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 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 mobile 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 are 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 mobile users. In cellular networks, the two-hop relay 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 mobile 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.

C. Chen and S. Baek are with the College of Information and Communications, Korea University, Seoul 136-701, Korea (e-mail: eckio_491@163.com, 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).

The work of S. Baek was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2013R1A1A1062500). The work of G. de Veciana was supported in part by the National Science Foundation under Grant CNS-1343383.

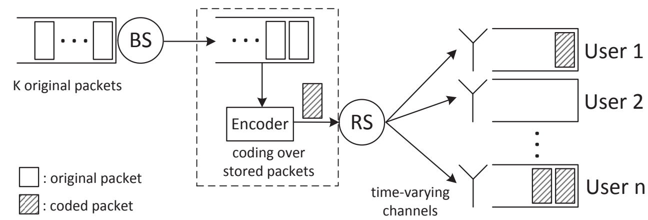

Fig. 1. System model.

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, which is defined as the time 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 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.

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.

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.

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. Section IX concludes the paper.

[1] In 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].

II. System Model and Problem Formulation

We shall use boldface letters to denote vectors and matrices. The ith entry of vector $\boldsymbol{x}$ 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 $I = \{1, ..., n\}$ 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 ν over a finite field $\mathbb{F}_d$ and has a length of $\nu\lceil \log_2 d\rceil$ 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. Due to the time-varying channels, we assume for simplicity that one of two cases may occur: a given user either successfully receives the packet or not. Let us define a binary random process $X_i(l)$ such that $X_i(l) = 1$ if user i would be able to receive the packet in 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 allocates[3] 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 further assume that $X_i(l)$'s are stationary and ergodic discrete-time Markov chains (DTMC) on $\{0, 1\}$, and are independent across users. Let $p_i := \mathbb{P}(X_i(l) = 1)$ denote the steady-state probability that the channel of user i is in the ON state. 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 $\boldsymbol{s} := (s_1, \cdots, s_n) \in \{0, 1\}^n$ 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 $\boldsymbol{s} = (1, 1, 0)$ indicates that the channels of user 1 and 2 are in the ON state. We let $S := \{0, 1\}^n$ denote the set of all possible channel states. Let the random process $\boldsymbol{X}(l) := (X_1(l) \cdots X_n(l)) \in S$ denote the channel state in time slot l. The probability that the channel state is $\boldsymbol{s}$ in the steady state is denoted by $p(\boldsymbol{s})$ where

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

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]

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

Note that (1) 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 (1) is equal to 1 if $B_R(l) > \hat{B}_i(l)$. 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.

Note that once the RS fetches K original packets from the BS, the optimal action is to repeatedly multicast the coded packets. Thus, we will divide the overall transmission policy into two stages:

- Opportunistic Fetch stage: time duration from the beginning of the transmission until the RS fetches 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 time-varying multicast channels.
- Flush stage: time duration following the Opportunistic Fetch stage until all users have received K innovative packets. The RS always performs a multicast in this stage.

We briefly show that our problem can be formulated as an MDP. We define $\mathcal{Q} := \{0, 1, \ldots, K\} \times \{0, 1, \ldots, K\}^n$ as the system state space, which represents the number of

[2]As 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.

[3]By 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.

TABLE I
SUMMARY OF THE NOTATION

General System Parameters	
n	number of users
I	$:= \{1, \ldots, n\}$, 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
$\boldsymbol{s}$	$:= (s_1, \ldots, s_n)$, channel state
S	$:= \{0,1\}^n$, set of channel states
$p(\boldsymbol{s})$	probability that the channel state is $\boldsymbol{s}$
$B_i(l)$	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
$\boldsymbol{w}$	$:= (w_1, \ldots, w_n)$, vector of users' weights
$\rho(\boldsymbol{w}, \boldsymbol{s})$	$:= \sum_{i \in I} s_i w_i$, revenue associated with channel state $\boldsymbol{s}$
$\hat{\rho}(\boldsymbol{w}, \boldsymbol{s})$	$:= \rho(\boldsymbol{w}, \boldsymbol{s})/(1 + s_{i^{\text{b}}})$, normalized revenue associated with channel state $\boldsymbol{s}$
Fluid Model Related Parameters	
$\pi(\boldsymbol{s}, t)$	instantaneous fraction of time spent for multicasting under the condition that the channel state is $\boldsymbol{s}$ 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
$r_{\text{in}}(t)$	inflow rate to the RS at time t

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 $|S| = O(2^n)$. 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 [18]–[20].

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

$$B_R(l) = \sum_{\boldsymbol{s} \in S} H_{\boldsymbol{s}}(l); \quad B_i(l) = \sum_{\boldsymbol{s} \in S} s_i G_{\boldsymbol{s}}(l), \ \forall i \in I.$$

Recall that $s_i = 1$ if in channel state $\boldsymbol{s}$ 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

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

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:

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

Let $\mathbb{D}[0,\infty)$ be the set of functions $f : [0,\infty) \to \mathbb{R}$ that are right-continuous with left limits (RCLL). Then, clearly, $\widetilde{G}_{\boldsymbol{s}}(t)$ and other extended processes defined previously, are random elements in $\mathbb{D}[0,\infty)$. Consider a sequence of systems for the fluid scaling parameter $m = 1, 2, \ldots$. In the mth 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 mth system, e.g., $\widetilde{G}_s^{(m)}(t)$ is the process $\widetilde{G}_{\boldsymbol{s}}(t)$ in the mth system. We denote the rescaling of the random process $\widetilde{Y}^{(m)}(t)$ by $y^{(m)}(t)$ given by

$$y^{(m)}(t) := \frac{1}{m}\widetilde{Y}^{(m)}(mt). \tag{2}$$

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

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

$$g_{\boldsymbol{s}}^{(m)}(t) \to g_{\boldsymbol{s}}(t),$$
$$h_{\boldsymbol{s}}^{(m)}(t) \to h_{\boldsymbol{s}}(t),$$
$$b_R^{(m)}(t) \to b_R(t) = \sum_{\boldsymbol{s} \in S} h_{\boldsymbol{s}}(t),$$
$$b_i^{(m)}(t) \to b_i(t) = \sum_{\boldsymbol{s} \in S} g_{\boldsymbol{s}}(t) s_i, \ \forall i \in I$$

where $g_{\boldsymbol{s}}(t)$, $h_{\boldsymbol{s}}(t)$, $b_R(t)$, and $b_i(t)$ are Lipschitz continuous on $[0,\infty)$.

Proof: The proof is similar to Lemma 1 in [18]. ∎

Hence, $g_{\boldsymbol{s}}(t)$, $h_{\boldsymbol{s}}(t)$, $b_R(t)$, 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 [18]. 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

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

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

$$\frac{G_{\boldsymbol{s}}(m(t+\delta)) - G_{\boldsymbol{s}}(mt)}{m\delta} \tag{3}$$

$$= \frac{\sum_{j=\lfloor mt \rfloor + 1}^{\lfloor m(t+\delta) \rfloor} \mathbf{1}(\boldsymbol{X}(j) = \boldsymbol{s})}{m\delta} \cdot \frac{\sum_{j=\lfloor mt \rfloor + 1}^{\lfloor m(t+\delta) \rfloor} \mathbf{1}(\boldsymbol{X}(j) = \boldsymbol{s}, \Psi(j) = 1)}{\sum_{j=\lfloor mt \rfloor + 1}^{\lfloor m(t+\delta) \rfloor} \mathbf{1}(\boldsymbol{X}(j) = \boldsymbol{s})}. \tag{4}$$

The limit of the factor on the RHS of (4) as $m \to \infty$ is denoted by $\pi(\boldsymbol{s}, t) \in [0,1]$, which can be interpreted as the instantaneous fraction of time spent for multicasting, on the condition that the channel state is $\boldsymbol{s}$ at time t. Additionally, as $m \to \infty$, the factor on the LHS of (4) converges to $p(\boldsymbol{s})$, and (3) converges to $\dot{g}_{\boldsymbol{s}}(t)$ because t is regular. Thus, we have that

$\dot{g}_{\boldsymbol{s}}(t) = p(\boldsymbol{s})\pi(\boldsymbol{s},t)$. By taking similar steps for the process $H_{\boldsymbol{s}}(t)$, we get $\dot{h}_{\boldsymbol{s}}(t) = p(\boldsymbol{s})[1-\pi(\boldsymbol{s},t)]$.

Let $r_i(t) := \dot{b}_i(t)$ and $r_{\text{in}}(t) := \dot{b}_R(t)$ 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

$$\hat{r}_i(t) = \begin{cases} r_i(t), & \text{if } b_R(t) > \hat{b}_i(t), \\ \min[r_i(t), r_{\text{in}}(t)], & \text{if } b_R(t) = \hat{b}_i(t). \end{cases} \quad (5)$$

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 $\min[r_i(t), r_{\text{in}}(t)]$ 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 mth system; thus, under the fluid scaling (2), we want all of the scaled trajectories $\hat{b}_i^{(m)}(t), i \in I$ to reach K. We define the *asymptotic* decoding delay in the fluid regime as

$$T = \inf_t \left\{ t \in \mathbb{R}_+ \middle| \min_{i\in I}\{\hat{b}_i(t)\} = K \right\}. \quad (6)$$

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:

$$\textbf{(F)} \quad \underset{\pi}{\text{minimize}} \quad T = \inf_t \left\{ t \in \mathbb{R}_+ \middle| \min_{i\in I}\{\hat{b}_i(t)\} = K \right\}$$

$$\text{subject to} \quad b_i(0) = \hat{b}_i(0) = 0, \ \forall i \in I, \quad (7)$$

$$b_i(t) = b_i(0) + \int_0^t r_i(\tau)d\tau, \ \forall i \in I, \quad (8)$$

$$\hat{b}_i(t) = \hat{b}_i(0) + \int_0^t \hat{r}_i(t)d\tau, \ \forall\ i \in I, \quad (9)$$

$$b_R(t) = b_R(0) + \int_0^t r_{\text{in}}(\tau)d\tau, \ b_R(0) = 0, \quad (10)$$

$$r_i(t) = \sum_{\boldsymbol{s}\in S} p(\boldsymbol{s})\pi(\boldsymbol{s},t)s_i, \forall i \in I, \quad (11)$$

$$r_{\text{in}}(t) = \sum_{\boldsymbol{s}\in S} p(\boldsymbol{s})[1-\pi(\boldsymbol{s},t)], \quad (12)$$

$$\pi(\boldsymbol{s},t) \in [0,1], \ \forall \boldsymbol{s} \in S, \quad (13)$$

$$\pi(\boldsymbol{s},t) = 1, \ \text{if } b_R(t) = K. \quad (14)$$

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. 2 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.

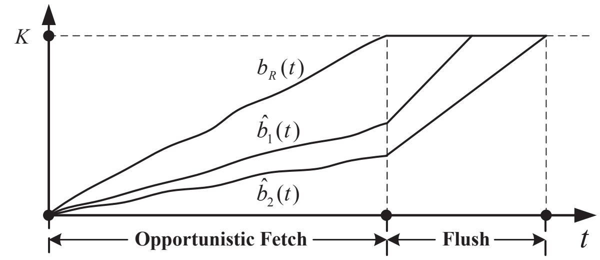

Fig. 2. An example of buffer trajectories with $n = 2$.

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 $\pi(\boldsymbol{s},t)$, $r_{\text{in}}(t)$, $r_i(t)$, and $\hat{r}_i(t)$, and instead, use $\pi(\boldsymbol{s})$, r_{in}, r_i, and $\hat{r}_i$. Based on (5), we have that

$$\hat{r}_i = \min\{r_i, r_{\text{in}}\}, \ \forall i \in I. \quad (15)$$

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

$$K - \frac{\hat{r}_i}{r_{\text{in}}}K = \frac{[r_{\text{in}} - r_i]^+}{r_{\text{in}}}K. \quad (16)$$

It follows that, the decoding delay (6) is given by

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

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

$$\underset{\pi}{\text{minimize}} \quad \frac{K}{r_{\text{in}}} + \max_i \left\{ \frac{[r_{\text{in}} - r_i]^+}{r_{\text{in}}} \frac{K}{p_i} \right\} \quad (17)$$

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

$$r_i = \sum_{\boldsymbol{s}\in S} p(\boldsymbol{s})\pi(\boldsymbol{s})s_i, \ \forall i \in I, \quad (19)$$

$$\pi(\boldsymbol{s}) \in [0,1], \ \forall \boldsymbol{s} \in S. \quad (20)$$

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 $i^{\text{b}} \in \arg\min_{i\in I}\{p_i\}$ 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:

$$\textbf{(P)} \quad \underset{\pi}{\text{maximize}} \quad \min_{i\in I}\{r_i\} \tag{21}$$

$$\text{subject to} \quad r_{\text{in}} = r_{i^{\text{b}}}, \tag{22}$$

$$(18) - (20).$$

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:

$$r_{\text{in}} = \min_{i\in I}\{r_i\}. \tag{23}$$

If we combine (23) and the definition of $\hat{r}_i$ in (15), we have that

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

That is, all users achieve the same innovative fluid rate given by r_{in} under (23). 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 (17) is resolved, because the second term of (17) becomes 0 under constraint (23), which results in the one-stage LP given by **(P)**. In the proof of Theorem 2, we also observe that $r_{i^{\text{b}}} = \min_{i\in I}\{r_i\}$ holds under the optimal policy in addition to (23). 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:

$$\textbf{(MW)} \quad \underset{\pi}{\text{maximize}} \quad \sum_{i\in I} w_i r_i$$

$$\text{subject to} \quad \sum_{\boldsymbol{s}\in S} p(\boldsymbol{s})\pi(\boldsymbol{s})[1 + s_{i^{\text{b}}}] = 1, \tag{24}$$

$$(18) - (20),$$

where (24) is derived from (22) by expressing r_{in} and $r_{i^{\text{b}}}$ 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

$$\sum_{\boldsymbol{s}\in S} p(\boldsymbol{s})\pi(\boldsymbol{s})\rho(\boldsymbol{w}, \boldsymbol{s}) \tag{25}$$

Define $\boldsymbol{w} := (w_1, \ldots, w_n)$ as the weight vector, and $\rho(\boldsymbol{w}, \boldsymbol{s}) := \sum_{i\in I} w_i s_i$ as the revenue earned by the system when the RS multicasts under channel state $\boldsymbol{s}$. Therefore, (25) is the expected revenue earned by the system under policy $\pi(\cdot)$. We will also define $\hat{\rho}(\boldsymbol{w}, \boldsymbol{s}) := \rho(\boldsymbol{w}, \boldsymbol{s})/(1 + s_{i^{\text{b}}})$ as the "normalized" revenue. By definition, $\hat{\rho}(\boldsymbol{w}, \boldsymbol{s})$ penalizes $\rho(\boldsymbol{w}, \boldsymbol{s})$ by a factor of 2 if the bottleneck user's channel is ON in channel state $\boldsymbol{s}$, i.e., $s_{i^{\text{b}}} = 1$. The penalty arises because there exists a constraint which involves the bottleneck user, i.e., (24) 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 $\boldsymbol{w}$, let $\boldsymbol{\eta}(k) := (\eta_1(k), \ldots, \eta_n(k)) \in S$ denote the channel state such that $\hat{\rho}(\boldsymbol{w}, \boldsymbol{\eta}(k))$ is the kth largest (ties are arbitrarily broken) among all $\{\hat{\rho}(\boldsymbol{w}, \boldsymbol{s}) | \boldsymbol{s} \in S\}$. Additionally, define k^* as

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

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

$$\pi^*(\boldsymbol{s}) = \begin{cases} 1, & \hat{\rho}(\boldsymbol{w}, \boldsymbol{s}) > \xi, \\ \beta, & \hat{\rho}(\boldsymbol{w}, \boldsymbol{s}) = \xi, \\ 0, & \text{otherwise}, \end{cases} \tag{26}$$

where

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

Proof: The proof is given in Appendix F. ■

Remark: By definition, the fluid policy $\pi^*(\boldsymbol{s})$ represents the time fraction of multicasting under channel state $\boldsymbol{s}$. In the original stochastic network, $\pi^*(\boldsymbol{s})$ (resp. $1 - \pi^*(\boldsymbol{s})$) is analogous to the *probability* of multicasting (resp. fetching) at a time slot, under the condition that the channel state is $\boldsymbol{s}$. 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 ξ. 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 $\boldsymbol{s}, \boldsymbol{\sigma} \in S$, we may say that $\boldsymbol{s}$ *precedes* $\boldsymbol{\sigma}$ in terms of channel quality if $\hat{\rho}(\boldsymbol{w}, \boldsymbol{s}) > \hat{\rho}(\boldsymbol{w}, \boldsymbol{\sigma})$. The RS multicasts w.p. 1 under $\pi^*(\cdot)$ only if the current channel state precedes ξ. 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 ξ for **(MW)** requires the knowledge of $\boldsymbol{\eta}(k)$, which is the channel state with the kth largest normalized revenue. We briefly argue that determining $\boldsymbol{\eta}(k)$ for given $\boldsymbol{w}$ and k is related to an NP-hard problem called the kTH LARGEST SUBSET [21], 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 kTH LARGEST SUBSET as follows. Let us define the multiset A such that $A := \{w_1, \ldots, w_n\}$. For channel state $\boldsymbol{s} = (s_1, \ldots, s_n)$, define multiset $B_{\boldsymbol{s}} := \{w_i | s_i = 1,\ i \in I\}$. $B_{\boldsymbol{s}}$ is a subset of A, and we have that $\|B_{\boldsymbol{s}}\| = \sum_{i\in I} w_i s_i = \rho(\boldsymbol{w}, \boldsymbol{s})$. Thus, the set of revenues $\{\rho(\boldsymbol{w}, s)\}$ 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 [22] 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 $\boldsymbol{x} = (x_1, \ldots, x_k)$, we define $\langle x_1 \cdots x_k \rangle$ as the k-bit nonnegative binary number associated with $\boldsymbol{x}$ 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 $\boldsymbol{x} = (x_1, \ldots, x_k)$ and $\boldsymbol{y} = (y_1, \ldots, y_k)$ of length $k \in \mathbb{Z}_+$, we define $\boldsymbol{x} \succ \boldsymbol{y}$ 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 $\xi \in \mathbb{R}$, $\boldsymbol{x} \succ \xi$ 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 $w_1 \geq \cdots \geq w_n$, i.e., that the users are indexed in descending order of weight. Consider two channel states $\boldsymbol{s}, \boldsymbol{\sigma} \in S$ such that $\boldsymbol{s} \neq \boldsymbol{\sigma}$. WBP is a precedence rule such that, we say that $\boldsymbol{s}$ precedes $\boldsymbol{\sigma}$ if $\boldsymbol{s} \succ \boldsymbol{\sigma}$. 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 $\boldsymbol{s} = (1, 0, 0)$ and $\boldsymbol{\sigma} = (0, 1, 1)$. Since $\langle 100 \rangle = 4$ is greater than $\langle 011 \rangle = 3$, we say $\boldsymbol{s}$ is a better channel state than $\boldsymbol{\sigma}$ under the WBP rule. Recall that, in the binary number $\langle s_1 \ldots s_n \rangle$ associated with channel state $\boldsymbol{s}$, the ith bit s_i represents the channel state of user i whose weight is w_i. Because $w_1 \geq \cdots \geq w_n$, 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 [23]. 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:

$$\textbf{(P-WBP)} \quad \underset{\pi}{\text{maximize}} \ \sum_{i\in I} w_i r_i$$
$$\text{subject to} \ \ (18) - (19), (24)$$
$$\pi(\boldsymbol{s}) = \begin{cases} 1, & \boldsymbol{s} \succ \xi, \\ \beta, & \boldsymbol{s} = \xi, \\ 0, & \boldsymbol{s} \prec \xi, \end{cases} \tag{27}$$
$$\xi \in \{0, \ldots, 2^n - 1\},\ \beta \in [0, 1). \tag{28}$$

(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 (24), the optimal ξ is the maximum ξ such that

$$\sum_{\boldsymbol{s}\in S, \boldsymbol{s}\geq\xi} p(\boldsymbol{s})[1 + s_{i^\flat}] \geq 1. \tag{29}$$

The LHS of (29), 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 (29), (ii) compare the result with 1, and (iii) adjust ξ accordingly to make (29) 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 [22] is a general method for translating a deterministic fluid

Algorithm 1 Finding the optimal ξ and β

$\xi \leftarrow 2^{n-1}$, Residual $\leftarrow 1$, SumProb $\leftarrow p_1(1+p_{i^{\rm b}})$.
if $i^{\rm b}=1$ **then**
  SumProb $\leftarrow 2p_1$.
end if
for $j=1$ to $n-1$ **do**
  if SumProb $\leq$ Residual **then**
    Residual $\leftarrow$ Residual $-$ SumProb, $\xi \leftarrow \xi - 2^{n-j-1}$.
    if $j=i^{\rm b}-1$ **then**
      SumProb $\leftarrow \frac{2(1-p_j)p_{j+1}}{p_j(1+p_{j+1})}$SumProb.
    else if $j=i^{\rm b}$ **then**
      SumProb $\leftarrow \frac{(1-p_j)p_{j+1}}{2p_j}$SumProb.
    else
      SumProb $\leftarrow \frac{(1-p_j)p_{j+1}}{p_j}$SumProb.
    end if
  else
    $\xi \leftarrow \xi + 2^{n-j-1}$.
    if $j=i^{\rm b}-1$ **then**
      SumProb $\leftarrow \frac{2p_{j+1}}{1+p_{j+1}}$SumProb.
    else
      SumProb $\leftarrow p_{j+1}$SumProb.
    end if
  end if
end for
if SumProb $\leq$ Residual **then**
  Residual $\leftarrow$ Residual $-$ SumProb, $\xi \leftarrow \xi - 1$.
  if $i^{\rm b}=n$ **then**
    SumProb $\leftarrow \frac{1-p_n}{2p_n}$SumProb.
  else
    SumProb $\leftarrow \frac{1-p_n}{p_n}$SumProb.
  end if
end if
$\beta \leftarrow$ Residual/SumProb.

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* [24]. 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 $l_j=jL, j=0,1,\ldots$. At the beginning of jth review period, the RS queue length $B_R(l_j)$ and the number of innovative packets received for each user $\{\hat{B}_i(l_j), i\in I\}$ are observed; the following problem, which is similar to **(P-WBP)**, is then constructed and to be solved:

$$\textbf{(P-DR)}\quad \underset{\pi}{\text{maximize}} \sum_{i\in I} w_i r_i$$
$$\text{subject to}\quad (18)-(19),(27)-(28),$$
$$B_R(l_j)-\hat{B}_{i^{\rm b}}(l_j)+L(r_{\rm in}-r_{i^{\rm b}}) \geq u, \tag{30}$$
$$w_i=\left[K-\hat{B}_i(l_j)\right]^+, \forall i\in I. \tag{31}$$

From (31), the weight of each user is set to the remaining number of innovative packets to be received. Constraint (30) 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 (27) and (28) 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 $\{l_j,\ldots,l_j+L-1\}$. $\pi^*(\boldsymbol{s})$ is implemented as the probability to perform multicasting by the RS in a time slot, under the condition that the channel state is $\boldsymbol{s}$. WBP-DR is described in Algorithm 2. Guidelines for setting parameters L and u will be provided in the next section.

Algorithm 2 WBP-DR policy

for Each time slot $l=jL$, $j=0,1,2,\ldots$ **do**
  Observe $\hat{B}_i(l), \forall i\in I$. Set $w_i \leftarrow \left[K-\hat{B}_i(l)\right]^+, \forall i\in I$.
  Sort $w_1,\ldots,w_n$ 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.
  Solve **(P-DR)** using Algorithm 1 by setting $1-\frac{u+\hat{B}_{i^{\rm b}}(l)-B_R(l)}{L}$ as the initial value of "Residual"; obtain threshold parameters ξ and β.
  for each time slot τ in the review period **do**
    Observe the channel state $\boldsymbol{s}$ and RS queue length $B_R(\tau)$.
    if $B_R(\tau)=0$ **then**
      RS performs a fetch.
    else if $0<B_R(\tau)<K$ **then**
      if $\boldsymbol{s}\succ\xi$ **then**
        RS performs a multicast.
      else if $\boldsymbol{s}=\xi$ **then**
        RS performs a multicast (resp. fetch) w.p. β (resp. $1-\beta$).
      else
        RS performs a fetch.
      end if
    else
      RS performs a multicast.
    end if
  end for
end for

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 mth system, where $\alpha \geq 1$ and $f_L(\cdot)$ is a function such that

$$\frac{f_L(x)}{\log(x)} \to \infty \text{ and } \frac{f_L(x)}{x} \to 0, \quad \text{as } x\to\infty. \tag{32}$$

For example, one can set $L^{(m)}=f_L(mK)=\lfloor\sqrt{mK}\rfloor$. (32) implies that the length of a review period tends to infinity in the fluid limit. We have that

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

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

$$w_i = \left[mK - \hat{B}_i(jL^{(m)})\right]^+, \ \forall i \in I. \tag{33}$$

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

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

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

for all $m > M$, where $\pi^*(\boldsymbol{s})$ denotes the solution to **(P-WBP)** with the weights defined in (33).

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 (15) 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.

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 [24] 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 $K = 10^3$. For the channel model, we use a DTMC $X_i(t) \in \{0, 1\}$ with the following transition matrix:

$$\begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix} = \begin{bmatrix} 1 - \lambda_i & \lambda_i \\ \mu_i & 1 - \mu_i \end{bmatrix}$$

for some $\lambda_i, \mu_i \in [0, 1]$. Thus, the steady-state probability that user i's channel is in the ON state, or p_i, is given by $\frac{\lambda_i}{\lambda_i + \mu_i}$. For simplicity, we set $\lambda_i = 0.2$ for all $i \in I$. Thus, we have that $p_i = \frac{1}{1 + 5\mu_i}$. We will change μ_i to control the value of p_i in our simulations. For WBP-DR, we set $L = u = 30$.

Fig. 3(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 1–5% higher than the lower bound. ARQ and GREEDY incur delays that are up to 66% and 31% 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. 3(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. 3(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 $p \leq 0.4$, the delay incurred by ARQ is at least 119% higher than that of WBP-DR.

In Fig. 3(c), we consider symmetric-user systems with fixed $p = 0.5$ and varying n. WBP-DR incurs a decoding delay that is 2–7% higher than the lower bound and reduces decoding delay by up to 50% and 19% 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 $N \sim$ 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

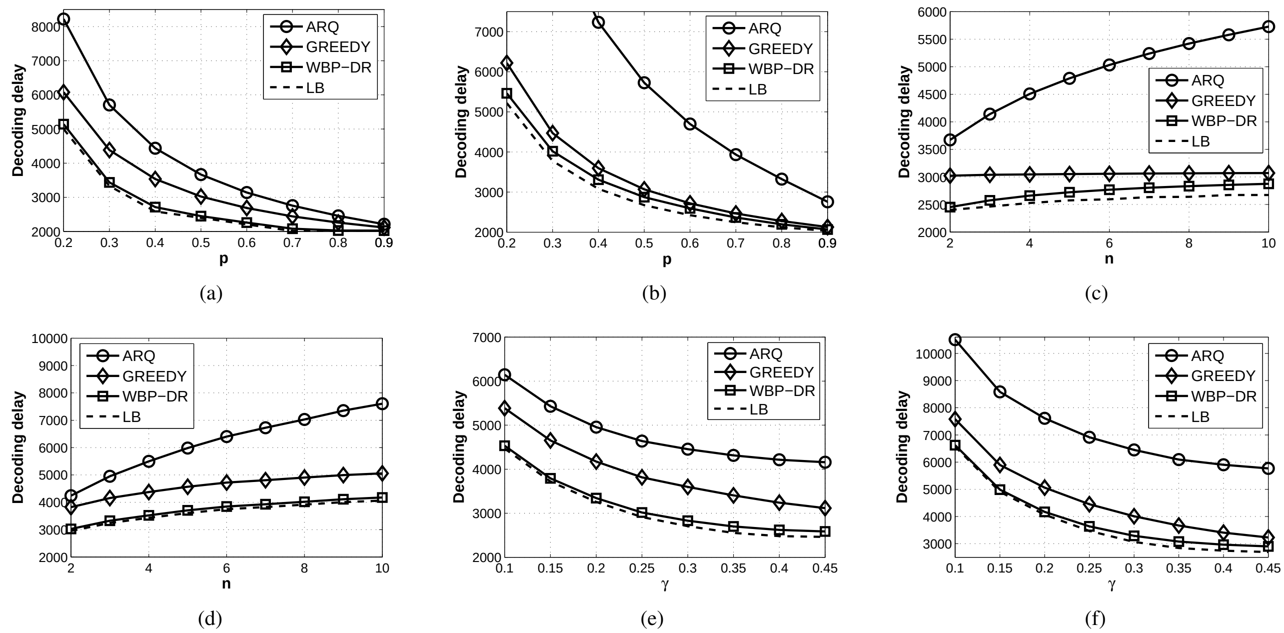

Fig. 3. Comparison of decoding delays based on simulations: (a) symmetric channels with $n = 2$ and varying p; (b) symmetric channels with $n = 10$ and varying p; (c) symmetric channels with $p = 0.5$ and varying n; (d) asymmetric channels with p_i's are uniformly and independently generated from $[0.2, 0.8]$ 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.

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. 3(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 3% of the lower bound. By contrast, ARQ and GREEDY incur delays that are 41–82% and 22–26% 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. 3(c) and Fig. 3(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. 3(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 $p_{i^{\mathrm{b}}}$ 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 12–15% and 33–46% lower than that of GREEDY and ARQ, respectively. Fig. 3(f) considers a similar scenario as Fig. 3(e) with $n = 10$. WBP-DR incurs a lower delay by 12–19%, relative to GREEDY. Additionally, WBP-DR significantly outperforms ARQ, reducing the delay by up to 50%.

IX. 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 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 $\pi^*(\boldsymbol{s},t)$. Let T_1^* and T_2^* be the durations of the Opportunistic Fetch stage and the Flush stage, respectively, under policy $\pi^*(\boldsymbol{s},t)$. It is clear that $\pi^*(\boldsymbol{s},t)=1$ for $T_1^*<t\le T_1^*+T_2^*$. Consider policy $\pi'(\boldsymbol{s},t)$ such that

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

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

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

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

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 $\pi'(\boldsymbol{s},t)$ is exactly T_1^*. It is easy to verify that $\pi'(\boldsymbol{s},t)$ satisfies (13) and (14); therefore, $\pi'(\boldsymbol{s},t)$ is a feasible policy. The fluid rates are constant under policy $\pi'(\boldsymbol{s},t)$ during the Opportunistic Fetch stage, i.e.,

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

Combining the above with (5), 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 $0\le t\le T_1^*$, which further implies that

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

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

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

Therefore, policy $\pi'(\boldsymbol{s},t)$ incurs no higher decoding delay than policy $\pi^*(\boldsymbol{s},t)$, 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

$$r_{\text{in}}=\min_{i\in I}\{r_i\}. \tag{34}$$

The proof of Lemma 3 is provided in Appendix C. Note that, the Flush stage has zero duration under the policies satisfying (34), because (16) 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

$$(\tilde{\mathbf{P}})\quad \underset{\pi}{\text{maximize}}\quad \min_{i\in I}\{r_i\}$$
$$\text{subject to}\quad (18)-(20),(34).$$

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

$$\min_{i\in I}\{r_i\}=r_{i^\flat}. \tag{35}$$

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

$$\begin{aligned}\underset{\pi}{\text{maximize}}\quad & \min_{i\in I}\{r_i\}\\ \text{subject to}\quad & r_{\text{in}}=r_{i^\flat},\\ & (18)-(20),(35).\end{aligned}$$

Note that, **(P)** is a relaxation of the above problem by dropping the constraint (35).

Lemma 5: The constraint (35) is a necessary condition for the optimality of **(P)**.

Lemma 5 can be proved in a similar way 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 (15), all users achieve the same innovative fluid rate given by r_{in}. We can revise $\pi(\cdot)$ by decreasing $\pi(\boldsymbol{s})$ for some $\boldsymbol{s}\in S$ 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

$$\begin{aligned}& r_{\text{in}}^*\ge\min_{i\in I}\{r_i^*\},\quad r_{\text{in}}^*T_1^*=K,\\ & r_i^*T_1^*+p_iT_2^*\ge K,\ \forall i\in I,\\ & \exists j\in I:\ r_j^*T_1^*+p_jT_2^*=K.\end{aligned} \tag{36}$$

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

$$r'_{\text{in}}=\frac{r_{\text{in}}^*T_1^*}{T_1^*+T_2^*}=\frac{K}{T_1^*+T_2^*}, \tag{37}$$

$$r'_i=\frac{r_i^*T_1^*+p_iT_2^*}{T_1^*+T_2^*}\ge\frac{K}{T_1^*+T_2^*},\ \forall i\in I. \tag{38}$$

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

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

It is clear that the duration of the Opportunistic Fetch stage is $T_1^* + T_2^*$ under policy $\pi'(\cdot)$. From (36) and (38), there exists some $j \in I$ such that

$$r'_j = \frac{r_j^* T_1^* + p_j T_2^*}{T_1^* + T_2^*} = \frac{K}{T_1^* + T_2^*}. \tag{39}$$

Combining (37), (38), and (39), we have that

$$r'_{\text{in}} = \min_{i \in I}\{r'_i\}. \tag{40}$$

By (15), we have that

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

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 (34) because of (40).

Appendix D
Proof of Lemma 4

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

$$r_{\text{in}}^* = \min_{i \in I}\{r_i^*\} = r_j^* < r_i^*, \ \forall i \neq j \tag{41}$$

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

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

such that $\pi^*(\boldsymbol{s}^{(1)}) > \pi^*(\boldsymbol{s}^{(2)})$; otherwise, we have that $r_{i^{\text{b}}}^* \leq r_j^*$, which contradicts (41). Consider a policy $\pi'(\cdot)$ such that

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

where Δ satisfies

$$0 < \Delta \leq \min\{p(\boldsymbol{s}^{(1)})\pi^*(\boldsymbol{s}^{(1)}), p(\boldsymbol{s}^{(2)})[1 - \pi^*(\boldsymbol{s}^{(2)})]\}. \tag{42}$$

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

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

Furthermore, if Δ satisfies

$$0 < \Delta \leq \frac{r^*_{i^{\text{b}}} - r^*_j}{2}, \tag{43}$$

we have that $r'_{\text{in}} < r'_j \leq r'_{i^{\text{b}}}$. Therefore, by choosing a value of Δ that satisfies (42) and (43), 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 $\pi^\dagger(\cdot)$:

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

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

$$\begin{aligned} &r^\dagger_{\text{in}} = r'_{\text{in}} + \frac{\mu}{2}, \quad r^\dagger_k = r'_k - \frac{\mu}{2}, \\ &r^\dagger_i \geq r'_i - \frac{\mu}{2}, \ \forall i \in I, \ i \neq k. \end{aligned}$$

Thus, we have that

$$\min_{i \in I}\{r^\dagger_i\} = r^\dagger_k = r^\dagger_{\text{in}} > r'_{\text{in}} = r^*_{\text{in}} = \min_{i \in I}\{r^*_i\},$$

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

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

$$r^*_{\text{in}} = \min_{i \in I}\{r^*_i\} = r^*_j < r^*_{i^{\text{b}}}, \ \forall j \in \bar{I}$$

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

Appendix E
Proof of Theorem 3

We will reformulate **(P)** as an LP with respect to $r_1, \ldots, r_n$. The feasible set associated with constraints (18)-(20) and (22) is denoted by $\mathcal{A}$ such that

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

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

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

for all $i \in I$ and $j \in \{1, \ldots, 2^n\}$. Let $\boldsymbol{r} = (r_1, \ldots, r_n)$. Then, the constraint (19) is given by $\boldsymbol{r} = \boldsymbol{H}\boldsymbol{\pi}$. Thus, the feasible set for **(P)** is given by

$$\mathcal{B} = \left\{ \boldsymbol{r} \in \mathbb{R}^n | \boldsymbol{r} = \boldsymbol{H}\boldsymbol{\pi}, \ \boldsymbol{\pi} \in \mathcal{A} \right\}.$$

$\mathcal{A}$ 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

$$\underset{\boldsymbol{r}}{\text{maximize}} \quad \min_{i \in I}\{r_i\} \quad \text{subject to } \boldsymbol{r} \in \mathcal{B}. \tag{44}$$

Let the optimal value of (44) be $u^* > 0$, and define the set $\mathcal{C}_{u^*} := \{\boldsymbol{r} | \min_{i \in I}\{r_i\} \geq u^*\}$. By the separation theorem [25] for convex polyhedra, there exists a separating hyperplane $\mathcal{D} := \{\boldsymbol{r} | \boldsymbol{w}^T \boldsymbol{r} = u^*\}$ which separates relint$(\mathcal{C}_{u^*})$ and relint$(\mathcal{B})$, where relint$(\cdot)$ denotes the relative interior of a set. This implies that u^* is also a solution to

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

which is equivalent to **(MW)**. The solution to (44) lies in $\mathcal{D}$; thus, $\mathcal{D}$ must be a supporting hyperplane [25] of set $\mathcal{C}_{u^*}$. It is straightforward to show that, for a hyperplane of form $\{\boldsymbol{r}|\boldsymbol{w}^T\boldsymbol{r}=u^*\}$ to be a supporting hyperplane of $\mathcal{C}_{u^*}$ for any $u^*>0$, it is necessary that the vector $\boldsymbol{w}$ has non-negative entries. This completes the proof.

Appendix F
Proof of Theorem 4

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

$$\begin{aligned}\underset{x}{\text{maximize}} \quad & \sum_{\boldsymbol{s}\in S} x(\boldsymbol{s})\hat{\rho}(\boldsymbol{w},\boldsymbol{s}) \\ \text{subject to} \quad & \sum_{\boldsymbol{s}\in S} x(\boldsymbol{s}) = 1, \\ & 0\le x(\boldsymbol{s}) \le p(\boldsymbol{s})[1+s_{i^\flat}],\ \forall \boldsymbol{s}\in S.\end{aligned} \tag{45}$$

The above problem can be solved in a greedy manner as follows. First, we select $x(\boldsymbol{s})$ with the largest value of $\hat{\rho}(\boldsymbol{w},\boldsymbol{s})$, i.e., $x(\boldsymbol{\eta}(1))$. Then, we attempt to set $x(\boldsymbol{\eta}(1))$ as close to $p(\boldsymbol{\eta}(1))[1+\eta_{i^\flat}(1)]$ as possible until (45) is satisfied. If (45) cannot be satisfied even after setting $x(\boldsymbol{\eta}(1)) = p(\boldsymbol{\eta}(1))[1+\eta_{i^\flat}(1)]$, we select $x(\boldsymbol{\eta}(2))$, and so on. This procedure is repeated until (45) is satisfied, which leads to the threshold-based solution stated in the theorem.

Appendix G
Proof of Lemma 2

Let $\pi'(\boldsymbol{s})$ denote the solution to **(P-DR)** formulated at the beginning of the jth review period in the mth system. We will first compare $\pi'(\boldsymbol{s})$ and $\pi^*(\boldsymbol{s})$. Recall $\pi^*(\boldsymbol{s})$ is the solution to **(P-WBP)** with input weights (33). Note that **(P-DR)** and **(P-WBP)** share the same weights; the only difference is that **(P-DR)** has the safety-stock requirement (30), whereas **(P-WBP)** has the constraint (24), or equivalently, (22). Note the (30) should be strictly satisfied at each review period. Following the proof of Theorem 5.1. in [22], we have that

$$\frac{u^{(m)}+\hat{B}_{i^\flat}(jL^{(m)})-B_R(jL^{(m)})}{L^{(m)}}\to 0,\ \forall j\ge\lceil\alpha\rceil,$$

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

$$|r_{\text{in}}-r_{i^\flat}| = \left|\frac{u^{(m)}+\hat{B}_{i^\flat}(jL^{(m)})-B_R(jL^{(m)})}{L^{(m)}}\right| \le \frac{\epsilon}{2^{n+1}} \tag{46}$$

for all $m>M_1$. Furthermore, (46) implies that

$$|\pi'(\boldsymbol{s})-\pi^*(\boldsymbol{s})| \le \frac{\epsilon}{2^{n+1}p(\boldsymbol{s})[1+s_{i^\flat}]},\ \forall\boldsymbol{s}\in S \tag{47}$$

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

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

for all $m>M_2$. Combining the above with (47), we have that

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

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 way, 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 jth review period under fluid scaling, they achieve rates that are arbitrarily close to the optimal rates for **(P-WBP)** with the weights defined in (33). 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 (33). Define $a_i, i\in I$ as follows:

$$a_i := \frac{\sum_{\boldsymbol{s}\in S}p(\boldsymbol{s})\pi^*(\boldsymbol{s})s_i}{\sum_{\boldsymbol{s}\in S}p(\boldsymbol{s})\pi^*(\boldsymbol{s})}\le 1. \tag{48}$$

We have that

$$\begin{aligned}&\sum_{\boldsymbol{s}\in S}p(\boldsymbol{s})\pi^*(\boldsymbol{s}) \\ &= \sum_{\boldsymbol{s}\in S:s_i=1}p(\boldsymbol{s})\pi^*(\boldsymbol{s}) + \sum_{\boldsymbol{s}\in S:s_i=0}p(\boldsymbol{s})\pi^*(\boldsymbol{s}) &(49)\\ &\le \sum_{\boldsymbol{s}\in S:s_i=1}p(\boldsymbol{s})\pi^*(\boldsymbol{s}) + \sum_{\boldsymbol{s}\in S:s_i=1}\frac{1-p_i}{p_i}p(\boldsymbol{s})\pi^*(\boldsymbol{s}) &(50)\\ &= \sum_{\boldsymbol{s}\in S:s_i=1}\frac{1}{p_i}p(\boldsymbol{s})\pi^*(\boldsymbol{s}). &(51)\end{aligned}$$

We can derive the inequality (50) as follows. For any channel state $\boldsymbol{s}\in S$ such that $s_i=0$, there exists a channel state $\boldsymbol{\sigma}\in S$ such that

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

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

$$\pi^*(\boldsymbol{\sigma})\ge\pi^*(\boldsymbol{s}). \tag{52}$$

Also we have that

$$\frac{p(\boldsymbol{s})}{p(\boldsymbol{\sigma})} = \frac{\prod_{j\in I}p_j^{s_j}(1-p_j)^{1-s_j}}{\prod_{j\in I}p_j^{\sigma_j}(1-p_j)^{1-\sigma_j}} = \frac{1-p_i}{p_i}. \tag{53}$$

By applying (52) and (53) to the second summation of (49), we obtain the inequality (50). Combining (48) and (51), we conclude that

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

According to (24), we have that

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

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

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

From (24), or equivalently (22), we have that

$$r_{\text{in}} = r_{i^{\mathrm{b}}} \geq \frac{p_{i^{\mathrm{b}}}}{2}.$$

According to (15), the innovative fluid rate achieved by each user under the solution to **(P-WBP)** in the jth review period is bounded below by $p_{i^{\mathrm{b}}}/2$ for any $j \in \mathbb{Z}_+$. Therefore, the asymptotic decoding delay under WBP-DR policy is bounded above by $2K/p_{i^{\mathrm{b}}}$. Because the optimal innovative fluid rate achieved by user i^{b} is no more than $p_{i^{\mathrm{b}}}$, the optimal decoding delay is bounded below by $K/p_{i^{\mathrm{b}}}$. Thus, the theorem is proved.

APPENDIX I
PROOF OF THEOREM 6

Without loss of generality, we assume $p_1 \leq p_2$, i.e., $i^{\mathrm{b}} = 1$. 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 $r_{\text{in}}^{(j)}$ denote the associated rates with policy $\pi^{(j)}(\cdot)$. We consider the following cases:

Case 1: $p_1p_2 \geq \frac{1}{2}$. We have that, by solving **(P-WBP)**,

$$\pi^{(j)}((11)) = \frac{1}{2p_1p_2},\ \pi^{(j)}(\boldsymbol{s}) = 0,\ \forall \boldsymbol{s} \neq (11);$$
$$r_1^{(j)} = r_2^{(j)} = r_{\text{in}}^{(j)} = \frac{1}{2},\quad \forall j = 1,2.$$

Based on (15), both users achieve the same innovative fluid rate $\frac{1}{2}$, irrespective of the priority among the users.

Case 2: $p_1p_2 < \frac{1}{2}$ and $1-2p_1 \geq p_1(1-p_2)$. We have that

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

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

Case 3: $p_1p_2 < \frac{1}{2}$ and $1-2p_1 < p_1(1-p_2)$. We have that

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

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. 4 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 (24), 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 $r_{\text{in}}^{(2)}$, 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

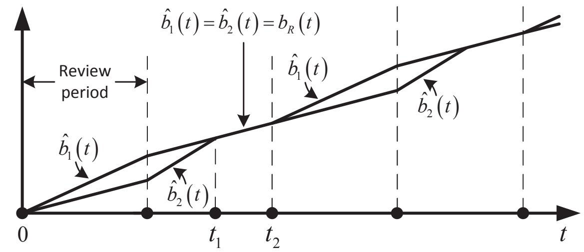

Fig. 4. Example of trajectories for Case 3.

$$\kappa = \frac{r_1^{(1)} - r_2^{(1)}}{r_2^{(2)} - r_1^{(2)}} > 0. \tag{54}$$

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

$$c = \frac{\hat{r}_1^{(1)} + \lceil\kappa\rceil\hat{r}_1^{(2)}}{1+\lceil\kappa\rceil} = \frac{r_1^{(1)} + \lceil\kappa\rceil r_1^{(2)}}{1+\lceil\kappa\rceil}. \tag{55}$$

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: $p_1p_2 \geq \frac{1}{2}$ and *Case 2:* $p_1p_2 < \frac{1}{2}$ 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: $p_1p_2 < \frac{1}{2}$ and $1-2p_1 < p_1(1-p_2)$. We have that

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

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 (24), $\pi^*(\cdot)$ 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

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

Therefore, we have that

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

Since $r_1^* = r_2^*$ and based on (54), one can show that

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

Combining the above with (55), we have that

$$\frac{c}{r_1^*} = \frac{r_1^{(1)} + \lceil\kappa\rceil r_1^{(2)}}{r_1^{(1)} + \kappa r_1^{(2)}} \cdot \frac{1+\kappa}{1+\lceil\kappa\rceil} = \frac{\theta + \lceil\kappa\rceil}{\theta+\kappa} \cdot \frac{1+\kappa}{1+\lceil\kappa\rceil} \quad (56)$$

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 (56). Define $x = \lceil\kappa\rceil \in \mathbb{Z}_+$. For fixed $\theta \in (1, 2]$, (56) satisfies

$$\frac{\theta + x}{\theta+\kappa} \cdot \frac{1+\kappa}{1+x} > \frac{\theta+x}{\theta+x-1} \cdot \frac{x}{1+x}, \quad (57)$$

because the LHS of (57) is increasing in $\kappa \in (x-1, x]$. One can show that the RHS of (57) is increasing in $x \in \mathbb{Z}_+$. Thus, for fixed $\theta \in (1, 2]$, (56) 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, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi, “A survey on 3GPP heterogeneous networks,” *IEEE Wireless Communications*, vol. 18, no. 3, pp. 10–21, 2011.

[2] S. Peters, A. Panah, K. Truong, and R. Heath, “Relay architectures for 3GPP LTE-advanced,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, pp. 1–14, 2009.

[3] D. Lecompte and F. Gabin, “Evolved multimedia broadcast/multicast service (eMBMS) in lte-advanced: overview and rel-11 enhancements,” *IEEE Communications Magazine*, vol. 50, no. 11, pp. 68–74, 2012.

[4] D. Gunduz, O. Simeone, A. J. Goldsmith, H. V. Poor, and S. Shamai, “Multiple multicasts with the help of a relay,” *IEEE Transactions on Information Theory*, vol. 56, no. 12, pp. 6142–6158, 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 Transactions on Information Theory*, vol. 60, no. 6, pp. 3559–3575, 2014.

[6] J. Du, M. Xiao, M. Skoglund, and M. Médard, “Wireless multicast relay networks with limited-rate source-conferencing,” *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 8, pp. 1390–1401, 2013.

[7] K. Loa, C.-C. Wu, S.-T. Sheu, Y. Yuan, M. Chion, D. Huo, and L. Xu, “IMT-advanced relay standards [WiMAX/LTE update],” *IEEE Communications Magazine*, vol. 48, no. 8, pp. 40–48, 2010.

[8] D. Lun, M. Médard, R. Koetter, and M. Effros, “On coding for reliable communication over packet networks,” *Physical Communication*, vol. 1, no. 1, pp. 3–20, 2008.

[9] B. Swapna, A. Eryilmaz, and N. B. Shroff, “Throughput-delay analysis of random linear network coding for wireless broadcasting,” *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6328–6341, 2013.

[10] Y. Yang and N. Shroff, “Throughput of rateless codes over broadcast erasure channels,” in *Proc. of 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 Transactions on Information Theory*, vol. 54, no. 12, pp. 5511–5524, 2008.

[12] B. Shrader and A. Ephremides, “Queueing delay analysis for multicast with random linear coding,” *IEEE Transactions on Information Theory*, vol. 58, no. 1, pp. 421–429, 2012.

[13] R. Cogill, B. Shrader, and A. Ephremides, “Stable throughput for multicast with random linear coding,” *IEEE Transactions on Information Theory*, vol. 57, no. 1, pp. 267–281, 2011.

[14] D. Lun, P. Pakzad, C. Fragouli, M. Médard, and R. Koetter, “An analysis of finite-memory random linear coding on packet streams,” in *IEEE WiOpt 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 *Communications Workshops (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 255–260.

[16] K. Sundaresan and S. Rangarajan, “Cooperation versus multiplexing: Multicast scheduling algorithms for OFDMA relay networks,” *IEEE/ACM Transactions on Networking*, vol. 22, no. 3, pp. 756–769, 2014.

[17] D. J. MacKay, “Fountain codes,” *IEE Proceedings-Communications*, vol. 152, no. 6, pp. 1062–1068, 2005.

[18] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, and P. Whiting, “Scheduling in a queuing system with asynchronously varying service rates,” *Probability in the Engineering and Informational Sciences*, vol. 18, no. 02, pp. 191–217, 2004.

[19] K. Avrachenkov, A. Piunovskiy, Y. Zhang *et al.*, “Asymptotic fluid optimality and efficiency of the tracking policy for bandwidth-sharing networks,” *Journal of Applied Probability*, vol. 48, no. 1, pp. 90–113, 2011.

[20] N. Bäuerle, “Optimal control of queueing networks: an approach via fluid models,” *Advances in Applied Probability*, vol. 34, no. 2, pp. 313–328, 2002.

[21] M. R. Garey and D. S. Johnson, *Computers and intractability*. Freeman, 2002, vol. 29.

[22] C. Maglaras, “Discrete-review policies for scheduling stochastic networks: Trajectory tracking and fluid-scale asymptotic optimality,” *Annals of Applied Probability*, vol. 10, no. 3, pp. 897–929, 2000.

[23] A. L. Stolyar, “Maxweight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic,” *Annals of Applied Probability*, pp. 1–53, 2004.

[24] S. Meyn, *Control techniques for complex networks*. Cambridge Univ Pr, 2008.

[25] R. T. Rockafellar, *Convex analysis*. Princeton University Press, 1997.