymbol{h}) = \frac{K}{N T_s} \left(\sum_{j=1}^{M} \frac{\gamma_j^*}{j} \right)^{-1}$$
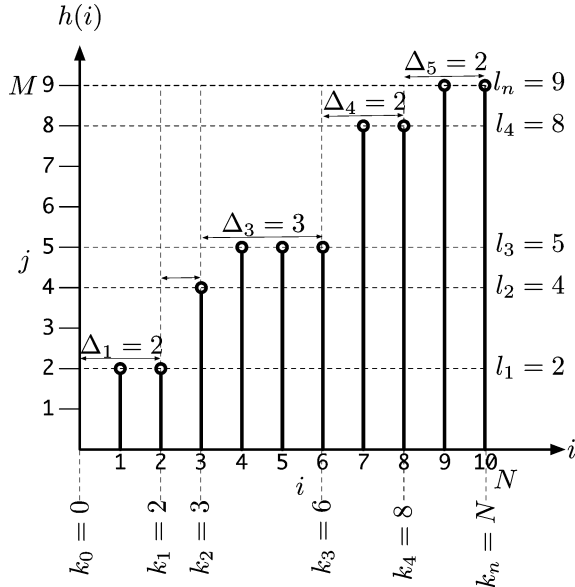
Fig. 7. Illustration of the decoding deadline sequence $\boldsymbol{h} = (h(1), \ldots, h(N))$, where $h(i)$ is the number of packets that must be received to guarantee that frame i may be decoded for the case $N = 10$ frames and $M = 9$ packets. Using the notation $n, \boldsymbol{l}, \boldsymbol{k}, \Delta$ introduced in the proof of the theorem, there are $n = 5$ distinct deadlines, namely, $\boldsymbol{l} = (l_1, \ldots, l_n) = (2, 4, 5, 8, 9)$, and the last frame indices associated with each deadline are $\boldsymbol{k} = (k_0, \ldots, k_n) = (0, 2, 3, 6, 8, 10)$. The duration of each deadline is given as $\Delta = (\Delta_1, \ldots, \Delta_n) = (2, 1, 3, 2, 2)$.

$$= \frac{K}{T_s} \left(\sum_{m=1}^{n(\boldsymbol{h})} \frac{\Delta(\boldsymbol{h})_m}{l(\boldsymbol{h})_m} \right)^{-1}. \tag{51}$$

Finally, the rate–delay tradeoff (16) expressed in terms of the decoding deadlines is

$$\max_{\boldsymbol{h}: \mathbb{E}[D_p(\boldsymbol{h})] \leq d} \frac{K}{T_s} \left(\sum_{m=1}^{n(\boldsymbol{h})} \frac{\Delta(\boldsymbol{h})_m}{l(\boldsymbol{h})_m} \right)^{-1}. \tag{52}$$

The remaining optimization is solved by exhaustive enumeration over all decoding constraints that satisfy the delay constraint d. We have thus shown that the rate–delay tradeoff, although correctly posed as a continuous optimization, can be found as the solution to an integer programming problem in the decoding deadlines $\boldsymbol{h}$ in the uniform case.

APPENDIX F
PROOF OF THEOREM 6: CONSTANT INTER-ARRIVAL TIME APPROXIMATE RATE–DELAY TRADEOFF

The calculus of variations problem (21) for normalized delay bound $\bar{d} \in [0, 1]$ and $p = 1$ in (19) may be written as

$$\frac{1}{\bar{\rho}_{\bar{d}}^*} = \inf_{z \in \mathcal{Z}(\bar{d})} \int_0^1 \frac{1}{z(t)} dt$$

where $\mathcal{Z}(\bar{d})$ is the set of differentiable $z : [0, 1] \to [0, 1]$ obeying

$$\int_0^1 (z(t) - t)^+ dt \leq \frac{M-1}{2M} \bar{d}, \quad \frac{\partial z(t)}{\partial t} \geq 0.$$

Note that the objective function is monotone decreasing in $z(t)$, and if $z(t) < t$, the delay does not change by changing it to $z(t) = t$. This allows

$$\frac{1}{\bar{\rho}_{\bar{d}}^*} = \min_{z \in \mathcal{Z}(\bar{d}),\, z(t) \geq t} \int_0^1 \frac{1}{z(t)} dt.$$

The Lagrangian is

$$\mathcal{L}(z, \mu, a, b) = \int_0^1 \frac{1}{z(t)} dt - \mu \left(\int_0^1 (z(t) - t) dt - \frac{M-1}{2M} \bar{d} \right)$$
$$+ \int_0^1 a(t) \frac{\partial z(t)}{\partial t} dt + \int_0^1 b(t)(z(t) - t) dt \tag{53}$$

where $a(t)$ is the Lagrange multiplier on the constraint $\frac{\partial z(t)}{\partial t} \geq 0$ at t and $b(t)$ is the multiplier on the constraint $z(t) \geq t$ at t, for each $t \in [0, 1]$. The solution to the variational problem (53) must satisfy the algebraic equation, obtained by differentiating (53) with respect to $z(t)$ at each $t = t_0$

$$-\frac{1}{(z^*(t_0))^2} - \mu + \alpha(t_0) \mathbf{1}_{\{\frac{\partial z^*(t_0)}{\partial t}\}} + \beta(t_0) \mathbf{1}_{\{z(t_0) = t_0\}} = 0$$

for $t_0 \in [0, 1]$. Here $\alpha(t_0)$ and $\beta(t_0)$ are defined to be nonzero when the slope of $z(t_0)$ is zero or $z(t_0) = t_0$, respectively. These constraints dictate the values of $\alpha(t_0)$ and $\beta(t_0)$, although it is not necessary to solve for them explicitly. Solving this for z^*, we see that a necessary condition for z^* to be an extremal for the constrained optimization is

$$z^*(t_0) = (\mu + \alpha(t_0) \mathbf{1}_{\{\frac{\partial z^*(t_0)}{\partial t} = 0\}} + \beta(t_0) \mathbf{1}_{\{z^*(t_0) = t_0\}})^{-\frac{1}{2}}$$

for each $t_0 \in [0, 1]$. From this equation it follows that z^* will be a piecewise function that is either constant or increasing at unit slope, i.e., $z(t) = t$. Furthermore, the inverse rate objective function is decreased more by having the constant portion of the function for small t. Thus, the optimal z^* takes the form

$$z^*(t) := \begin{cases} \zeta, & 0 \leq t \leq \zeta \\ t, & \zeta < t \leq 1 \end{cases}$$

for some $\zeta \in [0, 1]$. This gives a normalized rate of

$$\bar{\rho} = \left(\int_0^1 \frac{1}{z^*(t)} dt \right)^{-1} = (1 - \log(\zeta))^{-1}$$

and a normalized delay of

$$\bar{\delta} = \frac{2M}{M-1} \left(\zeta^2 - \frac{\zeta^2}{2} \right) = \frac{M\zeta^2}{M-1}.$$

Since $\zeta = \sqrt{\frac{M-1}{M} \bar{\delta}}$, the approximate (asymptotically accurate) rate–delay tradeoff for delay bound $\bar{d}$ and rate bound $\bar{\rho}$ is

$$\bar{\rho}_{\bar{d}}^* = \left(1 - \log \left(\sqrt{\frac{M-1}{M} \bar{d}} \right) \right)^{-1} \tag{54}$$

$$\bar{\delta}_{\bar{\rho}}^* = \frac{M}{M-1} \exp \left(2 - \frac{2}{\bar{\rho}} \right). \tag{55}$$

APPENDIX G

PROOF OF THEOREM 7: I.I.D. EXPONENTIAL ARRIVAL TIME
APPROXIMATE RATE–DELAY TRADEOFF

The fundamental rate–delay tradeoff (21) for i.i.d. exponential arrival times with parameter λ and with approximate normalized delay functional (27) is

$$\min_{z \in \mathcal{Z}'(\bar{d})} \int_0^1 \frac{1}{z(t)} dt \quad (56)$$

where $\mathcal{Z}'(\bar{d})$ is the set of differentiable $z : [0,1] \to [0,1]$ obeying

$$\int_0^1 (-\lambda^{-1} \log(1 - z(t)) - tNT_s)^+ dt \le c(M)\bar{d} \quad (57)$$

$$\frac{\partial z(t)}{\partial t} \ge 0$$

and

$$c(M) = \int_0^1 \mathbb{E}[(\tau_M - NT_s t)^+] dt$$

is the denominator in (19) (for $p = 1$) and is independent of the functional z. Since the inverse rate (which we want to minimize) is monotone decreasing in $z(t)$ for all $t \in [0,1]$, and the delay is insensitive to any increase in $z(t)$ when $-\lambda^{-1} \log(1 - z(t)) \ge NT_s t$, it suffices to consider only those z such that $z(t) \ge 1 - e^{-\lambda NT_s t}$. We parameterize this set of z as $z(t) = 1 - e^{-\lambda NT_s t - y(t)}$ with $y(t) \ge 0$, so that

$$\frac{\partial z(t)}{\partial t} \ge 0 \Leftrightarrow \left(\lambda NT_s + \frac{\partial y(t)}{\partial t} \right) e^{-\lambda NT_s t - y(t)} \ge 0 \Leftrightarrow$$

$$\frac{\partial y(t)}{\partial t} \ge -\lambda NT_s.$$

This allows us to express (56) in terms of the functional y

$$\inf_{y \in \mathcal{Y}(\bar{d})} \int_0^1 \frac{1}{1 - e^{-\lambda NT_s t - y(t)}} dt \quad (58)$$

where $\mathcal{Y}(\bar{d})$ is the set of differentiable $y : [0,1] \to [0,\infty)$ obeying

$$\int_0^1 y(t) dt \le \lambda c(M)\bar{d}, \quad \frac{\partial y(t)}{\partial t} \ge -\lambda NT_s, \quad y(t) \ge 0.$$

The Lagrangian $\mathcal{L}(y, \mu, a, b)$ is

$$\int_0^1 \frac{1}{1 - e^{-\lambda NT_s t - y(t)}} dt - \mu \left(\int_0^1 y(t) dt - \lambda c(M)\bar{d} \right)$$

$$+ \int_0^1 a(t) \left(\frac{\partial y(t)}{\partial t} + \lambda NT_s \right) dt + \int_0^1 b(t) y(t) dt \quad (59)$$

where $a(t)$ is the Lagrange multiplier on the constraint $\frac{\partial y(t)}{\partial t} \ge 0$ at t and $b(t)$ is the multiplier on the constraint $y(t) \ge 0$ at t, for each $t \in [0,1]$. The solution to the variational problem (58) must satisfy the algebraic equation, obtained by differentiating (59) with respect to $y(t)$ at each $t = t_0$

$$-\frac{e^{-\lambda NT_s t_0 - y^*(t_0)}}{(1 - e^{-\lambda NT_s t_0 - y^*(t_0)})^2} - \mu + \alpha(t_0) \mathbf{1}_{\{\frac{\partial y^*(t_0)}{\partial t} = -\lambda NT_s\}}$$

$$+ \beta(t_0) \mathbf{1}_{\{y^*(t_0) = 0\}} = 0 \quad (60)$$

for $t_0 \in [0,1]$. Here $\alpha(t_0)$ and $\beta(t_0)$ are defined to be nonzero when the slope of $y(t_0)$ is $-\lambda NT_s$ or $y(t_0) = 0$, respectively. These constraints dictate the values of $\alpha(t_0)$ and $\beta(t_0)$, although it is not necessary to solve for them explicitly. From this equation it follows that y^* will be a piecewise function that is either decreasing at slope $-\lambda NT_s$ or is equal to zero. To see this, note that if both inequality constraints $\frac{\partial y(t)}{\partial t} \ge -\lambda NT_s$ and $y(t) \ge 0$ are inactive for some interval $t \in (t_r, t_l)$, so that the only nonzero Lagrange multiplier is $\mu \le 0$, we find that $z(t)$ must be a constant on this interval, since (60) in this case could be rewritten as

$$\frac{1 - z(t)}{(z(t))^2} = -\mu, \quad t \in (t_r, t_l)$$

but this implies $\frac{\partial y(t)}{\partial t} = -\lambda NT_s$, which is a contradiction because this constraint was assumed inactive. Thus, one of the two constraints $\frac{\partial y(t)}{\partial t} \ge -\lambda NT_s$ and $y(t) \ge 0$ is always active, and this dictates the form of $y(t)$. Thus, the optimal y^* takes the form

$$y^*(t) = \begin{cases} \lambda NT_s(\zeta - t), & 0 \le t \le \zeta \\ 0, & \zeta < t \le 1 \end{cases} \quad (61)$$

for some $\zeta \in [0,1]$. The delay constraint dictates the optimal value of ζ

$$\int_0^1 y^*(t) dt = \lambda c(M)\bar{d} \to \frac{1}{2} \lambda NT_s(\zeta^*)^2$$

$$= \lambda c(M)\bar{d} \Leftrightarrow \zeta^* = \sqrt{\frac{2c(M)\bar{d}}{NT_s}}.$$

Further, the optimal z^* takes the form

$$z^*(t) = \begin{cases} 1 - e^{-\lambda NT_s \zeta^*}, & 0 \le t \le \zeta^* \\ 1 - e^{-\lambda NT_s t}, & \zeta^* < t \le 1. \end{cases} \quad (62)$$

The corresponding inverse normalized rate is

$$\frac{1}{\bar{\rho}_{\bar{d}}^*} = \int_0^1 \frac{1}{z^*(t)} dt$$

$$= \int_0^{\zeta^*} \frac{1}{1 - \exp(-\lambda NT_s \zeta^*)} dt + \int_{\zeta^*}^1 \frac{1}{1 - \exp(-\lambda NT_s t)} dt$$

$$= \frac{\zeta^*}{1 - \exp(-\lambda NT_s \zeta^*)} + \frac{1}{\lambda NT_s} \log \left(\frac{\exp(\lambda NT_s) - 1}{\exp(\lambda NT_s \zeta^*) - 1} \right)$$

$$= \frac{\sqrt{2c(M)\bar{d}}}{\sqrt{NT_s}(1 - \exp(-\lambda\sqrt{2NT_s c(M)\bar{d}}))}$$

$$+ \frac{1}{\lambda NT_s} \log \left(\frac{\exp(\lambda NT_s) - 1}{\exp(\lambda\sqrt{2NT_s c(M)\bar{d}}) - 1} \right). \quad (63)$$

It remains to approximate $c(M)$. To do this, we first find the limiting distribution of a normalization of τ_M to be a Gumbel distribution. In the limit, τ_M is increasingly concentrated around its mean because its variance does not scale with M while its mean does. This allows us to approximate the delay by replacing τ_M by its mean for very large M. In more detail, following [44, pp. 299–300], we find that $\tilde{\tau}_M = \lambda \tau_M - \log(M)$ has a Gumbel distribution with cumulative distribution function $F(t) = \exp(-e^{-t})$. Because the mean of this distribution is

the Euler–Mascheroni constant $\mathbb{E}[\tilde{\tau}_M] \to \gamma \approx 0.5772156649$ as $M \to \infty$, it follows that $\mathbb{E}[\tau_M] \approx \frac{\gamma + \log M}{\lambda}$ for large M. The asymptotic variances are $\mathrm{Var}\ (\tilde{\tau}_M) \to \frac{\pi^2}{6}$ and thus $\mathrm{Var}\ (\tau_M) \to \frac{\pi^2}{6\lambda^2}$. The fact that the mean of τ_M grows unbounded in M while the asymptotic variance of τ_M is independent of M allows us to approximate the expectation

$$\mathbb{E}[(\tau_M - NT_s t)^+] \approx (\mathbb{E}[\tau_M] - NT_s t)^+$$
$$\approx \left(\frac{\gamma + \log M}{\lambda} - NT_s t \right)^+ .$$

Integrating this over t yields

$$c(M) \approx \int_0^1 \left(\frac{\gamma + \log M}{\lambda} - NT_s t \right)^+ \mathrm{d}t \tag{64}$$

$$= \int_0^{\min\{\frac{\gamma + \log M}{\lambda NT_s}, 1\}} \left(\frac{\gamma + \log M}{\lambda} - NT_s t \right) \mathrm{d}t \tag{65}$$

$$= \begin{cases} \frac{(\gamma + \log M)^2}{2\lambda^2 NT_s}, & N > \frac{\gamma + \log M}{\lambda T_s} \\ \frac{\gamma + \log M}{\lambda} - \frac{NT_s}{2}, & N \le \frac{\gamma + \log M}{\lambda T_s}. \end{cases} \tag{66}$$

We may then substitute this into (63) to obtain the rate–delay tradeoff for large M given by

$$\bar{\rho}_{\bar{d}}^* \approx \lambda NT_s \left(\frac{(\gamma + \log M)\sqrt{\bar{d}}}{1 - \mathrm{e}^{-\gamma\sqrt{\bar{d}}} M^{-\sqrt{\bar{d}}}} + \log \frac{\mathrm{e}^{\lambda NT_s} - 1}{\mathrm{e}^{\gamma\sqrt{\bar{d}}} M^{\sqrt{\bar{d}}} - 1} \right)^{-1}$$

when $N > \frac{\gamma + \log M}{\lambda T_s}$ and

$$\bar{\rho}_{\bar{d}}^* \approx \lambda NT_s \left[\frac{\sqrt{(2\gamma + 2\log M - \lambda NT_s)\lambda NT_s \bar{d}}}{1 - \exp(-\sqrt{(2\gamma + 2\log M - \lambda NT_s)\lambda NT_s \bar{d}})} \right.$$
$$\left. + \log \frac{\exp(\lambda NT_s) - 1}{\exp(\sqrt{(2\gamma + 2\log M - \lambda NT_s)\lambda NT_s \bar{d}}) - 1} \right]^{-1}$$

when $N \le \frac{\gamma + \log M}{\lambda T_s}$.

REFERENCES

[1] D. S. Lun, "Efficient Operation of Coded Packet Networks," Ph.D. dissertation, MIT, Cambridge, MA, 2006.

[2] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.

[3] C. Hopps, "Analysis of an equal-cost multi-path algorithm," IETF Network Working Group RFC 2992 Nov. 2000 [Online]. Available: http://tools.ietf.org/html/rfc2992

[4] D. Thaler and C. Hopps, "Multipath issues in unicast and multicast next-hop selection," IETF Network Working Group RFC 2991 Nov. 2000 [Online]. Available: http://tools.ietf.org/html/rfc2991

[5] T. M. Cover, "Comments on broadcast channels," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2524–2530, Oct. 1998.

[6] T. M. Cover, "Broadcast channels," *IEEE Trans. Inf. Theory*, vol. IT-18, no. 1, pp. 2–14, Jan. 1972.

[7] P. P. Bergmans, "Random coding theorem for broadcast channels with degraded components," *IEEE Trans. Inf. Theory*, vol. IT-19, no. 2, pp. 197–207, Mar. 1973.

[8] A. Albanese, J. Blömer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoded transmission," *IEEE Trans. Inf. Theory*, vol. 42, no. 6, pp. 1737–1744, Nov. 1996.

[9] D. Silva and F. R. Kschischang, "Rank-metric codes for priority encoding transmission with network coding," in *Proc. Canadian Workshop on Information Theory*, Edmonton, AB, Canada, Jun. 2007, pp. 81–84.

[10] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[11] R. W. Yeung, S.-Y. R. Li, N. Cai, and Z. Zhang, "Network coding theory," *Found. Trends Commun. Inf. Theory*, vol. 2, no. 4–5, pp. 241–381, 2005.

[12] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, "Network coding: An instant primer," *ACM SIGCOMM Computer Commun. Rev.*, vol. 36, no. 1, pp. 63–68, 2006.

[13] Z. Zhang, "Some recent progresses in network error correction coding theory," in *Proc. 4rth Workshop on Network Coding, Theory, and Applications (NetCod 2008)*, Hong Kong, China, Jan. 2008, pp. 86–90.

[14] D. Silva, F. R. Kschischang, and R. Koetter, "A rank-metric approach to error control in random network coding," in *Proc. 2007 IEEE Information Theory Workshop on Information Theory for Wireless Networks*, Solstrand, Norway, Jul. 2007, pp. 1–5.

[15] R. Koetter and F. R. Kschischang, "Coding for errors and erasures in random network coding," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Nice, France, Jun. 2007, pp. 791–795.

[16] D. Silva, F. R. Kschischang, and R. Koetter, "A rank-metric approach to error control in random network coding," *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 3951–3967, Sep. 2008.

[17] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. 41st Annu. Allerton Conf. Communication, Control, and Computing*, Monticello, IL, Sep. 2003.

[18] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-Cost multicast over coded packet networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2608–2623, Jun. 2006.

[19] L. Keller, E. Drinea, and C. Fragouli, "Online broadcasting with network coding," in *Proc. 4th Workshop on Network Coding, Theory, and Applications (NetCod 2008)*, Hong Kong, China, Jan. 2008, pp. 68–73.

[20] J. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data," in *Proc. ACM SIGCOMM'98*, Vancouver, BC, Canada, Jan. 1998, pp. 556–567.

[21] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, Jun. 2006.

[22] R. G. Gallager, *Low Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.

[23] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 2, pp. 21–28, 1962.

[24] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–639, Feb. 2001.

[25] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 585–598, Feb. 2001.

[26] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 3, pp. 399–431, Mar. 1999.

[27] S. Sanghavi, "Intermediate performance of rateless codes," in *Proc. IEEE Information Theory Workshop (ITW)*, Lake Tahoe, CA, Sep. 2007, pp. 478–482.

[28] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: Model and erasure channel properties," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2657–2673, Nov. 2004.

[29] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE Int. Conf. Communications (ICC 93)*, Geneva, Italy, May 1993, vol. 2, pp. 1064–1070.

[30] C. Berrou and A. Glavieux, "Near optimum error correction coding and decoding: Turbo codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1262–1271, Oct. 1996.

[31] S. Benedetto and G. Montorsi, "Unveiling turbo codes: Some results on parallel concatenated coding structures," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 409–428, Mar. 1996.

[32] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 909–926, May 1998.

[33] S. Boucheron and M. R. Slamatian, "About priority encoding transmission," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 699–705, Mar. 2000.

[34] R. L. Urbanke and A. D. Wyner, "Methods and Apparatus for Packetizing Data for Transmission Through an Erasure Broadcast Channel," U.S. Patent 6 175 944, 1997.

[35] R. L. Urbanke and A. D. Wyner, "Packetizing for the Erasure Broadcast Channel With an Internet Application," 1997 [Online]. Available: citeseer.ist.psu.edu/urbanke97packetizing.html, preprint

[36] C. Huang, R. Janakiraman, and L. Xu, "Loss-resilient on-demand media streaming using priority encoding," in *Proc. ACM Multimedia (MM 2004)*, New York, Oct. 2004, pp. 152–159.

[37] R. Janakiraman and L. Xu, "Layered priority encoded transmission for video streaming to heterogeneous clients," in *Proc. IEEE Int. Symp. Information Theory* , Chicago, IL, Jun/Jul. 2004, p. 331.
[38] R. Janakiraman and L. Xu, "Efficient and flexible parallel retrieval using priority encoded transmission," in *Proc. 14th ACM Int. Workshop on Network and Operating Systems Support for Digital Audio and Video (ACM NOSSDAV 2004)*, Kinsale, Ireland, June 2004, pp. 48–53.
[39] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley , 1968.
[40] R. Berry and R. Gallager, "Communication over fading channels with delay constraints," *IEEE Trans. Inf. Theory*, vol. 48, no. 5, pp. 1135–1149, May 2002.
[41] R. G. Gallager, "Capacity and coding for degraded broadcast channels," *Probl. Pered. Inform.*, vol. 10, no. 3, pp. 3–14, 1974.
[42] F. Y. Wan, *Introduction to the Calculus of Variations and Its Applications*. London, U.K.: Chapman and Hall Mathematics, 1995.
[43] I. Gelfand and S. Fomin, *Calculus of Variations*. New York: Dover, 1963.
[44] H. David and H. Nagaraja, *Order Statistics*. Hoboken, N.J.: Wiley, 2003.
[45] S. Lin and D. J. Costello, Jr., *Error Control Coding*. Upper Saddle River, NJ: Pearson Prentice Hall, 2004.
[46] A. E. Gamal, "The feedback capacity of degraded broadcast channels," *IEEE Trans. Inf. Theory*, vol. IT-24, no. 3, pp. 379–381, May 1978.
[47] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. : Wiley, 2006.
[48] T. S. Han, "Nonnegative entropy measures of multivariate symmetric correlations," *Inf. Contr.*, vol. 36, no. 2, pp. 133–156, 1978.

John MacLaren Walsh (S'01–M'07) was born in Carbondale, IL, in 1981. He received the B.S. (*magna cum laude*), M.S., and Ph.D. degrees from Cornell University, Ithaca, NY in 2002, 2004, and 2006, respectively.

In September 2006, he joined the Department of Electrical and Computer Engineering at Drexel University, Philadelphia, PA, where he is currently an Assistant Professor. His current research interests include: (a) delay mitigating codes and rate–delay tradeoffs in multipath routed and network coded networks, (b) joint source separation and identification, and (c) the performance and convergence of distributed collaborative estimation in wireless sensor networks via expectation propagation.

Prof. Walsh is a member of HKN and TBP.

Steven Weber (S'91–M'97) was born in Milwaukee, WI. He received the B.S. degree in 1996 from Marquette University, Milwaukee, and the M.S. and Ph.D. degrees from The University of Texas at Austin, in 1999 and 2003, respectively.

He joined the Department of Electrical and Computer Engineering at Drexel University , Philadelphia, PA, in 2003, where he is currently an Associate Professor. His research interests are centered around mathematical modeling of computer and communication networks, with much of his recent work focused on wireless *ad hoc* networks.

Ciira wa Maina (S'08) was born in Nairobi, Kenya, in 1983. He received the B.S. degree from the University of Nairobi, in 2007.

He is currently working towards the Ph.D. degree at Drexel University, Philadelphia, PA. His research interests include approximate Bayesian inference techniques applied to joint sound source separation and identification and the performance and convergence of blind adaptive channel shorteners.