

Opportunistic Scheduling of Randomly Coded Multicast Transmissions at Half-Duplex Relay Stations

Chao Chen, Seung Jun Baek, *Member, IEEE*, and Gustavo de Veciana, *Fellow, IEEE*

Abstract—We consider the multicast scheduling problem for the block transmission of packets in a heterogeneous network using a half-duplex relay station (RS). The RS uses random linear coding to efficiently transmit packets over time-varying multicast channels. Our goal is to minimize the average decoding delay. Because of the half-duplex operation, at each time slot, the RS must decide to either: 1) fetch a new packet for encoding from the base station or 2) multicast a coded packet to wireless users. Thus, optimal scheduling hinges on exploiting multicast opportunities while persistently supplying the encoder (at the RS) with new packets. We formulate an associated fluid control problem and show that the optimal policy incorporates opportunism across multicast channels, i.e., the RS performs a multicast transmission only if the collection of channel conditions is favorable; otherwise, it performs a fetch. Based on the fluid policy, we propose an online algorithm. We prove that our algorithm asymptotically incurs no more than $4/3$ and 2 times the optimal delay, for two-user and arbitrary number of user system, respectively. Simulation results show that, in fact, our algorithm's performance is very close to theoretical bounds.

Index Terms—Heterogeneous networks, network coding, opportunistic scheduling, fluid approximation, asymptotic performance.

I. INTRODUCTION

TO MEET the ever-growing demand for throughput and coverage, relay technologies have been widely considered in next-generation cellular systems, e.g., heterogeneous networks with relay stations [1], [2]. We focus on *two-hop* relay networks in which there exists an intermediate node dedicated for relaying. The first hop serves as the high-speed wireless backhaul, and the second hop provides the access links to wireless users. In cellular networks, the two-hop relay

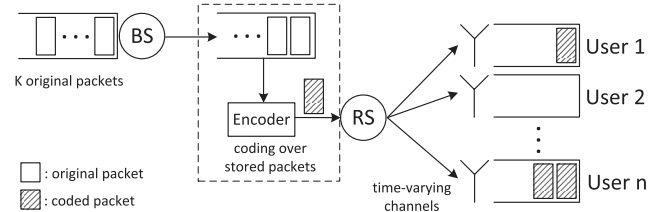


Fig. 1. System model.

architecture is preferable to multi-hop (more than two hops) relay networks, because of relatively low packet delays and reduced routing/signalling overheads. Meanwhile, with advances in cellular technology, recent years have witnessed an explosion of multimedia traffic over cellular networks. Accordingly, multicasting over wireless networks has received much interest as a means for efficient dissemination of multimedia information to wireless users, e.g., evolved multimedia broadcast/multicast services (eMBMS) [3]. While multicasting over relay networks has been studied recently, e.g., from information theoretic perspectives [4]–[6], very little is known about the channel-aware (opportunistic) scheduling in such systems. Opportunism of time-varying multicast channels is not well understood even for simple relay networks, and there exist a number of related open problems.

In this paper, we study multicast scheduling algorithms using Random Linear Network Coding (RLNC) in heterogeneous networks with relay stations. As shown in Fig. 1, the network consists of a macro Base Station (BS), a Relay Station (RS) and users associated with the RS. A data block consisting of K packets, initially backlogged at a BS queue, must be delivered to all users. The RS stores the packets received from the BS in its buffer, and applies RLNC over the stored packets to generate coded packets, which are then transmitted to the users. Our goal is to minimize the average *decoding delay*. The decoding delay is defined as the duration of time beginning from the instant at which the first packet of the information block is relayed to the RS until every user has received K linearly independent packets and thus is able to decode the entire data block.

To exploit the broadcasting nature of wireless links, the RS transmits a coded packet on a single time-frequency resource. For simplicity, this is done at a fixed multicast rate across all packets. We assume that the users' channels are time-varying and that their Channel State Information (CSI) is known to the RS. Consequently, the transmission of a coded packet is only successful for users whose channel state can

Manuscript received July 20, 2015; revised January 15, 2016; accepted February 19, 2016. Date of publication March 3, 2016; date of current version September 13, 2016. C. Chen and S. J. Baek were supported in part by the Institute for Information and Communications Technology Promotion within the Ministry of Science, ICT and Future Planning (MSIP) under Grant B0126-15-1046 and in part by the Basic Science Research Program within MSIP through the National Research Foundation of Korea under Grant NRF-2013R1A1A1062500. G. de Veciana was supported by the National Science Foundation within the Division of Computer and Network Systems under Grant CNS-1343383.

C. Chen is with the School of Information and Electronic Engineering, Zhejiang Gongshang University, Hangzhou 310018, China (e-mail: eckio_491@163.com).

S. J. Baek is with the College of Information and Communications, Korea University, Seoul 136-701, South Korea (e-mail: sjbaek@korea.ac.kr).

G. de Veciana is with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712-0240 USA (e-mail: gustavo@ece.utexas.edu).

Communicated by C. E. Koksal, Associate Editor for Communication Networks.

Digital Object Identifier 10.1109/TIT.2016.2537837

support the multicast rate. We assume that the RS operates in the half-duplex mode, as in many practical systems [2]. Thus, the RS must repeatedly make scheduling decisions as to whether to *multicast* a coded packet or to *fetch* an original packet.¹ Our model is intended to capture the salient characteristics of RS scheduling for such multicast transmissions.

Our focus is on scheduling strategies leveraging opportunism across time-varying multicast channels. We show that it is important to strike a good balance between multicasting and fetching, as described below. On the one hand, the RS should frequently fetch original packets from the BS. Otherwise, there may exist many users who already have received the *same* number of linearly independent packets as the original packets stored at the RS. Because the coded packets are random linear combinations of those stored at the RS, the coded packets transmitted thereafter will *not* be linearly independent of the packets already received by these users, rendering the transmitted packets useless to them. On the other hand, fetching too often can be inefficient, because fetching is always performed at the expense of the opportunity to multicast, due to the half-duplex operation of the RS. A key insight is that it is desirable to multicast only when the channel conditions are favorable (i.e., many users are likely to receive the packet and the packet contains new information on degrees of freedom relative to those already received) and to fetch original packets otherwise. One can achieve a good balance between multicasting and fetching by making the best use of multicasting opportunities while keeping the coded packets “innovative” to many users. Optimal scheduling, however, is a complex function of, for example, the queue lengths of the RS and users, and channel parameters (possibly asymmetric among users). This problem is particularly challenging for large systems. We aim to develop a scalable algorithm with provable performance guarantees. Despite their importance, such scheduling/queueing strategies for the opportunistic coded multicast have not yet been well explored for two-hop relay networks. The related work is described below.

A. Related Work

One of the fundamental studies of random linear coding in packet networks is [8]. The authors show that RLNC achieves the min-cut capacity for both unicast and multicast in lossy packet networks. They assume, unlike in our work, that all packets have predetermined schedules for transmission. Studies [9] and [10] examine the throughput and delay performance of RLNC for broadcasting over one-hop networks as a function of the coding window size and the number of users. In [11], the authors focus on minimum delay scheduling in single-hop networks and show that an RLNC scheme, even without CSI, can outperform traditional uncoded schemes with CSI. Note that the analyses in [9]–[11] are mainly restricted to the case in which the users are statistically symmetric in terms of channel conditions. However, in this paper, we will address the challenging case of asymmetric channels.

¹In order to coordinate with the BS on scheduling, the RS can generate cell control messages, e.g., the RS operates as a “non-transparent relay” specified by the IEEE 802.16 working group [7].

The queueing aspects of wireless coded multicast over single-hop networks have been studied in [12] and [13]. The study in [12] analyzes the queueing delay performance of a packet coding scheme with an adaptive coding window size based on the number of packets buffered at the source. In [13], the authors consider a model in which a BS serves multiple multicast flows. They propose a simple coding scheme that combines packets from different queues, and derive conditions under which their scheme achieves the maximum throughput.

Prior work on RLNC for multi-hop networks includes [14], where the authors study packet coding schemes over lossy networks in which the intermediate nodes can encode packets stored in their finite-size buffers; however, they focus on serving a single flow without scheduling conflicts, which differs from our work. The study in [15] examines a packet scheduling problem using RLNC in a cooperative network comprising a source and two receivers. The authors take a dynamic programming approach to find the optimal schedule, which has scalability issues when the number of receivers is large. Recently, a multicast scheduling algorithm for two-hop OFDMA relay networks was proposed in [16]. However, they do not consider network coding, and their objective differs from ours; they consider finding the optimal subchannel allocation over multiple multicast sessions to maximize the aggregate multicast throughput.

B. Contributions and Paper Organization

In this paper, we propose a scalable scheduling algorithm for opportunistic coded multicast in two-hop relay networks. Below, we summarize our contributions.

- (a) The underlying decision problem is a Markov Decision Process (MDP) that is intractable, even for a moderate problem size. Thus, we formulate an associated fluid control problem. The fluid problem is nonconvex because of the nonlinear characteristics of decoding delays under RLNC; however, we are able to show the optimal fluid control problem can be reformulated as a linear program (LP).
- (b) The LP is still difficult to solve because its size is exponential in the number of users. However, we show that the solution has a threshold-based structure. Indeed the optimal policy is for the RS to multicast only if the current channel conditions are such that the “revenue” (defined later) exceeds a certain threshold; otherwise, the RS fetches a packet from the BS. This result explicitly characterizes the optimal tradeoff point between fetching new packets and opportunistic multicast transmission.
- (c) We propose a low-complexity scheduling algorithm motivated by the thresholding property of the optimal policy. We prove that our algorithm asymptotically achieves at most twice the optimal decoding delay. For two-user systems, the approximation ratio is shown to be at most 4/3. Simulation results show that our algorithm in fact performs far better than the derived bounds, achieving decoding delays that are very close to theoretical lower bounds.

The remainder of this paper is organized as follows. Section II introduces our system model and

problem formulation. The associated fluid problem is developed in Section III; its reformulation as an LP is discussed in Section IV. In Section V, we identify the properties of the optimal fluid policy. We propose our algorithm in Section VI and analyze its performance in Section VII. Section VIII contains simulation results. We discuss some extensions of our problem in Section IX. Section X concludes the paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We shall use boldface letters to denote vectors and matrices. The i th entry of vector \mathbf{x} is denoted by x_i . All of the vectors are assumed to be column vectors, unless stated otherwise. We define $[x]^+ := \max\{x, 0\}$.

Let n denote the number of users, and $I = \{1, \dots, n\}$ denote the set of user indices. A block of K packets is initially backlogged at the BS, and is to be disseminated to all users through the RS. Each packet is represented as a vector of length v over a finite field \mathbb{F}_d and has a length of $v \lceil \log_2 d \rceil$ bits. We assume that all original packets received by the RS are stored in the RS queue until the transmission completes. The RS creates coded packets by applying RLNC over all the original packets in its queue. We consider a time-slotted system in which the integer l denotes the time index. At each time slot, the half-duplex RS can either fetch an original packet from the BS or multicast a coded packet to the users. The decoding delay is defined as the number of time slots it takes for *all* of the users to receive K linearly independent coded packets. The decoding overhead of the users is assumed to be negligible.²

Multicast packets are encoded and transmitted on a single time-frequency resource unit, e.g., a Resource Block (RB), destined for all the users, at a predetermined fixed rate used for multicasting. The signal to noise ratio (SNR) of the RB for user i is represented by discrete-time random process $C_i(l)$, $l \in \mathbb{Z}$. We assume $C_i(l)$'s are independent across the users. We define the SNR threshold $\gamma_0 > 0$ such that, γ_0 is the minimum SNR required for a user to receive a fixed-size packet encoded at the RB. Hence for user i , two cases may occur at time slot l ; either the instantaneous SNR of user i is high enough for the packet, i.e., $C_i(l) \geq \gamma_0$, in which case user i is capable of receiving the packet, or otherwise. Let us define a binary random process

$$X_i(l) := \mathbf{1}(C_i(l) \geq \gamma_0), \quad l \in \mathbb{Z}$$

where $\mathbf{1}(\cdot)$ denotes the indicator function. Thus, we have $X_i(l) = 1$ if user i would be able to receive the packet at time slot l , and $X_i(l) = 0$ otherwise. We assume that the CSI is available at the transmitter, i.e., the RS knows $X_i(l)$, $\forall i \in I$, prior to transmission. The RS allocates³ the RB only to the

users who can receive the packet, i.e., the RS multicasts the packets to the set of users given by $\{i \in I | X_i(l) = 1\}$ at time slot l . Because of such resource sharing in multicast transmissions, the users' channels are inherently "ON-OFF" channels. We say that the channel of user i is in the ON (resp. OFF) state at time slot l if $X_i(l) = 1$ (resp. $X_i(l) = 0$). Note for simplicity, we assume that there are no "packet erasures", i.e., the users in the ON state can successfully receive the transmitted packet with probability 1.

We will consider the Rayleigh fading channel as our time-varying channel model. We assume that $C_i(l)$ is the SNR process of a Rayleigh fading channel induced by user mobility under a multipath environment with independent isotropic scatterers. This is the well-known Clarke's fading model [18]. Under the assumption of additive Gaussian noise, the received SNR $C_i(l)$ has the marginal probability density function (pdf) given by

$$f_i(a) = \frac{1}{\phi_i} \exp\left(-\frac{a}{\phi_i}\right), \quad a \geq 0 \quad (1)$$

where ϕ_i denotes the average SNR for user i . The fluctuation of the fading process $C_i(l)$ is quantified by the level crossing rate (LCR) [19] as follows. The LCR is defined as the average number of times the SNR falls below a certain threshold per second. The LCR of user i 's channel with respect to the SNR threshold γ_0 , denoted by N_i , is given by [19]

$$N_i = \sqrt{\frac{2\pi\gamma_0}{\phi_i}} f_m \exp\left(-\frac{\gamma_0}{\phi_i}\right) \quad (2)$$

where f_m denotes the maximum Doppler frequency of the users.

We will approximate the process $X_i(l)$ as a two-state Markov chain as follows. We will adopt the framework of [20] and [21] on modelling Rayleigh fading processes as finite-state Markov chains. Based on the marginal pdf (1) and the LCR (2) of the Rayleigh fading process [20], we can model $X_i(l)$ as a stationary and ergodic discrete-time Markov chain (DTMC) on $\{0, 1\}$ where state 0 (resp. 1) corresponds to the state such that the SNR is below (resp. above) γ_0 . Throughout the paper, let

$$p_i := \mathbb{P}(X_i(l) = 1) \quad (3)$$

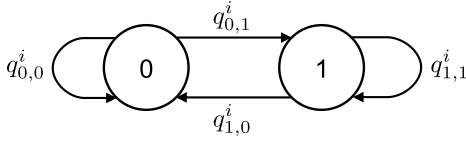
denote the steady-state probability that the channel of user i is in the ON state. The stationary distribution of $X_i(l)$ is given by

$$\begin{aligned} \mathbb{P}(X_i(l) = 0) &= \mathbb{P}(C_i(l) \leq \gamma_0) = 1 - \exp\left(-\frac{\gamma_0}{\phi_i}\right), \\ \mathbb{P}(X_i(l) = 1) &= p_i = \exp\left(-\frac{\gamma_0}{\phi_i}\right). \end{aligned} \quad (4)$$

The transition probability of DTMC $X_i(l)$ from state j to k for $j, k = 0, 1$, denoted by $q_{j,k}^i$, can be approximated as follows [20]. The probability $q_{0,1}^i$ is proportional to the ratio of the LCR from state 0 to 1 and the average number of slots per second the channel is at state 0; combining this

²As discussed in [17], the receiver is able to retrieve the coding coefficients for decoding, if it runs an identical pseudo-random number generator (PRNG) initialized with the same seed as the encoder. We can show that, it is sufficient for a coded packet to contain the information on (1) the number of original packets used for the encoding at the RS, and (2) the total number of coded packets transmitted. The users can feed this information to the PRNG to retrieve the coefficients.

³By allocation we mean that the RS notifies a subset of users through a control channel so that those users may "listen" to the RB for reception. The users who did not get the allocation can simply stay idle, or temporarily enter "sleep" mode to save power.

Fig. 2. Markov chain $X_i(l)$.

with (2), $q_{0,1}^i$ is given by

$$q_{0,1}^i \approx \frac{N_i}{f_s \cdot \mathbb{P}(X_i(l) = 0)} = \sqrt{\frac{2\pi\gamma_0}{\phi_i}} \cdot \frac{\exp(-\frac{\gamma_0}{\phi_i})}{1 - \exp(-\frac{\gamma_0}{\phi_i})} \cdot \frac{f_m}{f_s} \quad (5)$$

where f_s denotes the symbol (packet) rate. Similarly, we have that

$$q_{1,0}^i \approx \frac{N_i}{f_s \cdot \mathbb{P}(X_i(l) = 1)} = \sqrt{\frac{2\pi\gamma_0}{\phi_i}} \cdot \frac{f_m}{f_s}. \quad (6)$$

The Markov chain $X_i(l)$ is shown in Fig. 2. Later in the simulation section, we will generate $X_i(l)$'s for the users undergoing different degrees of fading by varying the system parameters in expressions (4)–(6).

The channel between the BS and the RS is called the wireless *backhaul*. We assume that the backhaul is always in the ON state. This is a reasonable assumption because, in practice, the backhaul tends to have a high capacity, e.g., the RS may be located in the line-of-sight of the BS.

We will use vector $\mathbf{s} := (s_1, \dots, s_n) \in \{0, 1\}^n$ to denote the aggregate channel state of users. Specifically, $s_i = 1$ (resp. $s_i = 0$) indicates that the channel state of user i is in the ON (resp. OFF) state. For example, for $n = 3$, the channel state $\mathbf{s} = (1, 1, 0)$ indicates that the channels of user 1 and 2 are in the ON state. We let $\mathcal{S} := \{0, 1\}^n$ denote the set of all possible channel states. Let the random process $\mathbf{X}(l) := (X_1(l) \cdots X_n(l)) \in \mathcal{S}$ denote the channel state in time slot l . The probability that the channel state is \mathbf{s} in the steady state is denoted by $p(\mathbf{s})$ where

$$p(\mathbf{s}) := \mathbb{P}(\mathbf{X}(l) = \mathbf{s}) = \prod_{i \in I} p_i^{s_i} (1 - p_i)^{1-s_i}.$$

When a coded packet is received by user i , we say that it is an *innovative* packet for user i if it is linearly independent of all of the packets previously received by user i . Let $B_i(l)$ (resp. $\hat{B}_i(l)$) denote the number of packets (resp. innovative packets) received by user i up to time slot l . Note that $B_i(l)$ counts both innovative and noninnovative packets. In fact, only the innovative packets matter to the users. Let $B_R(l)$ be the number of packets received by the RS up to time slot l . We have that for all $i \in I$

$$B_i(l) \geq \hat{B}_i(l), \text{ and } B_R(l) \geq \hat{B}_i(l).$$

When a coded packet is multicast at time slot l , the probability that it is innovative to user i is given by [13]

$$1 - d^{\hat{B}_i(l) - B_R(l)}. \quad (7)$$

Note that (7) is equal to 0 when $B_R(l) = \hat{B}_i(l)$, and is close to 1 when $B_R(l) > \hat{B}_i(l)$ and d is sufficiently large. For simplicity, we assume that (7) is equal to 1 if

TABLE I
SUMMARY OF THE NOTATION

General System Parameters	
n	number of users
I	$:= \{1, \dots, n\}$, set of user indices
K	number of packets in a block
d	coding field size
$X_i(l)$	channel state of user i in time slot l
p_i	probability that user i 's channel is in the ON state
i^b	$\in \arg \min_{i \in I} \{p_i\}$, index of the bottleneck user
\mathbf{s}	$:= (s_1, \dots, s_n)$, channel state
\mathcal{S}	$:= \{0, 1\}^n$, set of channel states
$p(\mathbf{s})$	probability that the channel state is \mathbf{s}
$B_i(l)$	number of packets received by user i up to time slot l
$\hat{B}_i(l)$	number of innovative packets received by user i up to time slot l
$B_R(l)$	number of packets in the RS queue in time slot l
\mathbf{w}	$:= (w_1, \dots, w_n)$, vector of users' weights
$\rho(\mathbf{w}, \mathbf{s})$	$:= \sum_{i \in I} s_i w_i$, revenue associated with channel state \mathbf{s}
$\hat{\rho}(\mathbf{w}, \mathbf{s})$	$:= \rho(\mathbf{w}, \mathbf{s}) / (1 + s_{i^b})$, normalized revenue associated with channel state \mathbf{s}
Fluid Model Related Parameters	
$\pi(\mathbf{s}, t)$	instantaneous fraction of time spent for multicasting under the condition that the channel state is \mathbf{s} at time t
$b_i(t)$	units of fluid received by user i up to time t
$\hat{b}_i(t)$	units of innovative fluid received by user i up to time t
$b_R(t)$	units of fluid in the RS queue at time t
$r_i(t)$	fluid rate achieved by user i at time t
$\hat{r}_i(t)$	innovative fluid rate achieved by user i at time t
$r_{in}(t)$	inflow rate to the RS at time t

$B_R(l) > \hat{B}_i(l)$. In other words, if user i has received fewer innovative packets than the original packets received by the RS, any coded packet multicast by the RS is innovative to user i . This assumption is widely used, e.g., see [12] and [15]. Once a user receives K innovative packets, the user can decode the whole block of original packets. The goal for the RS is to take optimal scheduling actions, either a multicast or a fetch, each time slot so as to minimize the mean decoding delay. Table I is a summary of the notation used in our work.

Observe that there exists a specific structure to our problem. Since the RS stores all the original packets fetched from the BS, $B_R(\cdot)$ is nondecreasing over time. Thus, under any policy, $B_R(\cdot)$ will eventually reach K . When $B_R(\cdot)$ is equal to K , i.e., the whole information block is received at the RS, the RS does not need to fetch additional packets from the BS; thereafter an optimal policy for the RS will keep generating coded packets from the stored block, and keep multicasting the coded packets, in order to minimize the decoding delay. Based on this observation, we will divide a transmission policy into two stages, depending on whether the whole information block has been received at the RS, as follows.

- **Opportunistic Fetch stage:** the time duration from the beginning of the transmission until the RS fetches the K original packets. At each time slot in this stage, the RS should decide to perform either a fetch or a multicast depending on the state of the time-varying multicast channels.
- **Flush stage:** the time duration following the Opportunistic Fetch stage until all users have received K innovative packets. The RS always multicasts at each time slot in this stage.

Thus, without loss of generality, we only consider two-stage policies such that, in the Flush stage, the RS keeps multicasting (newly coded) packets until all users can decode the block.

We briefly show that our problem can be formulated as an MDP. We define $\mathcal{Q} := \{0, 1, \dots, K\} \times \{0, 1, \dots, K\}^n$ as the system state space, which represents the number of original packets received by the RS and the number of linearly independent packets received by the n users. The action space \mathcal{A} of our MDP consists of two elements: the RS may perform either a multicast or a fetch. If we associate a fixed action with each system state, the system clearly evolves as a Markov chain. Note that the system state space $|\mathcal{Q}| = O(K^{n+1})$ and the channel states $|\mathcal{S}| = O(2^n)$. Because of the “curse of dimensionality,” our MDP quickly becomes computationally intractable as K and n increase.

III. FLUID APPROXIMATION

Stochastic control problems are both analytically and computationally difficult, but in this section we will consider the *fluid model* as a deterministic relaxation. Fluid models provide tractable approximations, as shown in many studies [22]–[24].

We first introduce the related random processes. Let $G_s(l)$ (resp. $H_s(l)$) denote the cumulative number of time slots in which the RS performed a multicast (resp. a fetch) and the channel was in state $s \in \mathcal{S}$ up to time slot l . We have that

$$B_R(l) = \sum_{s \in \mathcal{S}} H_s(l); \quad B_i(l) = \sum_{s \in \mathcal{S}} s_i G_s(l), \quad \forall i \in I.$$

Recall that $s_i = 1$ if in channel state s user i 's channel was ON. Define a binary random process $\Psi(l)$ such that $\Psi(l) = 1$ if the RS performs a multicast in time slot l , and 0 otherwise. Thus, we have that

$$G_s(l) = \sum_{j=1}^l \mathbf{1}(X(j) = s, \Psi(j) = 1),$$

$$H_s(l) = \sum_{j=1}^l \mathbf{1}(X(j) = s, \Psi(j) = 0).$$

Let $t \in \mathbb{R}$ denote a continuous time index. We extend the definition of a discrete-time random process $Y(\cdot)$ to continuous time as follows:

$$\tilde{Y}(t) := Y(\lfloor t \rfloor).$$

Let $\mathbb{D}[0, \infty)$ be the set of functions $f : [0, \infty) \rightarrow \mathbb{R}$ that are right-continuous with left limits (RCLL). Then, clearly, $\tilde{G}_s(t)$ and other extended processes defined previously, are random elements in $\mathbb{D}[0, \infty)$. Consider a sequence of systems for the fluid scaling parameter $m = 1, 2, \dots$. In the m th system, the total number of packets to be received is scaled from K to mK . We will use superscript (m) to denote the random processes associated with the m th system, e.g., $\tilde{G}_s^{(m)}(t)$ is the process $\tilde{G}_s(t)$ in the m th system. We denote the rescaling of the random process $\tilde{Y}^{(m)}(t)$ by $y^{(m)}(t)$ given by

$$y^{(m)}(t) := \frac{1}{m} \tilde{Y}^{(m)}(mt). \quad (8)$$

For example, $g_s^{(m)}(t)$ is the rescaling of the process $\tilde{G}_s^{(m)}(t)$.

Lemma 1: The rescaled processes satisfy the following convergence uniformly over compact sets (u.o.c.) as $m \rightarrow \infty$:

$$g_s^{(m)}(t) \rightarrow g_s(t),$$

$$h_s^{(m)}(t) \rightarrow h_s(t),$$

$$b_R^{(m)}(t) \rightarrow b_R(t) = \sum_{s \in \mathcal{S}} h_s(t),$$

$$b_i^{(m)}(t) \rightarrow b_i(t) = \sum_{s \in \mathcal{S}} g_s(t) s_i, \quad \forall i \in I$$

where $g_s(t)$, $h_s(t)$, $b_R(t)$, and $b_i(t)$ are Lipschitz continuous on $[0, \infty)$.

Proof: The proof is similar to [22, Lemma 1]. ■

Hence, $g_s(t)$, $h_s(t)$, $b_R(t)$, and $b_i(t)$ are Lipschitz continuous and have derivatives almost everywhere for $t \geq 0$. The points at which the derivatives exist are called *regular points* [22]. The derivative of $y(t)$ at a regular point t is denoted by $\dot{y}(t)$.

Let $\delta := m^{-\alpha}$ for some $\alpha \in (0, 1)$. For some regular point t , we have that

$$G_s(m(t + \delta)) - G_s(mt) = \sum_{j=\lfloor mt \rfloor + 1}^{\lfloor m(t+\delta) \rfloor} \mathbf{1}(X(j) = s, \Psi(j) = 1).$$

Dividing both sides by $m\delta$, we have that

$$\frac{G_s(m(t + \delta)) - G_s(mt)}{m\delta} = \frac{\sum_{j=\lfloor mt \rfloor + 1}^{\lfloor m(t+\delta) \rfloor} \mathbf{1}(X(j) = s)}{m\delta} \cdot \frac{\sum_{j=\lfloor mt \rfloor + 1}^{\lfloor m(t+\delta) \rfloor} \mathbf{1}(X(j) = s, \Psi(j) = 1)}{\sum_{j=\lfloor mt \rfloor + 1}^{\lfloor m(t+\delta) \rfloor} \mathbf{1}(X(j) = s)}. \quad (9)$$

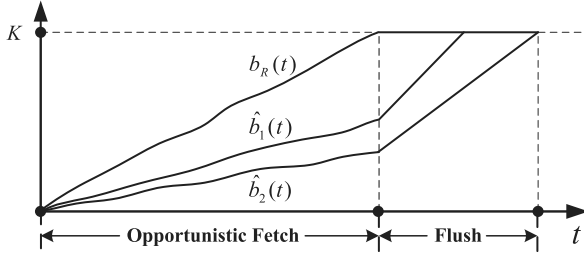
The limit of the factor on the RHS of (10) as $m \rightarrow \infty$ is denoted by $\pi(s, t) \in [0, 1]$, which can be interpreted as the instantaneous fraction of time spent for multicasting, on the condition that the channel state is s at time t . Additionally, as $m \rightarrow \infty$, the factor on the LHS of (10) converges to $p(s)$, and (9) converges to $\dot{g}_s(t)$ because t is regular. Thus, we have that $\dot{g}_s(t) = p(s)\pi(s, t)$. By taking similar steps for the process $H_s(t)$, we get $\dot{h}_s(t) = p(s)[1 - \pi(s, t)]$.

Let $r_i(t) := \dot{b}_i(t)$ and $r_{in}(t) := \dot{b}_R(t)$ denote the fluid rate achieved by user i and the inflow rate to the RS at time t , respectively. Define $\hat{b}_i(t)$ as the cumulative units of *innovative fluid* (analogous to innovative packets in the stochastic model) received by user i up to time t . Let $\hat{r}_i(t) := \dot{\hat{b}}_i(t)$ denote the *innovative fluid rate* achieved by user i at time t . We have that

$$\hat{r}_i(t) = \begin{cases} r_i(t), & \text{if } b_R(t) > \hat{b}_i(t), \\ \min[r_i(t), r_{in}(t)], & \text{if } b_R(t) = \hat{b}_i(t). \end{cases} \quad (11)$$

That is, the innovative fluid rate achieved by user i is equal to $r_i(t)$ if $b_R(t) > \hat{b}_i(t)$; this reduces to $\min[r_i(t), r_{in}(t)]$ if $b_R(t) = \hat{b}_i(t)$, because the total units of innovative fluid received by user i cannot exceed the units of fluid received by the RS.

The users need to receive mK innovative packets in the m th system; thus, under the fluid scaling (8), we want all of the

Fig. 3. An example of buffer trajectories with $n = 2$.

scaled trajectories $\hat{b}_i^{(m)}(t)$, $i \in I$ to reach K . We define the asymptotic decoding delay in the fluid regime as

$$T = \inf_t \left\{ t \in \mathbb{R}_+ \mid \min_{i \in I} \{\hat{b}_i(t)\} = K \right\}. \quad (12)$$

i.e., our goal is to minimize the time until every user receives at least K units of innovative fluid. Hence, delay minimization using a deterministic fluid model can be formulated as follows:

$$(\mathbf{F}) \quad \begin{aligned} & \text{minimize } T = \inf_t \left\{ t \in \mathbb{R}_+ \mid \min_{i \in I} \{\hat{b}_i(t)\} = K \right\} \\ & \text{subject to } b_i(0) = \hat{b}_i(0) = b_R(0) = 0, \quad \forall i \in I, \end{aligned} \quad (13)$$

$$b_i(t) = b_i(0) + \int_0^t r_i(\tau) d\tau, \quad \forall i \in I, \quad (14)$$

$$\hat{b}_i(t) = \hat{b}_i(0) + \int_0^t \hat{r}_i(\tau) d\tau, \quad \forall i \in I, \quad (15)$$

$$b_R(t) = b_R(0) + \int_0^t r_{\text{in}}(\tau) d\tau, \quad b_R(0) = 0, \quad (16)$$

$$r_i(t) = \sum_{s \in S} p(s) \pi(s, t) s_i, \quad \forall i \in I, \quad (17)$$

$$r_{\text{in}}(t) = \sum_{s \in S} p(s) [1 - \pi(s, t)], \quad (18)$$

$$\hat{r}_i(t) = \begin{cases} r_i(t), & \text{if } b_R(t) > \hat{b}_i(t), \\ \min[r_i(t), r_{\text{in}}(t)], & \text{if } b_R(t) = \hat{b}_i(t), \end{cases} \quad (19)$$

$$\begin{aligned} \pi(s, t) &\in [0, 1], \quad \forall s \in S, \\ \pi(s, t) &= 1, \quad \text{if } b_R(t) = K. \end{aligned} \quad (20)$$

Similar to the original stochastic problem, **(F)** is a two-stage fluid fill-up problem, i.e., the Opportunistic Fetch stage (resp. Flush stage) is the time before (resp. after) the instant at which the fluid level at the RS queue reaches K . Fig. 3 shows an example of buffer trajectories for **(F)** with $n = 2$. During the Flush stage, the optimal control will perform multicasting at all times. Hence, the inflow rate to the RS, $r_{\text{in}}(t)$, is exactly 0 during the Flush stage. Also, the fluid rate achieved by user i is p_i for all $i \in I$ during the Flush stage. Thus, it remains to find the optimal policy to be used during the Opportunistic Fetch stage.

IV. PROBLEM REFORMULATION

Solving **(F)** requires a technique called *calculus of variations*, which tends to be difficult. Instead, we convert **(F)** into a simpler optimization problem from which we identify several properties of the solution.

Theorem 1: There exists an optimal policy for the fluid problem **(F)** such that the fluid rates are constant during the Opportunistic Fetch stage.

Proof: The proof is given in Appendix A. ■

Recall that the optimal policy during the Flush stage is to keep multicasting. Thus, the problem reduces to optimizing the constant fluid rates during the Opportunistic Fetch stage. For notational simplicity, we will drop the time index from $\pi(s, t)$, $r_{\text{in}}(t)$, $r_i(t)$, and $\hat{r}_i(t)$, and instead, use $\pi(s)$, r_{in} , r_i , and \hat{r}_i . Based on (11), we have that

$$\hat{r}_i = \min\{r_i, r_{\text{in}}\}, \quad \forall i \in I. \quad (21)$$

Up to time $\frac{K}{r_{\text{in}}}$, the RS has received K units of fluid, and user i has received $\frac{K \hat{r}_i}{r_{\text{in}}}$ units of innovative fluid for each user $i \in I$. Thus, the remaining units of innovative fluid to be received by user i during the Flush stage is given by

$$K - \frac{\hat{r}_i}{r_{\text{in}}} K = \frac{[r_{\text{in}} - r_i]^+}{r_{\text{in}}} K. \quad (22)$$

It follows that, the decoding delay (12) is given by

$$\frac{K}{r_{\text{in}}} + \max_{i \in I} \left\{ \frac{[r_{\text{in}} - r_i]^+}{r_{\text{in}}} \frac{K}{p_i} \right\}.$$

Overall, we have that **(F)** is equivalent to the following problem:

$$\text{minimize } \frac{K}{r_{\text{in}}} + \max_{i \in I} \left\{ \frac{[r_{\text{in}} - r_i]^+}{r_{\text{in}}} \frac{K}{p_i} \right\} \quad (23)$$

$$\text{subject to } r_{\text{in}} = \sum_{s \in S} p(s) [1 - \pi(s)], \quad (24)$$

$$r_i = \sum_{s \in S} p(s) \pi(s) s_i, \quad \forall i \in I, \quad (25)$$

$$\pi(s) \in [0, 1], \quad \forall s \in S. \quad (26)$$

This problem is still difficult for two reasons. First, it is nonconvex, and second, the number of variables $\pi(\cdot)$ is exponential in n . Let us first address the nonconvexity.

Define $i^b \in \arg \min_{i \in I} \{p_i\}$ as the index of the “bottleneck” user, i.e., the user whose channel has the lowest probability of being in the ON state. Consider the following problem:

$$(\mathbf{P}) \quad \text{maximize } \min_{i \in I} \{r_i\} \quad (27)$$

$$\begin{aligned} & \text{subject to } r_{\text{in}} = r_{i^b}, \\ & (24) - (26). \end{aligned} \quad (28)$$

Theorem 2: The solution to **(P)** is optimal for **(F)** as well.

Proof: The proof is given in Appendix B. ■

Let us discuss some of the implications of Theorem 2. In the proof of the theorem, we show that there exists an optimal policy satisfying the following constraint:

$$r_{\text{in}} = \min_{i \in I} \{r_i\}. \quad (29)$$

If we combine (29) and the definition of \hat{r}_i in (21), we have that

$$\hat{r}_i = \min[r_i, r_{\text{in}}] = r_{\text{in}}, \quad \forall i \in I. \quad (30)$$

That is, all users achieve the same innovative fluid rate given by r_{in} under (29). Consequently, all users will simultaneously

finish receiving K units of innovative fluid at the end of the Opportunistic Fetch stage, in which case, the Flush stage will have *zero* duration. Therefore, we need only optimize over one stage, i.e., the Opportunistic Fetch stage. Intuitively, a good policy will dispense with the Flush stage if possible, because during that period it cannot exploit opportunism. Importantly, the nonconvexity of (23) is resolved, because the second term of (23) becomes 0 under constraint (29), which results in the one-stage LP given by (P). In the proof of Theorem 2, we also observe that $r_{ib} = \min_{i \in I} \{r_i\}$ holds under the optimal policy in addition to (29). That is, there exists an optimal policy under which the bottleneck user achieves the minimum fluid rate among users. The bottleneck user governs the decoding delay; thus, one should maximize the service rate of the bottleneck user, as in (P). Theorem 1 allows us to consider only (P), which is significantly simplified from (F), in the rest of this paper.

V. OPTIMAL FLUID POLICY

There are 2^n variables in (P); thus, it seems difficult to solve (P) directly. Instead, we will first examine the structure of the solutions to (P). For this purpose, we will reformulate (P) as a max-weight problem as follows:

$$\begin{aligned}
 \text{(MW)} \quad & \max_{\pi} \sum_{i \in I} w_i r_i \\
 & \text{subject to } \sum_{s \in S} p(s) \pi(s) [1 + s_{ib}] = 1, \\
 & \quad (24) - (26),
 \end{aligned} \tag{31}$$

where (31) is derived from (28) by expressing r_{in} and r_{ib} in terms of $\pi(\cdot)$.

Theorem 3: There exist non-negative weights w_i , $i \in I$, such that (P) is equivalent to (MW).

Proof: The proof is given in Appendix E. ■

One can interpret the weight w_i as the revenue per unit bandwidth accrued by user i . Therefore, we may regard (MW) as a revenue maximization problem as follows. If we express the objective of (MW) in terms of $\pi(\cdot)$, we have that

$$\sum_{s \in S} p(s) \pi(s) \rho(\mathbf{w}, s) \tag{32}$$

Define $\mathbf{w} := (w_1, \dots, w_n)$ as the weight vector, and $\rho(\mathbf{w}, s) := \sum_{i \in I} w_i s_i$ as the revenue earned by the system when the RS multicasts under channel state s . Therefore, (32) is the expected revenue earned by the system under policy $\pi(\cdot)$. We will also define $\hat{\rho}(\mathbf{w}, s) := \rho(\mathbf{w}, s) / (1 + s_{ib})$ as the “normalized” revenue. By definition, $\hat{\rho}(\mathbf{w}, s)$ penalizes $\rho(\mathbf{w}, s)$ by a factor of 2 if the bottleneck user’s channel is ON in channel state s , i.e., $s_{ib} = 1$. The penalty arises because there exists a constraint which involves the bottleneck user, i.e., (31) in the problem (MW). Hence it is natural that the solution of (MW) is biased by the bottleneck user, and that bias/penalty would be captured by the normalized revenue.

Next, we investigate the solution to (MW). Let us introduce the following notation, which will be useful for further discussion. For given \mathbf{w} , let $\boldsymbol{\eta}(k) := (\eta_1(k), \dots, \eta_n(k)) \in S$ denote the channel state such that $\hat{\rho}(\mathbf{w}, \boldsymbol{\eta}(k))$ is the k th largest (ties are

arbitrarily broken) among all $\{\hat{\rho}(\mathbf{w}, s) | s \in S\}$. Additionally, define k^* as

$$k^* := \min \left\{ k \left| \sum_{j=1}^k p(\boldsymbol{\eta}(j)) [1 + \eta_{ib}(j)] > 1 \right. \right\}.$$

Theorem 4: The following policy $\pi^*(\cdot)$ is optimal for (MW):

$$\pi^*(s) = \begin{cases} 1, & \hat{\rho}(\mathbf{w}, s) > \zeta, \\ \beta, & \hat{\rho}(\mathbf{w}, s) = \zeta, \\ 0, & \text{otherwise,} \end{cases} \tag{33}$$

where

$$\zeta := \hat{\rho}(\mathbf{w}, \boldsymbol{\eta}(k^*)), \quad \beta := \frac{1 - \sum_{s: \hat{\rho}(\mathbf{w}, s) > \zeta} p(s) [1 + s_{ib}]}{\sum_{s: \hat{\rho}(\mathbf{w}, s) = \zeta} p(s) [1 + s_{ib}]}.$$

Proof: The proof is given in Appendix F. ■

Remark: By definition, the fluid policy $\pi^*(s)$ represents the time fraction of multicasting under channel state s . In the original stochastic network, $\pi^*(s)$ (resp. $1 - \pi^*(s)$) is analogous to the *probability* of multicasting (resp. fetching) at a time slot, under the condition that the channel state is s . From that perspective, we see that $\pi^*(\cdot)$ clearly possesses a threshold-based structure; the RS performs a multicast w.p. 1 at a time slot only if the normalized revenue earned by the system exceeds the threshold ζ . Indeed, from $\pi^*(\cdot)$, we observe the aforementioned tradeoff between multicasting and fetching. The RS should perform multicasting only if the current channel condition is sufficiently favorable, i.e., it yields a high normalized revenue; otherwise, the RS should perform fetching.

We also find that the normalized revenue serves as a measure of channel quality/opportunism. The optimal policy implicitly *ranks* the channel states in the order of associated normalized revenue. That is, for two channel states $s, \sigma \in S$, we may say that s *precedes* σ in terms of channel quality if $\hat{\rho}(\mathbf{w}, s) > \hat{\rho}(\mathbf{w}, \sigma)$. The RS multicasts w.p. 1 under $\pi^*(\cdot)$ only if the current channel state precedes ζ . Our result explicitly characterizes the optimal tradeoff point in the opportunism.

Although we have found the solution structure, it proves to be difficult to compute the exact solution. Computing the threshold ζ for (MW) requires the knowledge of $\boldsymbol{\eta}(k)$, which is the channel state with the k th largest normalized revenue. We briefly argue that determining $\boldsymbol{\eta}(k)$ for given \mathbf{w} and k is related to an NP-hard problem called the k TH LARGEST SUBSET [25], which is defined as follows. For a finite multiset A , let $\|A\| := \sum_{a \in A} a$ denote the sum-element of A .

INSTANCE: Finite multiset A , positive integer k and c .

QUESTION: Are there k or more distinct subsets $B \subseteq A$ such that $\|B\| \leq c$?

We will refer to the sum-element of a subset as a *subset-sum*. (MW) is related to the k TH LARGEST SUBSET as follows. Let us define the multiset A such that $A := \{w_1, \dots, w_n\}$. For channel state $s = (s_1, \dots, s_n)$, define multiset $B_s := \{w_i | s_i = 1, i \in I\}$. B_s is a subset of A , and we have that $\|B_s\| = \sum_{i \in I} w_i s_i = \rho(\mathbf{w}, s)$. Thus, the set of revenues $\{\rho(\mathbf{w}, s)\}$ is a collection of all possible subset-sums of A . Consequently, in (MW), computing $\boldsymbol{\eta}(k^*)$, i.e., finding the k^* th

largest normalized revenue, is at least as hard as finding the k^{th} largest revenue, or, equivalently, finding the k^{th} largest subset-sum of A .

VI. LOW-COMPLEXITY RELAY SCHEDULING POLICY

Next we propose a low-complexity scheduling policy that achieves low decoding delays. Below, we summarize our policy.

First, we introduce the Weight-Based Priority (WBP) rule which we propose as a simplified measure of channel quality intended for low-complexity policy computations. We approximately solve (MW) based on the WBP rule and obtain a threshold-based fluid policy. Next, we map the fluid policy to an online policy for the original discrete networks, which we call *policy translation*. Policy computation and translation are performed periodically; in each period, the weights used for policy computation are adjusted according to the users' queue states. We will adopt the framework proposed in [26] for policy translation and renewal, as discussed later.

A. Weight-Based Priority (WBP) Rule

Solving (MW) is difficult, primarily because we use the normalized revenue as a measure of channel quality. Specifically, in Section V, we showed that the precedence relations among channel states have a combinatorial nature. Instead, we propose WBP, a simplified precedence rule for ranking channel states. Prior to that, we introduce the following definitions.

Definition 1: For a k -dimensional binary vector $\mathbf{x} = (x_1, \dots, x_k)$, we define $\langle x_1 \cdots x_k \rangle$ as the k -bit nonnegative binary number associated with \mathbf{x} where x_1 is the most significant bit, i.e.,

$$\langle x_1 \cdots x_k \rangle := \sum_{i=1}^k x_i 2^{k-i}.$$

Definition 2: For two binary vectors $\mathbf{x} = (x_1, \dots, x_k)$ and $\mathbf{y} = (y_1, \dots, y_k)$ of length $k \in \mathbb{Z}_+$, we define $\mathbf{x} \succ \mathbf{y}$ if

$$\langle x_1 \cdots x_k \rangle > \langle y_1 \cdots y_k \rangle.$$

The notation “ \prec ” is defined in a similar way. The definitions of the operators “ \succ ” and “ \prec ” are extended to compare a binary vector with a scalar quantity. For example, for some $\xi \in \mathbb{R}$, $\mathbf{x} \succ \xi$ means $\langle x_1 \cdots x_k \rangle > \xi$.

Using the above notations, we introduce the WBP rule as follows. Without loss of generality, assume that $w_1 \geq \dots \geq w_n$, i.e., that the users are indexed in descending order of weight. Consider two channel states $\mathbf{s}, \boldsymbol{\sigma} \in S$ such that $\mathbf{s} \neq \boldsymbol{\sigma}$. WBP is a precedence rule such that, we say that \mathbf{s} precedes $\boldsymbol{\sigma}$ if $\mathbf{s} \succ \boldsymbol{\sigma}$. In other words, we compare the channel states by treating them as binary numbers. For example, suppose that there are three users. Consider channel states $\mathbf{s} = (1, 0, 0)$ and $\boldsymbol{\sigma} = (0, 1, 1)$. Since $\langle 100 \rangle = 4$ is greater than $\langle 011 \rangle = 3$, we say \mathbf{s} is a better channel state than $\boldsymbol{\sigma}$ under the WBP rule. Recall that, in the binary number $\langle s_1 \dots s_n \rangle$ associated with channel state \mathbf{s} , the i th bit s_i represents the channel state of user i whose weight is w_i . Because $w_1 \geq \dots \geq w_n$, we see that the WBP rule gives higher priority to users associated with larger weights. Later,

we will set a user's weight to the remaining workload for that user, as in max-weight type schedulers [27]. Thus, by prioritizing users with high workloads, the WBP rule serves as a simple (yet effective) measure of channel quality; we will show this in the sequel.

We will use the WBP rule to approximately solve (MW) as follows. Consider the following problem:

$$\begin{aligned} (\mathbf{P}\text{-WBP}) \quad & \underset{\pi}{\text{maximize}} \quad \sum_{i \in I} w_i r_i \\ & \text{subject to} \quad (24) - (25), (31) \\ & \pi(\mathbf{s}) = \begin{cases} 1, & \mathbf{s} \succ \xi, \\ \beta, & \mathbf{s} = \xi, \\ 0, & \mathbf{s} \prec \xi, \end{cases} \quad (34) \\ & \xi \in \{0, \dots, 2^n - 1\}, \quad \beta \in [0, 1). \quad (35) \end{aligned}$$

(P-WBP) is identical to (MW), except that $\pi(\cdot)$ is constrained to be a threshold-based policy under the WBP rule, i.e., the RS should multicast w.p. 1 only if the binary number associated with the current channel state exceeds a threshold ξ .

Algorithm 1 exhibits an algorithm to determine the optimal threshold parameters ξ and β in (P-WBP). According to (31), the optimal ξ is the maximum ξ such that

$$\sum_{\mathbf{s} \in S, \mathbf{s} \geq \xi} p(\mathbf{s})[1 + s_i b] \geq 1. \quad (36)$$

The LHS of (36), as well as the objective of (P-WBP), are monotonically decreasing in ξ . In Algorithm 1, we initialize ξ with 2^{n-1} which is the median element in S , and iterate the following to find the optimal ξ : (i) calculate the LHS of (36), (ii) compare the result with 1, and (iii) adjust ξ accordingly to make (36) as tight as possible. This is the well-known *bisection method*. Because the searching space of ξ has 2^n elements, it takes $O(n)$ steps to determine the optimal ξ using bisection. Thus, Algorithm 1 has a complexity of $O(n)$.

B. Discrete-Review Policy

In this subsection, we will construct an online policy for the original network problem. The Discrete-Review (DR) policy [26] is a general method for translating a deterministic fluid policy into an implementable policy for stochastic networks. Under the DR policy, time is divided into *review periods* of constant length. At the beginning of each review period, we review the system state and formulate a fluid problem with a *safety-stock requirement* [28]. The safety-stock requirement is an additional constraint to fluid problems that guards against undesirable “boundary behaviors”, such as starvation of resources. Based on the solution to the formulated fluid problem, a scheduling policy for the stochastic network is implemented.

We will construct the DR policy associated with our problem (called WBP-DR) as follows. Let the integer $L > 0$ denote the length of the review period, and let the integer $u > 0$ be the safety-stock parameter. Denote the time index of the beginning of each review period by $l_j = jL$, $j = 0, 1, \dots$. At the beginning of j th review period, the RS queue length $B_R(l_j)$ and the number of innovative packets received for each user

Algorithm 1 Finding the Optimal ξ and β

```

1:  $\xi \leftarrow 2^{n-1}$ , Residual  $\leftarrow 1$ , SumProb  $\leftarrow p_1(1 + p_{i^b})$ .
2: if  $i^b = 1$  then
3:   SumProb  $\leftarrow 2p_1$ .
4: end if
5: for  $j = 1$  to  $n - 1$  do
6:   if SumProb  $\leq$  Residual then
7:     Residual  $\leftarrow$  Residual  $-$  SumProb,  $\xi \leftarrow \xi - 2^{n-j-1}$ .
8:     if  $j = i^b - 1$  then
9:       SumProb  $\leftarrow \frac{2(1-p_j)p_{j+1}}{p_j(1+p_{j+1})}$  SumProb.
10:    else if  $j = i^b$  then
11:      SumProb  $\leftarrow \frac{(1-p_j)p_{j+1}}{2p_j}$  SumProb.
12:    else
13:      SumProb  $\leftarrow \frac{(1-p_j)p_{j+1}}{p_j}$  SumProb.
14:    end if
15:  else
16:     $\xi \leftarrow \xi + 2^{n-j-1}$ .
17:    if  $j = i^b - 1$  then
18:      SumProb  $\leftarrow \frac{2p_{j+1}}{1+p_{j+1}}$  SumProb.
19:    else
20:      SumProb  $\leftarrow p_{j+1}$  SumProb.
21:    end if
22:  end if
23: end for
24: if SumProb  $\leq$  Residual then
25:   Residual  $\leftarrow$  Residual  $-$  SumProb,  $\xi \leftarrow \xi - 1$ .
26:   if  $i^b = n$  then
27:     SumProb  $\leftarrow \frac{1-p_n}{2p_n}$  SumProb.
28:   else
29:     SumProb  $\leftarrow \frac{1-p_n}{p_n}$  SumProb.
30:   end if
31: end if
32:  $\beta \leftarrow$  Residual/SumProb.

```

$\{\hat{B}_i(l_j), i \in I\}$ are observed; the following problem, which is similar to (P-WBP), is then constructed and to be solved:

$$\begin{aligned}
(\mathbf{P-DR}) \quad & \underset{\pi}{\text{maximize}} \sum_{i \in I} w_i r_i \\
& \text{subject to (24) - (25), (34) - (35),} \\
& B_R(l_j) - \hat{B}_{i^b}(l_j) + L(r_{\text{in}} - r_{i^b}) \geq u, \\
& w_i = [K - \hat{B}_i(l_j)]^+, \quad \forall i \in I. \quad (38)
\end{aligned}$$

From (38), the weight of each user is set to the remaining number of innovative packets to be received. Constraint (37) is the safety-stock requirement; the expected RS queue length at the end of each review period should be at least u greater than the expected number of innovative packets received by the bottleneck user. This constraint helps to prevent the bottleneck user from receiving noninnovative packets. Recall that, by adding constraints (34) and (35) to (P-DR), we are optimizing under the WBP rule. By solving (P-DR), we obtain policy $\pi^*(\cdot)$, which will be implemented over the next review period $\{l_j, \dots, l_j + L - 1\}$. $\pi^*(s)$ is implemented as the probability to perform multicasting by the RS in a time slot, under the

Algorithm 2 WBP-DR Policy

```

1: for Each time slot  $l = jL$ ,  $j = 0, 1, 2, \dots$  do
2:   Observe  $\hat{B}_i(l)$ ,  $\forall i \in I$ . Set  $w_i \leftarrow [K - \hat{B}_i(l)]^+$ ,  $\forall i \in I$ .
3:   Sort  $w_1, \dots, w_n$  in descending order. If multiple users have the same weight, prioritize users with worse channel conditions, i.e., smaller  $p_i$  values. Assign user indices in the order of the users' priorities.
4:   Solve (P-DR) using Algorithm 1 by setting  $1 - \frac{u + \hat{B}_{i^b}(l) - B_R(l)}{L}$  as the initial value of "Residual"; obtain threshold parameters  $\xi$  and  $\beta$ .
5:   for each time slot  $\tau$  in the review period do
6:     if there are no innovative packets to be fetched from the BS queue then
7:       if  $B_R(\tau) > 0$  then
8:         RS performs a multicast.
9:       end if
10:      continue \(* for loop *\
11:    end if
12:    Observe the channel state  $s$  and RS queue length  $B_R(\tau)$ .
13:    if  $B_R(\tau) = 0$  then
14:      RS performs a fetch.
15:    else if  $0 < B_R(\tau) < K$  then
16:      if  $s > \xi$  then
17:        RS performs a multicast.
18:      else if  $s < \xi$  then
19:        RS performs a fetch.
20:      else
21:        RS performs a multicast (resp. fetch) w.p.  $\beta$  (resp.  $1 - \beta$ ).
22:      end if
23:    else
24:      RS performs a multicast.
25:    end if
26:  end for
27: end for

```

condition that the channel state is s . WBP-DR is described in Algorithm 2. Guidelines for setting parameters L and u will be provided in the next section.

VII. PERFORMANCE ANALYSIS

In this section, we will characterize the asymptotic performance of the WBP-DR policy. Let $L^{(m)} := f_L(mK)$ and $u^{(m)} := \alpha L^{(m)}$ denote parameters L and u , respectively, in the m th system, where $\alpha \geq 1$ and $f_L(\cdot)$ is a function such that

$$\frac{f_L(x)}{\log(x)} \rightarrow \infty \quad \text{and} \quad \frac{f_L(x)}{x} \rightarrow 0, \quad \text{as } x \rightarrow \infty. \quad (39)$$

For example, one can set $L^{(m)} = f_L(mK) = \lfloor \sqrt{mK} \rfloor$. (39) implies that the length of a review period tends to infinity in the fluid limit. We have that

$$\frac{L^{(m)}}{m} = \frac{f_L(mK)}{m} \rightarrow 0, \quad \text{as } m \rightarrow \infty,$$

i.e., the scaled review period has zero duration in the fluid limit. In the m th system, the weights of users at the beginning of j th review period are adjusted to

$$w_i = [mK - \hat{B}_i(jL^{(m)})]^+, \quad \forall i \in I. \quad (40)$$

Next, we present an upper bound on the asymptotic decoding delay of WBP-DR. We first present the following lemma on the scaled queue length trajectories $b_R^{(m)}(t)$ and $b_i^{(m)}(t)$.

Lemma 2: For any $\epsilon > 0$, $0 < \delta \leq \frac{L^{(m)}}{m}$ and $j \in \mathbb{Z}_+$, there exists some $M \in \mathbb{Z}_+$ such that

$$\left| \frac{b_i^{(m)}(\frac{jL^{(m)}}{m} + \delta) - b_i^{(m)}(\frac{jL^{(m)}}{m})}{\delta} - \sum_{s \in S} p(s) \pi^*(s) s_i \right| \leq \epsilon, \\ \left| \frac{b_R^{(m)}(\frac{jL^{(m)}}{m} + \delta) - b_R^{(m)}(\frac{jL^{(m)}}{m})}{\delta} - \sum_{s \in S} p(s) [1 - \pi^*(s)] \right| \leq \epsilon,$$

almost surely for all $m > M$, where $\pi^*(s)$ denotes the solution to **(P-WBP)** with the weights defined in (40).

Proof: See Appendix G. ■

Lemma 2 indicates that $b_R^{(m)}(t)$ and $b_i^{(m)}(t)$, $i \in I$ achieve almost constant rates within each scaled review period for large values of m . That is, the queue trajectories are roughly piecewise linear over time for large values of m , which is useful in proving the next theorem.

Theorem 5: WBP-DR incurs an asymptotic decoding delay that is at most twice the optimal value.

Proof: See Appendix H for details. ■

In proving Theorem 5, we utilize the fact that the service priority of users is repeatedly adjusted based on the WBP rule; combining this fact with Lemma 2 and the constraint (21) on innovative fluid rates, we are able to bound the service rate achieved by the bottleneck user under fluid scaling. In the sequel, we will show via simulation that WBP-DR performs substantially better than the derived bound in the discrete-time stochastic network. Prior to that, we present a tighter bound for two-user systems.

Theorem 6: When $n = 2$, WBP-DR incurs an asymptotic decoding delay that is at most $4/3$ that of the optimal value.

Proof: See Appendix I for details. ■

In the proof of Theorem 6, we explicitly show how the priority among users varies during transmission and compute the innovative fluid rates achieved by users. This enables us to derive a tighter performance bound than that in Theorem 5.

We comment on the implications of the proposed algorithm design and analysis. WBP-DR is based on the framework of discrete-review (DR) policies [26]; however, our approach differs from the conventional construction of DR policies. Typically, a DR policy translates the solution of a deterministic fluid problem into an implementable policy for the original stochastic system. However, in our case, it is hard to solve the fluid problem. Instead, we formulated an *approximate* fluid problem **(P-WBP)**, and translated its solution to an online policy which is the proposed WBP-DR. The analysis of WBP-DR is, however, still challenging, because the fluid trajectories under adaptive policies are hard to characterize. In our analysis, we exploited several properties of the solution to **(P-WBP)**, and those properties arise from a simplified rule

for channel prioritization, i.e., the WBP rule. As a result, we were able to bound the rates of fluid trajectories, from which we derived guarantees in asymptotic decoding delay. We believe this is a novel approach for the performance analysis of fluid control problems.

VIII. SIMULATION RESULTS

In this section, we evaluate the performance of WBP-DR in discrete-time stochastic networks via simulation. We will make comparisons with the following schemes:

ARQ: This is the traditional automatic repeat request (ARQ) where the packets are not coded. The RS first fetches a packet from the BS, then repeatedly multicasts the packet until every user has received a copy. This procedure is repeated K times.

GREEDY: The RS greedily performs a fetch until it receives K original packets. Then, the RS repeatedly multicasts coded packets until all users have received K coded packets.

LB (Lower Bound): This is the numerically obtained solution to **(P)**. It has been shown in [28] that, in the case of finite-time horizon problems, the optimal cost of an MDP is bounded below by the solution to the associated fluid problem.

In our simulations, we set $K = 5 \times 10^3$. For the Rayleigh fading channel model, we set $f_m = 50$ Hz and $f_s = 10^3$ packets/sec. This results in $f_m/f_s = 0.05$, which is a typical value of the ratio f_m/f_s , e.g., see [29]. In order to generate a DTMC $X_i(l)$ with ON probability p_i , we leverage (4)–(6) to set the transition probabilities of $X_i(l)$; for example, $q_{0,1}^i$ is given by

$$q_{0,1}^i = \sqrt{-2\pi \ln(p_i)} \cdot \frac{p_i}{1 - p_i} \cdot \frac{f_m}{f_s}. \quad (41)$$

For WBP-DR, we set $L = u = 200$.

Fig. 4(a) shows the average decoding delay for symmetric-user systems with $n = 2$ and $p_1 = p_2 = p$ where we vary p in the plot. We observe that the decoding delay decreases as p increases for all policies. WBP-DR performs well, incurring a delay that is only 1–6% higher than the lower bound. ARQ and GREEDY incur delays that are up to 329% and 30% higher than WBP-DR, respectively.

The case of symmetric-user systems with $n = 10$, $p_i = p, \forall i \in I$, and varying p is shown in Fig. 4(b). WBP-DR performs best, achieving a delay that is 1–7% higher than the lower bound. The performance of GREEDY is close to WBP-DR when p is close to 1, which can be explained as follows. Due to the symmetry in channel distributions, larger p means better channel conditions for every user. Thus, if p is close to 1, the channel being in ON state is no longer a scarce opportunity for any user, i.e., opportunism has little impact on the performance. However, as p becomes smaller, the gap of the decoding delays between GREEDY and WBP-DR increases. A similar argument as the above holds for Fig. 4(a) when $n = 2$. We will discuss the effect of n in more detail in the sequel. Meanwhile, the performance of ARQ declines quickly with decreasing p . When $p \leq 0.6$, the delay incurred by ARQ is at least 496% higher than that of WBP-DR.

In Fig. 4(c), we consider symmetric-user systems with fixed $p = 0.5$ and varying n . WBP-DR incurs a decoding

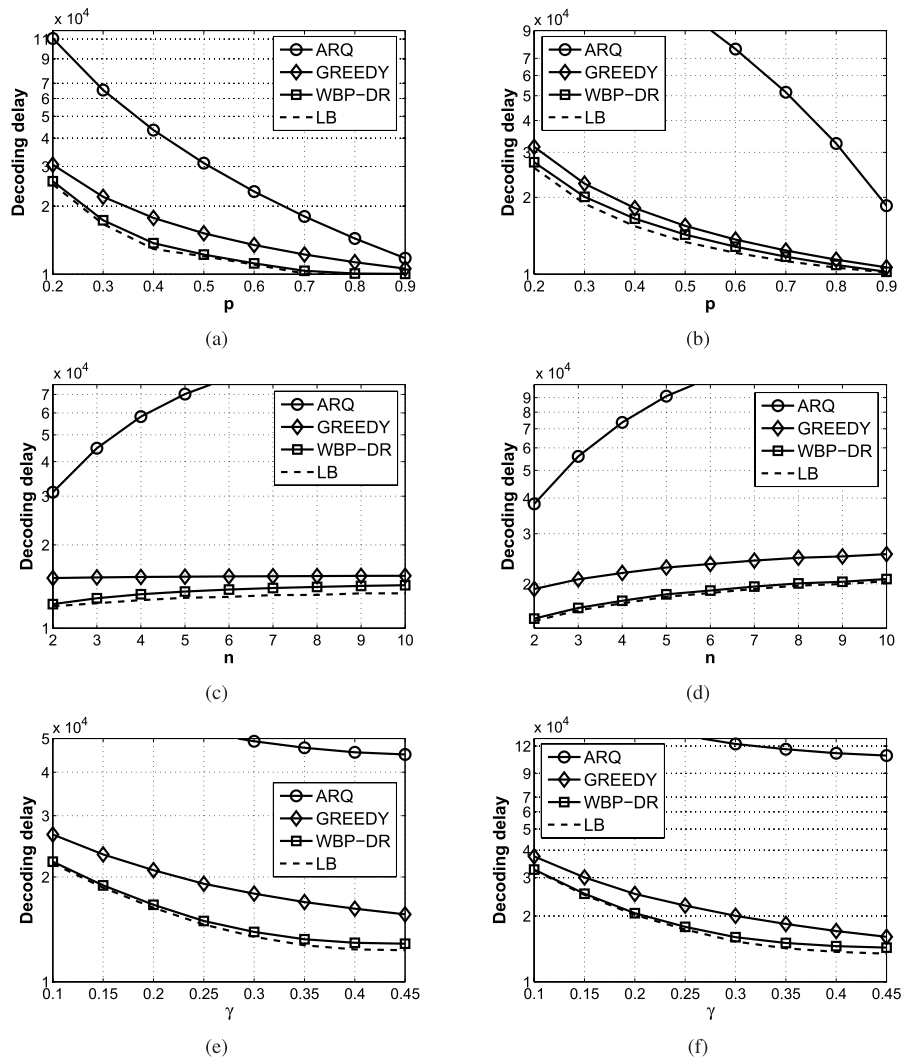


Fig. 4. Comparison of decoding delays based on simulations. The delays are plotted on a logarithmic scale. (a) symmetric channels with $n = 2$ and varying p ; (b) symmetric channels with $n = 10$ and varying p ; (c) symmetric channels with $p = 0.5$ and varying n ; (d) asymmetric channels with p_i 's are uniformly and independently generated from $[0.2, 0.8]$ and varying n ; (e) asymmetric channels with $n = 3$, p_i 's are uniformly and independently generated from $[\gamma, 1 - \gamma]$ and varying γ ; (f) asymmetric channels with $n = 10$, p_i 's are uniformly and independently generated from $[\gamma, 1 - \gamma]$ and varying γ .

delay that is 2–7% higher than the lower bound and reduces decoding delay by up to 87% and 20% compared with ARQ and GREEDY, respectively. The delay incurred by GREEDY increases slowly with increasing n , because it is related to the maximum of n i.i.d. numbers of time slots taken by a user to receive K coded packets. The LB also increases gradually with n . The performance of WBP-DR is between that of LB and GREEDY. The gap between WBP-DR and GREEDY becomes narrow as n increases. We attempt to explain this by examining the limiting properties of multicast channels as follows.

Suppose we scale the system by making n large. Assume that for simplicity, at each time slot, the channel state of a user is distributed as Bernoulli(p). Denote the number of ON channels at a given time slot by a random variable $N \sim \text{Binomial}(n, p)$. Note that N becomes concentrated around its mean $\mathbb{E}[N] = np$ due to the law of large numbers. Intuitively, the opportunism of the multicast channel is closely related to the extent to which N deviates from the mean. However, as n increases, the deviation of N relative to the

mean reduces due to the concentration, i.e., the inherent opportunism of the multicast channel “degrades” with respect to the mean np . This partly explains why the GREEDY scheme, which is oblivious of opportunism, performs closer to WBP-DR and LB for larger n . However, the above arguments are applicable to symmetric systems. If the channels are statistically asymmetric, it is important to exploit the opportunism of the bottleneck user who will dominate the system performance. We examine this aspect in the following simulation.

We present simulation results for asymmetric-user systems. In Fig. 4(d), we have varied n from 2 to 10. For each n , the values of p_i 's are uniformly and independently generated from $[0.2, 0.8]$ for 10^4 times. The delay incurred by WBP-DR is within 2% of the lower bound. By contrast, ARQ and GREEDY incur delays that are 153–613% and 23–27% higher, respectively, than that of WBP-DR. Note that the gap between the delays achieved by WBP-DR and GREEDY is relatively larger than that in the case of symmetric channels, e.g., compare Fig. 4(c) and Fig. 4(d). This shows that, one should carefully exploit opportunism to deal with the

bottleneck user in asymmetric systems, by frequently reviewing users' buffer status and updating the policy accordingly, which is the main idea behind WBP-DR.

In Fig. 4(e), we consider asymmetric systems in which $n = 3$. The values of p_i 's are independently and uniformly drawn from $[\gamma, 1 - \gamma]$ where we vary γ in the plot. Therefore, the smaller the value of γ , the more variability in the values of p_i 's. Interestingly, WBP-DR performs even better, i.e., closer to LB, with smaller values of γ , which we explain as follows. A smaller γ leads to a larger gap (on average) between p_{i^b} and other users' parameters. As a result, user i^b will further dominate the system performance because of its relatively poor channel condition, i.e., user i^b will often be the user who has the smallest number of innovative packets during the transmission. Because WBP-DR prioritizes users with a smaller number of innovative packets, user i^b will be served at the highest priority most of the time, which is also expected for the optimal policy. WBP-DR incurs a delay that is 17–23% and 70–72% lower than that of GREEDY and ARQ, respectively. Fig. 4(f) considers a similar scenario as Fig. 4(e) with $n = 10$. WBP-DR incurs a lower delay by 11–20%, relative to GREEDY. Additionally, WBP-DR significantly outperforms ARQ, reducing the delay by up to 87%.

IX. EXTENSIONS AND DISCUSSIONS

So far we have assumed that the BS queue is backlogged with the source information block. In this section, we discuss some extensions of our problem when there exists a source of arrivals to the BS queue.

We introduce a *fragment arrival* model as follows. In this model, the original information block is divided into multiple *fragments*. The fragments arrive at the BS queue at arbitrary instants. It is possible that, at some instant, all the previously arrived fragments in the BS queue have been transmitted to the RS queue, however a new fragment has not yet arrived at the BS queue. In this case, the BS queue is forced to idle, i.e., the queue can be “depleted”, and thus the system is no longer backlogged with the source block.

We will show that, the fluid policy which solves (P) is essentially optimal for the fragment arrival model in the fluid regime. Let $b_B(t)$ denote the difference between the total amount of fluid received by the BS and the total amount of fluid received by the RS up to time t . If all the fluid in the BS queue has been transmitted to the RS, but if no new fragment has arrived until time t , then $b_B(t) = 0$, i.e., the BS queue is depleted. We define a fluid policy denoted by \mathcal{P} as follows. If $b_B(t) > 0$, we use the policy which solves (P); otherwise, we let the system idle. One can prove that \mathcal{P} is optimal for the fragment arrival model as follows.

Theorem 7: Suppose M fragments of size K_1, K_2, \dots, K_M , arrive at the BS queue in the fluid regime. The arrival epoch of the i th fragment is denoted by τ_i , and $0 = \tau_1 \leq \tau_2 \leq \dots \leq \tau_M$. The fragments comprise the entire block, i.e.,

$$\sum_{i=1}^M K_i = K.$$

Then it follows that policy \mathcal{P} for the fluid model is optimal.

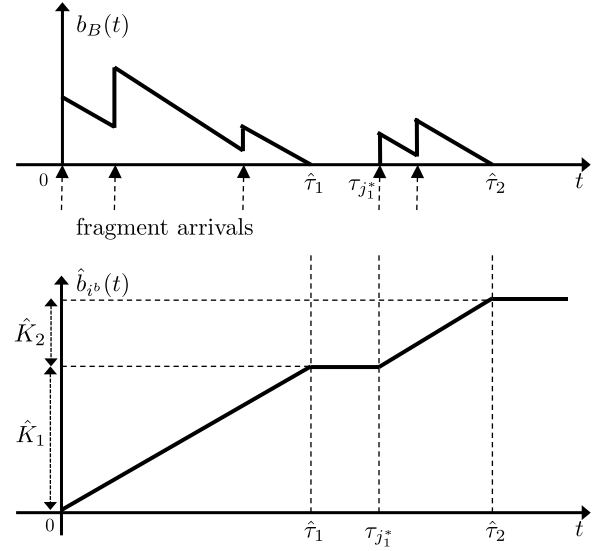


Fig. 5. Example of the trajectories for the fragment arrival model under policy \mathcal{P} . The plot in the above shows $b_B(t)$ which represents the amount of fluid arrived at the BS yet not received by the RS up to time t . The plot in the below shows the amount of innovative fluid received by the bottleneck user. The service rate of the innovative fluid is constant under \mathcal{P} when $b_B(t) > 0$.

Proof: Let $\hat{\tau}_1, \hat{\tau}_2, \dots$, denote the epochs at which $b_B(t)$ hits 0 under \mathcal{P} . Define

$$j_1^* := \min \{j | \tau_j > \hat{\tau}_1\}, \quad (42)$$

i.e., $\tau_{j_1^*}$ is the arrival time of the first segment since the BS queue became depleted for the first time under \mathcal{P} ; see Fig. 5. Let \hat{K}_1 denote the total amount of fluid arrived at the BS queue until $\hat{\tau}_1$, i.e.,

$$\hat{K}_1 := \sum_{i=1}^{j_1^*-1} K_i. \quad (43)$$

We will first show that, \mathcal{P} achieves the minimum delay in disseminating \hat{K}_1 amount of fluid as follows. The total amount of fluid fragments which arrived during $[0, \hat{\tau}_1)$ is \hat{K}_1 . Also we have $b_B(t) > 0$ for $t \in [0, \hat{\tau}_1)$, i.e., the BS queue is backlogged during $[0, \hat{\tau}_1)$. From Theorem 1 and 2, we know that \mathcal{P} serves users' queues with the innovative fluid at a fixed rate, provided that the BS queue is backlogged, i.e., $b_B(t) > 0$, and \mathcal{P} is optimal when the system is backlogged. Consequently, \mathcal{P} achieves the minimum delay which is given by $\hat{\tau}_1$. Moreover, by (29) and (30), the amount of fluid received by the RS and the amount of innovative fluid received by the users are both *exactly* \hat{K}_1 at time $\hat{\tau}_1$ under \mathcal{P} , i.e.,

$$b_R(\hat{\tau}_1) = \hat{b}_i(\hat{\tau}_1) = \hat{K}_1, \quad \forall i \in I.$$

The dissemination of \hat{K}_1 amount of fluid completes exactly at $t = \hat{\tau}_1$, and thus the system can stay idle until a new fragment arrives. Indeed \mathcal{P} will idle when the BS queue is depleted; see the definition of \mathcal{P} . Thus, the innovative fluid rate achieved by the users during the period $[\hat{\tau}_1, \tau_{j_1^*})$ is 0 under \mathcal{P} .

Now let us consider the second period during which the BS queue is backlogged. Let us make the following definitions

which are similar to (42) and (43):

$$j_2^* := \min \{j | \tau_j > \hat{\tau}_2\}, \quad \hat{K}_2 := \sum_{i=j_1^*}^{j_2^*-1} K_i,$$

i.e., \hat{K}_2 is the total amount of fluid arrived during the second “busy” period of the BS queue under \mathcal{P} . We claim that $\hat{\tau}_2$ is the minimum time for disseminating the fluid of the amount $\hat{K}_1 + \hat{K}_2$, which implies that \mathcal{P} is the optimal policy for the duration $[0, \hat{\tau}_2)$. We shall argue by contradiction as follows.

Suppose there exists a policy \mathcal{P}' which completes the dissemination of $\hat{K}_1 + \hat{K}_2$ amount of fluid strictly earlier than $\hat{\tau}_2$, say at time $\tilde{\tau} < \hat{\tau}_2$. First of all, we cannot have $\tilde{\tau} < \tau_{j_1^*}$, because the total amount of fluid arrived at the BS during $[0, \tau_{j_1^*})$ is exactly \hat{K}_1 . Thus we have that $\tilde{\tau} \in [\tau_{j_1^*}, \hat{\tau}_2)$, which implies that the amount of fluid disseminated during $[\tau_{j_1^*}, \tilde{\tau}]$ under \mathcal{P}' is at least \hat{K}_2 . We will show that, however, the minimum delay to disseminate \hat{K}_2 amount of fluid is $\hat{\tau}_2 - \tau_{j_1^*}$ which is strictly less than $\tilde{\tau} - \tau_{j_1^*}$, which leads to a contradiction, as follows.

Because \mathcal{P} has already disseminated \hat{K}_1 amount of fluid prior to time $\tau_{j_1^*}$, the amount of fluid disseminated during the period $[\tau_{j_1^*}, \hat{\tau}_2)$ is exactly \hat{K}_2 . Moreover, $b_B(t) > 0$ holds during this period, i.e., the BS queue is backlogged. Thus we can use a similar argument as the earlier part of this proof to show that \mathcal{P} achieves the minimum delay in disseminating \hat{K}_2 amount of fluid. Thus, the delay of disseminating \hat{K}_2 amount of fluid cannot be smaller than $\hat{\tau}_2 - \tau_{j_1^*}$ under any policy. This implies that such \mathcal{P}' does not exist.

We can extend the above argument to the subsequent “busy periods” of the BS queue. For example, by using the fact that \mathcal{P} is optimal for the duration $[0, \hat{\tau}_2)$, one can show that \mathcal{P} is optimal for the duration $[0, \hat{\tau}_3)$, and so forth. This establishes the optimality of \mathcal{P} . ■

Next we discuss discrete-time algorithms under the fragment arrival model. In discrete time, the BS queue may occasionally become depleted before the dissemination completes. Note that WBP-DR will exploit opportunism as long as the BS queue is backlogged, and will necessarily multicast if the BS queue is depleted, see lines 6–11 of Algorithm 2. Due to these underlying characteristics for Algorithm 2, WBP-DR can be extended to the fragment arrival model: the algorithm will exploit opportunism if the BS queue is backlogged, and will perform multicasting if the BS queue occasionally becomes depleted. Thus, WBP-DR prioritizes opportunism whenever possible. This heuristic is motivated by the observation such that the optimal fluid policy maximally prolongs the “Opportunistic Fetch” stage.

Although it appears to be difficult to analyze the performance of WBP-DR under the fragment arrival model, we may conclude the following. Theorem 7 shows that \mathcal{P} minimizes the delay for the fragment arrival model in fluid regime, and the simulation results show that WBP-DR performs very close to the optimal fluid policy. In conclusion, whenever the BS queue is backlogged, i) WBP-DR will perform similar to \mathcal{P} ; ii) WBP-DR will attempt to remain in the “Opportunistic Fetch” stage similar to \mathcal{P} . Thus, we can assume

that WBP-DR is approximately optimal for the fragment arrival model.

Another direction to extend our model is the *block arrival* model. In this model, the blocks of size K arrive at the system; let us consider an example of video multicast streaming. Suppose a sequence of video segments arrive at the BS queue as follows. Each segment consists of K packets, thus a segment corresponds to an information block. We assume the segments are disseminated to video multicast streaming subscribers sequentially, i.e., the next segment is requested and available for transmission only once the previous one has been delivered to all the receivers. Such a scheme is motivated by DASH [30] which is a widely used protocol for *unicast* video streaming. An example would be the system which serves a small to moderate number of users who are aligned to watch the same content via a streaming multicast session. Let us denote the block arrival rate to the BS queue by λ . Our objective for this model is to maximize the *block throughput*, i.e., to maximize λ subject to stability.

In order to maximize the throughput, we would like to deliver each segment to the users as fast as possible. Thus, the problem reduces to minimizing the dissemination delay of each segment of size K , which is the same problem as studied in the previous sections. Consequently, one can expect that WBP-DR performs close to the optimal policy for the throughput model when K is large.

In conclusion, we observe that the optimal fluid policy proposed in Section IV is applicable to both the fragment and block arrival models, i.e., it is asymptotically optimal for both the models. Since WBP-DR well-approximates the optimal fluid policy, WBP-DR will presumably perform well under these arrival models.

X. CONCLUSIONS

In this paper, we examined the problem of minimizing decoding delays using RLNC for multicasting in heterogeneous networks with relay stations. We formulated a deterministic fluid approximation of the original stochastic problem. The optimal fluid policy was shown to possess a threshold-based structure which captures the optimal tradeoff point between multicasting and fetching. We proposed a scheduling policy WBP-DR, which is based on prioritizing users with a large amount of unfinished work, and derived bounds on its asymptotic performance. We evaluated the performance of our algorithm in discrete-time stochastic networks via simulation; we showed that it incurs decoding delays that are close to the theoretical lower bounds. A future direction would be investigating the effect of other system parameters, such as packet length ν . For example, it would be interesting to study how to adaptively change ν using adaptive modulation and coding (AMC) at each time slot based on the current channel and queue states, so as to further improve the delay performance.

APPENDIX A PROOF OF THEOREM 1

Let us denote the optimal policy by $\pi^*(s, t)$. Let T_1^* and T_2^* be the durations of the Opportunistic Fetch stage and the

Flush stage, respectively, under policy $\pi^*(s, t)$. It is clear that $\pi^*(s, t) = 1$ for $T_1^* < t \leq T_1^* + T_2^*$. Consider policy $\pi'(s, t)$ such that

$$\pi'(s, t) := \begin{cases} \frac{1}{T_1^*} \int_0^{T_1^*} \pi^*(s, \tau) d\tau, & \text{if } 0 \leq t \leq T_1^*, \\ 1, & \text{if } T_1^* < t \leq T_1^* + T_2^*. \end{cases}$$

In what follows, we will use superscript $*$ (resp. $'$) to indicate the notation associated with policy $\pi^*(s, t)$ (resp. $\pi'(s, t)$), e.g., $r_{\text{in}}^*(t)$ (resp. $r'_{\text{in}}(t)$) is the inflow rate to the RS under policy $\pi^*(s, t)$ (resp. $\pi'(s, t)$). It is easy to verify that

$$r'_i(t) = \begin{cases} \frac{1}{T_1^*} \int_0^{T_1^*} r_i^*(\tau) d\tau, & \text{if } 0 \leq t \leq T_1^*, \\ p_i, & \text{if } T_1^* < t \leq T_1^* + T_2^*. \end{cases}$$

$$r'_{\text{in}}(t) = \begin{cases} \frac{1}{T_1^*} \int_0^{T_1^*} r_{\text{in}}^*(\tau) d\tau, & \text{if } 0 \leq t \leq T_1^*, \\ 0, & \text{if } T_1^* < t \leq T_1^* + T_2^*. \end{cases}$$

We also have that

$$b'_i(T_1^*) = \int_0^{T_1^*} r'_i(t) dt = b_i^*(T_1^*),$$

$$b'_R(T_1^*) = \int_0^{T_1^*} r'_{\text{in}}(t) dt = K = b_R^*(T_1^*).$$

Thus, the duration of the Opportunistic Fetch stage under policy $\pi'(s, t)$ is exactly T_1^* . It is easy to verify that $\pi'(s, t)$ satisfies (19) and (20); therefore, $\pi'(s, t)$ is a feasible policy. The fluid rates are constant under policy $\pi'(s, t)$ during the Opportunistic Fetch stage, i.e.,

$$r'_i(t) = \frac{b_i^*(T_1^*)}{T_1^*}, \quad r'_{\text{in}}(t) = \frac{b_R^*(T_1^*)}{T_1^*}, \quad 0 \leq t \leq T_1^*.$$

Combining the above with (11), we have that

$$\hat{r}'_i(t) = \min\{r'_i(t), r'_{\text{in}}(t)\} = \min\left\{\frac{b_i^*(T_1^*)}{T_1^*}, \frac{b_R^*(T_1^*)}{T_1^*}\right\}$$

for $0 \leq t \leq T_1^*$, which further implies that

$$\hat{b}'_i(T_1^*) = \min\{b_i^*(T_1^*), b_R^*(T_1^*)\} \geq \hat{b}_i^*(T_1^*).$$

Thus, the duration of the Flush stage under policy $\pi'(s, t)$ satisfies

$$\max_{i \in I} \left\{ \frac{K - \hat{b}'_i(T_1^*)}{p_i} \right\} \leq \max_{i \in I} \left\{ \frac{K - \hat{b}_i^*(T_1^*)}{p_i} \right\} = T_2^*.$$

Therefore, policy $\pi'(s, t)$ incurs no higher decoding delay than policy $\pi^*(s, t)$, which proves the theorem.

APPENDIX B PROOF OF THEOREM 2

We first introduce the following lemma.

Lemma 3: There exists an optimal policy for (F) such that

$$r_{\text{in}} = \min_{i \in I} \{r_i\}. \quad (44)$$

The proof of Lemma 3 is provided in Appendix C. Note that, the Flush stage has zero duration under the policies satisfying (44), because (22) is 0 for all $i \in I$.

Thus, minimizing the decoding delay is equivalent to maximizing r_{in} , or equivalently, $\min_{i \in I} \{r_i\}$. Hence, (F) reduces to

$$(\tilde{\mathbf{P}}) \quad \begin{aligned} & \underset{\pi}{\text{maximize}} \quad \min_{i \in I} \{r_i\} \\ & \text{subject to (24) - (26), (44).} \end{aligned}$$

Lemma 4: A necessary condition for the optimality of $(\tilde{\mathbf{P}})$ is

$$\min_{i \in I} \{r_i\} = r_{\text{in}}. \quad (45)$$

The proof Lemma 4 is provided in Appendix D. By Lemma 4, $(\tilde{\mathbf{P}})$ is equivalent to

$$\begin{aligned} & \underset{\pi}{\text{maximize}} \quad \min_{i \in I} \{r_i\} \\ & \text{subject to } r_{\text{in}} = r_{\text{in}}^b, \\ & \quad (24) - (26), (45). \end{aligned}$$

Note that, (P) is a relaxation of the above problem by dropping the constraint (45).

Lemma 5: The constraint (45) is a necessary condition for the optimality of (P).

Lemma 5 can be proved in a similar manner as Lemma 4. According to Lemma 4 and 5, $(\tilde{\mathbf{P}})$ and (P) are equivalent. Because $(\tilde{\mathbf{P}})$ and (F) are equivalent, the theorem is proved.

APPENDIX C PROOF OF LEMMA 3

We will first argue that any policy under which $r_{\text{in}} < \min_{i \in I} \{r_i\}$ cannot be optimal. Suppose $r_{\text{in}} < \min_{i \in I} \{r_i\}$ holds under the optimal policy $\pi(\cdot)$. Due to (21), all users achieve the same innovative fluid rate given by r_{in} . We can revise $\pi(\cdot)$ by decreasing $\pi(s)$ for some $s \in S$ such that $r_{\text{in}} = \min_{i \in I} \{r_i\}$ holds under $\pi(\cdot)$. Under the revised policy, all users still achieve the same innovative fluid rate r_{in} , but r_{in} has been increased because, by definition, it can only increase by decreasing $\pi(\cdot)$. Thus, the decoding delay would be decreased under the revised policy, which is a contradiction.

Now, denote the optimal policy by $\pi^*(\cdot)$ and the associated rates by r_i^* , r_{in}^* , and \hat{r}_i^* . Let T_1^* and T_2^* be the durations of the Opportunistic Fetch stage and the Flush stage, respectively, under $\pi^*(\cdot)$. We have that

$$\begin{aligned} r_{\text{in}}^* & \geq \min_{i \in I} \{r_i^*\}, \quad r_{\text{in}}^* T_1^* = K, \\ r_i^* T_1^* + p_i T_2^* & \geq K, \quad \forall i \in I, \\ \exists j \in I : & \quad r_j^* T_1^* + p_j T_2^* = K. \end{aligned} \quad (46)$$

Consider a rate vector $\mathbf{r}' = (r'_1, \dots, r'_n, r'_{\text{in}})$ such that

$$r'_{\text{in}} = \frac{r_{\text{in}}^* T_1^*}{T_1^* + T_2^*} = \frac{K}{T_1^* + T_2^*}, \quad (47)$$

$$r'_i = \frac{r_i^* T_1^* + p_i T_2^*}{T_1^* + T_2^*} \geq \frac{K}{T_1^* + T_2^*}, \quad \forall i \in I. \quad (48)$$

Note that \mathbf{r}' can be achieved under policy $\pi'(\cdot)$ such that

$$\pi'(s) = \frac{\pi^*(s) T_1^* + T_2^*}{T_1^* + T_2^*}, \quad \forall s \in S.$$

It is clear that the duration of the Opportunistic Fetch stage is $T_1^* + T_2^*$ under policy $\pi'(\cdot)$. From (46) and (48), there exists some $j \in I$ such that

$$r'_j = \frac{r_j^* T_1^* + p_j T_2^*}{T_1^* + T_2^*} = \frac{K}{T_1^* + T_2^*}. \quad (49)$$

Combining (47), (48), and (49), we have that

$$r'_{\text{in}} = \min_{i \in I} \{r'_i\}. \quad (50)$$

By (21), we have that

$$\hat{r}'_i = \min\{r'_i, r'_{\text{in}}\} = r'_{\text{in}} = \frac{K}{T_1^* + T_2^*}, \quad \forall i \in I,$$

which implies that all users receive K units of innovative fluid simultaneously at the end of the Opportunistic Fetch stage. Thus, the duration of the Flush stage is 0, and the decoding delay is given by $T_1^* + T_2^*$ under $\pi'(\cdot)$. Hence $\pi'(\cdot)$ is optimal for **(F)**, and also satisfies (44) because of (50).

APPENDIX D PROOF OF LEMMA 4

Without loss of generality, let $\pi^*(\cdot)$ denote the optimal policy for **(P)** such that

$$r_{\text{in}}^* = \min_{i \in I} \{r_i^*\} = r_j^* < r_i^*, \quad \forall i \neq j \quad (51)$$

where r_i^* and r_{in}^* are the associated rates with policy $\pi^*(\cdot)$, and j is some user index such that $p_j > p_{i^b}$. There must exist states $s^{(1)}$ and $s^{(2)}$ that satisfy

$$\begin{aligned} s_{i^b}^{(1)} &= 1, & s_{i^b}^{(2)} &= 0, \\ s_j^{(1)} &= 0, & s_j^{(2)} &= 1, \\ s_i^{(1)} &= s_i^{(2)}, & \forall i \in I, i \neq i^b, j, \end{aligned}$$

such that $\pi^*(s^{(1)}) > \pi^*(s^{(2)})$; otherwise, we have that $r_{i^b}^* \leq r_j^*$, which contradicts (51). Consider a policy $\pi'(\cdot)$ such that

$$\begin{aligned} \pi'(s^{(1)}) &= \pi^*(s^{(1)}) - \frac{\Delta}{p(s^{(1)})}, \\ \pi'(s^{(2)}) &= \pi^*(s^{(2)}) + \frac{\Delta}{p(s^{(2)})}, \\ \pi'(s) &= \pi^*(s), \quad \forall s \in S, s \neq s^{(1)}, s^{(2)} \end{aligned}$$

where Δ satisfies

$$0 < \Delta \leq \min\{p(s^{(1)})\pi^*(s^{(1)}), p(s^{(2)})[1 - \pi^*(s^{(2)})]\}. \quad (52)$$

(52) guarantees that $\pi'(s) \in [0, 1]$ for all $s \in S$. Let r'_i and r'_{in} be the associated rates with policy $\pi'(\cdot)$. We have that

$$\begin{aligned} r'_{\text{in}} &= r_{\text{in}}^*, & r'_{i^b} &= r_{i^b}^* - \Delta, & r'_j &= r_j^* + \Delta, \\ r'_i &= r_i^*, & \forall i \in I, i \neq i^b, j, \end{aligned}$$

Furthermore, if Δ satisfies

$$0 < \Delta \leq \frac{r_{i^b}^* - r_j^*}{2}, \quad (53)$$

we have that $r'_{\text{in}} < r'_{i^b} \leq r_{i^b}^*$. Therefore, by choosing a value of Δ that satisfies (52) and (53), we can construct policy $\pi'(\cdot)$ such that

$$\min_{i \in I} \{r'_i\} = r'_k > r'_{\text{in}} = r_{\text{in}}^* = \min_{i \in I} \{r_i^*\},$$

where $k = \operatorname{argmin}_{i \in I} \{r'_i\}$. Let $\mu = r'_k - r'_{\text{in}}$. Consider the following policy $\pi^\dagger(\cdot)$:

$$\begin{aligned} \pi^\dagger(s^{(3)}) &= \pi'(s^{(3)}) - \frac{\mu}{2p(s^{(3)})}, \\ \pi^\dagger(s) &= \pi'(s), \quad \forall s \in S, s \neq s^{(3)}. \end{aligned}$$

where $s^{(3)} \in S$ such that $s_k^{(3)} = 1$. One can show that

$$\begin{aligned} r_{\text{in}}^\dagger &= r'_{\text{in}} + \frac{\mu}{2}, & r_k^\dagger &= r'_k - \frac{\mu}{2}, \\ r_i^\dagger &\geq r'_i - \frac{\mu}{2}, & \forall i \in I, i \neq k. \end{aligned}$$

Thus, we have that

$$\min_{i \in I} \{r_i^\dagger\} = r_k^\dagger = r_{\text{in}}^\dagger > r'_{\text{in}} = r_{\text{in}}^* = \min_{i \in I} \{r_i^*\},$$

which contradicts the optimality of $\pi^*(\cdot)$.

Note that the above proof can be extended to the case in which

$$r_{\text{in}}^* = \min_{i \in I} \{r_i^*\} = r_j^* < r_{i^b}^*, \quad \forall j \in \bar{I}$$

for some $\bar{I} \subset I$ such that $|\bar{I}| > 1$ and $i^b \notin \bar{I}$. We omit the details here to save space.

APPENDIX E PROOF OF THEOREM 3

We will reformulate **(P)** as an LP with respect to r_1, \dots, r_n . The feasible set associated with constraints (24)-(26) and (28) is denoted by \mathcal{A} such that

$$\mathcal{A} = \left\{ \pi \in [0, 1]^{|S|} \mid \sum_{s \in S} p(s)\pi(s)[1 + s_{i^b}] = 1 \right\}$$

where π represents the $|S|$ -dimensional vector $(\pi(s), s \in S)$. Define an $n \times |S|$ matrix \mathbf{H} such that

$$H_{i,j} = p(s)s_i, \quad \text{if } j = \langle s_1 \dots s_n \rangle + 1$$

for all $i \in I$ and $j \in \{1, \dots, 2^n\}$. Let $\mathbf{r} = (r_1, \dots, r_n)$. Then, the constraint (25) is given by $\mathbf{r} = \mathbf{H}\pi$. Thus, the feasible set for **(P)** is given by

$$\mathcal{B} = \{\mathbf{r} \in \mathbb{R}^n \mid \mathbf{r} = \mathbf{H}\pi, \pi \in \mathcal{A}\}.$$

\mathcal{A} is a convex polyhedron, and \mathcal{B} is obtained by an affine transformation of \mathcal{A} . Therefore, \mathcal{B} is also a convex polyhedron. **(P)** is equivalent to following LP

$$\text{maximize } \min_{i \in I} \{r_i\} \quad \text{subject to } \mathbf{r} \in \mathcal{B}. \quad (54)$$

Let the optimal value of (54) be $u^* > 0$, and define the set $\mathcal{C}_{u^*} := \{\mathbf{r} \mid \min_{i \in I} \{r_i\} \geq u^*\}$. By the separation theorem [31] for convex polyhedra, there exists a separating hyperplane $\mathcal{D} := \{\mathbf{r} \mid \mathbf{w}^T \mathbf{r} = u^*\}$ which separates $\operatorname{relint}(\mathcal{C}_{u^*})$ and $\operatorname{relint}(\mathcal{B})$,

where $\text{relint}(\cdot)$ denotes the relative interior of a set. This implies that u^* is also a solution to

$$\underset{\mathbf{r}}{\text{maximize}} \mathbf{w}^T \mathbf{r} \quad \text{subject to } \mathbf{r} \in \mathcal{B},$$

which is equivalent to (MW). The solution to (54) lies in \mathcal{D} ; thus, \mathcal{D} must be a supporting hyperplane [31] of set \mathcal{C}_{u^*} . It is straightforward to show that, for a hyperplane of form $\{\mathbf{r} | \mathbf{w}^T \mathbf{r} = u^*\}$ to be a supporting hyperplane of \mathcal{C}_{u^*} for any $u^* > 0$, it is necessary that the vector \mathbf{w} has non-negative entries. This completes the proof.

APPENDIX F PROOF OF THEOREM 4

Define new variables $x(s) := p(s)\pi(s)[1 + s_{ib}]$ for all $s \in S$. We can rewrite (MW) as

$$\begin{aligned} & \underset{x}{\text{maximize}} \sum_{s \in S} x(s) \hat{p}(\mathbf{w}, s) \\ & \text{subject to} \sum_{s \in S} x(s) = 1, \\ & 0 \leq x(s) \leq p(s)[1 + s_{ib}], \quad \forall s \in S. \end{aligned} \quad (55)$$

The above problem can be solved in a greedy manner as follows. First, we select $x(s)$ with the largest value of $\hat{p}(\mathbf{w}, s)$, i.e., $x(\eta(1))$. Then, we attempt to set $x(\eta(1))$ as close to $p(\eta(1))[1 + \eta_{ib}(1)]$ as possible until (55) is satisfied. If (55) cannot be satisfied even after setting $x(\eta(1)) = p(\eta(1))[1 + \eta_{ib}(1)]$, we select $x(\eta(2))$, and so on. This procedure is repeated until (55) is satisfied, which leads to the threshold-based solution stated in the theorem.

APPENDIX G PROOF OF LEMMA 2

Let $\pi'(s)$ denote the solution to (P-DR) formulated at the beginning of the j th review period in the m th system. We will first compare $\pi'(s)$ and $\pi^*(s)$. Recall $\pi^*(s)$ is the solution to (P-WBP) with input weights (40). Note that (P-DR) and (P-WBP) share the same weights; the only difference is that (P-DR) has the safety-stock requirement (37), whereas (P-WBP) has the constraint (31), or equivalently, (28). Note the (37) should be strictly satisfied at each review period. Following [26, Proof of Theorem 5.1], we have that

$$\frac{u^{(m)} + \hat{B}_{ib}(jL^{(m)}) - B_R(jL^{(m)})}{L^{(m)}} \rightarrow 0, \quad \forall j \geq \lceil \alpha \rceil,$$

almost surely as $m \rightarrow \infty$. Thus, for any $\epsilon > 0$, there exists some $M_1 \in \mathbb{Z}_+$ such that

$$|r_{in} - r_{ib}| = \left| \frac{u^{(m)} + \hat{B}_{ib}(jL^{(m)}) - B_R(jL^{(m)})}{L^{(m)}} \right| \leq \frac{\epsilon}{2^{n+1}} \quad (56)$$

for all $m > M_1$. Furthermore, (56) implies that

$$|\pi'(s) - \pi^*(s)| \leq \frac{\epsilon}{2^{n+1}p(s)[1 + s_{ib}]}, \quad \forall s \in S \quad (57)$$

for all $m > M_1$. According to the Functional Strong Law of Large Numbers (FSLN), for any $\epsilon > 0$, there exists some $M_2 \in \mathbb{Z}_+$ such that

$$\left| \frac{b_i^{(m)}(\frac{jL^{(m)}}{m} + \delta) - b_i^{(m)}(\frac{jL^{(m)}}{m})}{\delta} - \sum_{s \in S} p(s)\pi'(s)s_i \right| \leq \frac{\epsilon}{2}$$

for all $m > M_2$. Combining the above with (57), we have that

$$\begin{aligned} & \left| \frac{b_i^{(m)}(\frac{jL^{(m)}}{m} + \delta) - b_i^{(m)}(\frac{jL^{(m)}}{m})}{\delta} - \sum_{s \in S} p(s)\pi^*(s)s_i \right| \\ & \leq \frac{\epsilon}{2} + \left| \sum_{s \in S} p(s)[\pi'(s) - \pi^*(s)]s_i \right| \leq \frac{\epsilon}{2} + 2^n \frac{\epsilon}{2^{n+1}} = \epsilon \end{aligned}$$

for all $m > M := \max\{M_1, M_2\}$. Thus, the first part of Lemma 2 is proved. The second part can be proved in a similar manner, we omit the details here to save space.

APPENDIX H PROOF OF THEOREM 5

By Lemma 2, the scaled trajectories $b_R^{(m)}(t)$ and $b_i^{(m)}(t)$, $i \in I$ become arbitrarily close to piecewise linear for sufficiently large values of m . Specifically, during the j th review period under fluid scaling, they achieve rates that are arbitrarily close to the optimal rates for (P-WBP) with the weights defined in (40). Thus, it suffices to derive bounds for such optimal rates to (P-WBP); the bounds can be applied to the trajectories under WBP-DR under fluid scaling. Let policy $\pi^*(\cdot)$ denote the solution to (P-WBP) with the weights defined in (40). Define $a_i, i \in I$ as follows:

$$a_i := \frac{\sum_{s \in S} p(s)\pi^*(s)s_i}{\sum_{s \in S} p(s)\pi^*(s)} \leq 1. \quad (58)$$

We have that

$$\begin{aligned} & \sum_{s \in S} p(s)\pi^*(s) \\ & = \sum_{s \in S: s_i=1} p(s)\pi^*(s) + \sum_{s \in S: s_i=0} p(s)\pi^*(s) \end{aligned} \quad (59)$$

$$\leq \sum_{s \in S: s_i=1} p(s)\pi^*(s) + \sum_{s \in S: s_i=1} \frac{1-p_i}{p_i} p(s)\pi^*(s) \quad (60)$$

$$= \sum_{s \in S: s_i=1} \frac{1}{p_i} p(s)\pi^*(s). \quad (61)$$

We can derive the inequality (60) as follows. For any channel state $s \in S$ such that $s_i = 0$, there exists a channel state $\sigma \in S$ such that

$$\sigma_i = 1 \text{ and } \sigma_j = s_j, \quad \forall j \in I, j \neq i.$$

Clearly, we have that $\sigma \succ s$. Thus, from the definition of the WBP rule, we have that

$$\pi^*(\sigma) \geq \pi^*(s). \quad (62)$$

Also we have that

$$\frac{p(s)}{p(\sigma)} = \frac{\prod_{j \in I} p_j^{s_j} (1-p_j)^{1-s_j}}{\prod_{j \in I} p_j^{\sigma_j} (1-p_j)^{1-\sigma_j}} = \frac{1-p_i}{p_i}. \quad (63)$$

By applying (62) and (63) to the second summation of (59), we obtain the inequality (60). Combining (58) and (61), we conclude that

$$1 \geq a_i \geq \frac{\sum_{s \in S: s_i=1} p(s) \pi^*(s)}{\sum_{s \in S: s_i=1} \frac{1}{p_i} p(s) \pi^*(s)} = p_i, \quad \forall i \in I.$$

According to (31), we have that

$$1 = \sum_{s \in S} p(s) \pi^*(s) [1 + s_{ib}] = (1 + a_{ib}) \sum_{s \in S} p(s) \pi^*(s).$$

Thus, for all $i \in I$, we have that

$$r_i = \sum_{s \in S} p(s) \pi^*(s) s_i = a_i \sum_{s \in S} p(s) \pi^*(s) = \frac{a_i}{1 + a_{ib}} \geq \frac{p_{ib}}{2}.$$

From (31), or equivalently (28), we have that

$$r_{in} = r_{ib} \geq \frac{p_{ib}}{2}.$$

According to (21), the innovative fluid rate achieved by each user under the solution to **(P-WBP)** in the j th review period is bounded below by $p_{ib}/2$ for any $j \in \mathbb{Z}_+$. Therefore, the asymptotic decoding delay under WBP-DR policy is bounded above by $2K/p_{ib}$. Because the optimal innovative fluid rate achieved by user i^b is no more than p_{ib} , the optimal decoding delay is bounded below by K/p_{ib} . Thus, the theorem is proved.

APPENDIX I PROOF OF THEOREM 6

Without loss of generality, we assume $p_1 \leq p_2$, i.e., $i^b = 1$. By Lemma 2, it suffices to examine the optimal rates for **(P-WBP)** to characterize the users' trajectories under WBP-DR. First, one can easily verify that the fluid rates as the solution to **(P-WBP)** depend only on the *relative* order of the priority among users. Thus, we introduce the following notation. Let $\pi^{(j)}(\cdot)$ denote the solution to **(P-WBP)** when user j receives the higher priority. Additionally, let $r_i^{(j)}$, $\hat{r}_i^{(j)}$, and $r_{in}^{(j)}$ denote the associated rates with policy $\pi^{(j)}(\cdot)$. We consider the following cases:

Case 1: $p_1 p_2 \geq \frac{1}{2}$. We have that, by solving **(P-WBP)**,

$$\begin{aligned} \pi^{(j)}((11)) &= \frac{1}{2p_1 p_2}, \quad \pi^{(j)}(s) = 0, \quad \forall s \neq (11); \\ r_1^{(j)} &= r_2^{(j)} = r_{in}^{(j)} = \frac{1}{2}, \quad \forall j = 1, 2. \end{aligned}$$

Based on (21), both users achieve the same innovative fluid rate $\frac{1}{2}$, irrespective of the priority among the users.

Case 2: $p_1 p_2 < \frac{1}{2}$ and $1 - 2p_1 \geq p_1(1 - p_2)$. We have that

$$\begin{aligned} \pi^{(1)}((11)) &= \pi^{(1)}((10)) = 1, \quad \pi^{(1)}((01)) \geq \frac{p_1(1 - p_2)}{p_2(1 - p_1)}; \\ r_{in}^{(1)} &= r_1^{(1)} = p_1 \leq r_2^{(1)}. \end{aligned}$$

In this case, User 1 always receives the higher priority, and both users achieve the same innovative fluid rate p_1 .

Case 3: $p_1 p_2 < \frac{1}{2}$ and $1 - 2p_1 < p_1(1 - p_2)$. We have that

$$\begin{aligned} \pi^{(1)}((11)) &= 1, \quad \frac{\pi^{(1)}((10))}{\pi^{(1)}((01))} > \frac{p_2(1 - p_1)}{p_1(1 - p_2)}, \quad \pi^{(1)}((00)) = 0; \\ r_{in}^{(1)} &= r_1^{(1)} > r_2^{(1)}. \end{aligned}$$

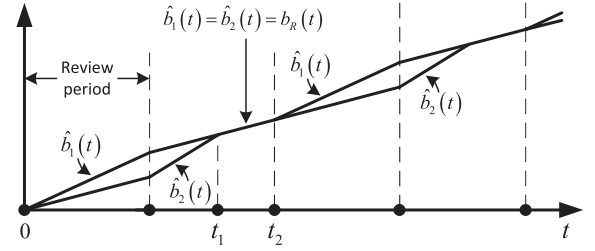


Fig. 6. Example of trajectories for Case 3.

We also have that $r_1^{(2)} \leq r_2^{(2)}$ since $p_1 \leq p_2$. Thus, the higher-priority user achieves a higher fluid rate in this case. As an example of Case 3, Fig. 6 shows the trajectories of innovative fluids of User 1 and 2 with $p_1 = 0.5$ and $p_2 = 0.6$. User 1 receives the higher priority at the first review period. At the second review period, the higher priority is assigned to User 2 whereas $\hat{b}_2(t)$ overtakes $\hat{b}_1(t)$ at $t = t_1$. Due to (31), we have that $b_R(t) = \hat{b}_1(t)$ for all $t \geq 0$. Hence, during $(t_1, t_2]$, $\hat{r}_2(t)$ is reduced from $r_2^{(2)}$ to $r_1^{(2)}$, or equivalently $r_{in}^{(2)}$, since $\hat{b}_2(t)$ cannot exceed $b_R(t)$ for all $t \geq 0$. Since $\hat{b}_1(t) = \hat{b}_2(t)$ at $t = t_2$, the trajectories during 3rd and 4th review periods will evolve in the same manner as in the first two review periods; such trajectory patterns repeat over all the subsequent review periods. In general, it takes $\lceil \kappa \rceil + 1$ review periods from $t = 0$ for $\hat{b}_2(t)$ to overtake $\hat{b}_1(t)$, where κ is given by

$$\kappa = \frac{r_1^{(1)} - r_2^{(1)}}{r_2^{(2)} - r_1^{(2)}} > 0. \quad (64)$$

Let c denote the mean innovative fluid rate achieved by both users under WBP-DR policy in the fluid limit. We have that

$$c = \frac{\hat{r}_1^{(1)} + \lceil \kappa \rceil \hat{r}_1^{(2)}}{1 + \lceil \kappa \rceil} = \frac{r_1^{(1)} + \lceil \kappa \rceil r_1^{(2)}}{1 + \lceil \kappa \rceil}. \quad (65)$$

Next, we compute the optimal rates by directly solving **(P)**. Denote the solution to **(P)** by $\pi^*(\cdot)$, and the associated rates by r_i^* , r_{in}^* , and \hat{r}_i^* . Similar to **(P-WBP)**, we consider three cases:

Case 1: $p_1 p_2 \geq \frac{1}{2}$ and *Case 2:* $p_1 p_2 < \frac{1}{2}$ and $1 - 2p_1 \geq p_1(1 - p_2)$. For both *Case 1* and *2*, one can show that users achieve the same innovative fluid rates as in **(P-WBP)**.

Case 3: $p_1 p_2 < \frac{1}{2}$ and $1 - 2p_1 < p_1(1 - p_2)$. We have that

$$\begin{aligned} \pi^*((11)) &= 1, \quad \frac{\pi^*((10))}{\pi^*((01))} = \frac{p_2(1 - p_1)}{p_1(1 - p_2)}, \quad \pi^*((00)) = 0, \\ r_{in}^* &= r_1^* = r_2^* < p_1. \end{aligned}$$

Thus, both users achieve the same innovative fluid rate given by r_1^* . Recall that $\pi^{(j)}((11)) = 1$ and $\pi^{(j)}((00)) = 0$ for $j = 1, 2$ in this case. Because $\pi^*(\cdot)$, $\pi^{(1)}(\cdot)$, and $\pi^{(2)}(\cdot)$ satisfy (31), $\pi^*(\cdot)$ can be represented as a convex combination of $\pi^{(1)}(\cdot)$ and $\pi^{(2)}(\cdot)$, i.e., there exist $\alpha_1, \alpha_2 \in [0, 1]$ such that

$$\alpha_1 + \alpha_2 = 1; \quad \pi^*(s) = \alpha_1 \pi^{(1)}(s) + \alpha_2 \pi^{(2)}(s), \quad \forall s \in S.$$

Therefore, we have that

$$r_i^* = \alpha_1 r_i^{(1)} + \alpha_2 r_i^{(2)}, \quad \forall i = 1, 2.$$

Since $r_1^* = r_2^*$ and based on (64), one can show that

$$\alpha_1 = \frac{1}{1+\kappa}, \quad \alpha_2 = \frac{\kappa}{1+\kappa}, \quad r_1^* = r_2^* = \frac{r_1^{(1)} + \kappa r_1^{(2)}}{1+\kappa}.$$

Combining the above with (65), we have that

$$\frac{c}{r_1^*} = \frac{r_1^{(1)} + [\kappa]r_1^{(2)}}{r_1^{(1)} + \kappa r_1^{(2)}} \cdot \frac{1+\kappa}{1+[\kappa]} = \frac{\theta + [\kappa]}{\theta + \kappa} \cdot \frac{1+\kappa}{1+[\kappa]} \quad (66)$$

where $\theta := r_1^{(1)}/r_1^{(2)} > 1$. From Theorem 5, we have that $r_1^{(2)} \geq p_1/2$. Because $r_1^{(1)} \leq p_1$, we conclude that $\theta \leq 2$.

Finally, we derive a lower bound for (66). Define $x = [\kappa] \in \mathbb{Z}_+$. For fixed $\theta \in (1, 2]$, (66) satisfies

$$\frac{\theta + x}{\theta + \kappa} \cdot \frac{1+\kappa}{1+x} > \frac{\theta + x}{\theta + x - 1} \cdot \frac{x}{1+x}, \quad (67)$$

because the LHS of (67) is increasing in $\kappa \in (x-1, x]$. One can show that the RHS of (67) is increasing in $x \in \mathbb{Z}_+$. Thus, for fixed $\theta \in (1, 2]$, (66) is bounded below by $(\theta+1)/2\theta$. Because $(\theta+1)/2\theta \geq 3/4$ for $\theta \in (1, 2]$, the theorem is proved.

REFERENCES

- [1] A. Damnjanovic *et al.*, "A survey on 3GPP heterogeneous networks," *IEEE Wireless Commun.*, vol. 18, no. 3, pp. 10–21, Jun. 2011.
- [2] S. W. Peters, A. Y. Panah, K. T. Truong, and R. W. Heath, Jr., "Relay architectures for 3GPP LTE-advanced," *EURASIP J. Wireless Commun. Netw.*, vol. 2009, p. 618787, Jul. 2009.
- [3] D. Lecompte and F. Gabin, "Evolved multimedia broadcast/multicast service (eMBMS) in LTE-advanced: Overview and Rel-11 enhancements," *IEEE Commun. Mag.*, vol. 50, no. 11, pp. 68–74, Nov. 2012.
- [4] D. Gunduz, O. Simeone, A. J. Goldsmith, H. V. Poor, and S. Shamai (Shitz), "Multiple multicasts with the help of a relay," *IEEE Trans. Inf. Theory*, vol. 56, no. 12, pp. 6142–6158, Dec. 2010.
- [5] M. Cardone, D. Tuninetti, R. Knopp, and U. Salim, "Gaussian half-duplex relay networks: Improved constant gap and connections with the assignment problem," *IEEE Trans. Inf. Theory*, vol. 60, no. 6, pp. 3559–3575, Jun. 2014.
- [6] J. Du, M. Xiao, M. Skoglund, and M. Médard, "Wireless multicast relay networks with limited-rate source-conferencing," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 8, pp. 1390–1401, Aug. 2013.
- [7] K. Loa *et al.*, "TMT-advanced relay standards [WiMAX/LTE update]," *IEEE Commun. Mag.*, vol. 48, no. 8, pp. 40–48, Aug. 2010.
- [8] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Phys. Commun.*, vol. 1, no. 1, pp. 3–20, 2008.
- [9] B. T. Swapna, A. Eryilmaz, and N. B. Shroff, "Throughput-delay analysis of random linear network coding for wireless broadcasting," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6328–6341, Oct. 2013.
- [10] Y. Yang and N. Shroff, "Throughput of rateless codes over broadcast erasure channels," in *Proc. ACM Mobihoc*, 2012, pp. 125–134.
- [11] A. Eryilmaz, A. Ozdaglar, M. Médard, and E. Ahmed, "On the delay and throughput gains of coding in unreliable networks," *IEEE Trans. Inf. Theory*, vol. 54, no. 12, pp. 5511–5524, Dec. 2008.
- [12] B. Shrader and A. Ephremides, "Queueing delay analysis for multicast with random linear coding," *IEEE Trans. Inf. Theory*, vol. 58, no. 1, pp. 421–429, Jan. 2012.
- [13] R. Cogill, B. Shrader, and A. Ephremides, "Stable throughput for multicast with random linear coding," *IEEE Trans. Inf. Theory*, vol. 57, no. 1, pp. 267–281, Jan. 2011.
- [14] D. S. Lun, P. Pakzad, C. Fragouli, M. Médard, and R. Koetter, "An analysis of finite-memory random linear coding on packet streams," in *Proc. IEEE WiOpt*, Apr. 2006, pp. 1–6.
- [15] H. Khamfroush, D. E. Lucani, and J. Barros, "Network coding for wireless cooperative networks: Simple rules, near-optimal delay," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Jun. 2014, pp. 255–260.
- [16] K. Sundaresan and S. Rangarajan, "Cooperation versus multiplexing: Multicast scheduling algorithms for OFDMA relay networks," *IEEE/ACM Trans. Netw.*, vol. 22, no. 3, pp. 756–769, Jun. 2014.
- [17] D. J. C. MacKay, "Fountain codes," *IEEE Proc.-Commun.*, vol. 152, no. 6, pp. 1062–1068, Dec. 2005.
- [18] R. H. Clarke, "A statistical theory of mobile-radio reception," *Bell Syst. Tech. J.*, vol. 47, no. 6, pp. 957–1000, Jul. 1968.
- [19] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.
- [20] H. S. Wang and N. Moayeri, "Finite-state Markov channel—A useful model for radio communication channels," *IEEE Trans. Veh. Technol.*, vol. 44, no. 1, pp. 163–171, Feb. 1995.
- [21] Q. Zhang and S. A. Kassam, "Finite-state Markov model for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 47, no. 11, pp. 1688–1692, Nov. 1999.
- [22] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, and P. Whiting, "Scheduling in a queueing system with asynchronously varying service rates," *Probab. Eng. Inf. Sci.*, vol. 18, no. 2, pp. 191–217, 2004.
- [23] K. Avrachenkov, A. Piunovskiy, and Y. Zhang, "Asymptotic fluid optimality and efficiency of the tracking policy for bandwidth-sharing networks," *J. Appl. Probab.*, vol. 48, no. 1, pp. 90–113, 2011.
- [24] N. Bäuerle, "Optimal control of queueing networks: An approach via fluid models," *Adv. Appl. Probab.*, vol. 34, no. 2, pp. 313–328, 2002.
- [25] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, vol. 29. San Francisco, CA, USA: Freeman, 2002.
- [26] C. Maglaras, "Discrete-review policies for scheduling stochastic networks: Trajectory tracking and fluid-scale asymptotic optimality," *Ann. Appl. Probab.*, vol. 10, no. 3, pp. 897–929, 2000.
- [27] A. L. Stolyar, "MaxWeight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic," *Ann. Appl. Probab.*, vol. 14, no. 1, pp. 1–53, 2004.
- [28] S. Meyn, *Control Techniques for Complex Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2008.
- [29] K. E. Baddour and N. C. Beaulieu, "Autoregressive modeling for fading channel simulation," *IEEE Trans. Wireless Commun.*, vol. 4, no. 4, pp. 1650–1662, Jul. 2005.
- [30] T. Stockhammer, "Dynamic adaptive streaming over HTTP: Standards and design principles," in *Proc. 2nd Annu. ACM Conf. Multimedia Syst.*, 2011, pp. 133–144.
- [31] R. T. Rockafellar, *Convex Analysis*. Princeton, NJ, USA: Princeton Univ. Press, 1997.

Chao Chen received his B.S. degree in Communication Engineering from Harbin institute of technology, China in 2009, and Ph.D. degree in Electrical and Computer Engineering from Korea University, South Korea in 2015. Since December 2005, he is with the School of Information and Electronic Engineering at Zhejiang Gongshang University, China, where he currently is a Lecturer. His research interest includes network coding and wireless systems.

Seung Jun Baek (S'04–M'09) received the B.S. degree from Seoul National University in 1998, and the M.S. and Ph.D. degrees from the University of Texas at Austin in 2002 and 2007, respectively, all in electrical and computer engineering. From 2007 to 2009, he was a Member of Technical Staff with the Communications Systems Lab, DSPS R&D Center, Texas Instruments, Dallas, TX. In 2009, he joined the College of Information and Communications, Korea University, Seoul, Korea, where he is currently an associate professor. His research interests include modeling and analysis of communication networks; cellular networks, wireless ad-hoc networks, sensor networks, game theory and compressive sensing.

Gustavo de Veciana (S'88–M'94–SM'01–F'09) received his B.S., M.S. and Ph.D. in electrical engineering from the University of California at Berkeley in 1987, 1990, and 1993 respectively, and joined the Department of Electrical and Computer Engineering where he is currently a Cullen Trust Professor of Engineering. He served as the Director and Associate Director of the Wireless Networking and Communications Group (WNCG) at the University of Texas at Austin, from 2003–2007. His research focuses on the analysis and design of communication and computing networks; data-driven decision-making in man-machine systems, and applied probability and queueing theory. Dr. de Veciana served as editor and is currently serving as editor-at-large for the IEEE TRANSACTIONS ON NETWORKING/ACM Transactions on Networking. He was the recipient of a National Science Foundation CAREER Award 1996 and a co-recipient of five best paper awards including: IEEE William McCalla Best ICCAD Paper Award for 2000, Best Paper in ACM TODAES Jan 2002–2004, Best Paper in ITC 2010, Best Paper in ACM MSWIM 2010, and Best Paper IEEE INFOCOM 2014. In 2009 he was designated IEEE Fellow for his contributions to the analysis and design of communication networks. He currently serves on the board of trustees of IMDEA Networks Madrid.