Fresher Content or Smoother Playback? 
A Brownian-Approximation Framework for Real-Time Video Delivery in Wireless Networks

Ping-Chun Hsieh*, Xi Liu+, I-Hong Hou+
*National Chiao Tung University  +Texas A&M University
pinghsieh@nctu.edu.tw, xiliu.tamu@gmail.com, ihou@tamu.edu

ABSTRACT
This paper presents a Brownian-approximation framework to optimize the quality of experience (QoE) for real-time video delivery in wireless networks. In real-time video delivery, one major challenge is to tackle the natural tension between the two most critical QoE metrics: playback latency and video playback interruption. To study this trade-off, we first propose an analytical model that precisely captures all aspects of the playback process of a real-time video stream, including playback latency, video interruptions, and packet dropping. Built on this model, we show that the playback process of a real-time video can be approximated by a two-sided reflected Brownian motion. Through such Brownian approximation, we are able to study the fundamental limits of the two QoE metrics and characterize a necessary and sufficient condition for a set of QoE performance requirements to be feasible. We propose a simple scheduling policy that satisfies any feasible set of QoE performance requirements and then demonstrate the wide applicability of the proposed policy in network utility maximization for QoE. Finally, simulation results verify the accuracy of the proposed approximation and show that the proposed policy outperforms other popular baseline policies.

1 INTRODUCTION
Real-time video delivery has become ubiquitous due to the widespread use of mobile devices and the rapid development of various live streaming platforms, such as YouTube, Facebook Live, and Twitch. These platforms support not only peer-to-peer video chatting but also a variety of emerging applications, such as virtual reality and cloud video-gaming. In these live video streaming applications, video contents are continuously generated by the content providers in real-time, and the video content is designated to be played at the video client with low latency. To guarantee quality of experience (QoE) of video clients, real-time streaming applications need to tackle a natural tension between the two most critical factors of QoE: playback latency and video playback interruption. Playback latency refers to the difference between the generation time of a video frame at the video source and its designated playback time at the video client. Playback latency reflects the freshness of the video content and needs to be kept as small as possible. To maintain a constantly low playback latency, each video is configured to meet a certain playback latency requirement, and the video contents that are not delivered to the client by the designated playback time will be dropped. Meanwhile, due to the lack of video content to play, the video client instantly experiences video playback interruption. To achieve smooth playback, the amount of video playback interruption also needs to be kept as small as possible. However, with a more stringent playback latency, it becomes more difficult to avoid playback interruption as there is less room for coping with any randomness in network condition during video delivery. This issue becomes even more challenging in a wireless network environment due to the shared wireless resource and the unreliable nature of wireless channels.

In this paper, we aim to address this challenge by studying the trade-off between playback latency and video interruption for real-time videos as well as the trade-off of such QoE performance metrics between different clients. This paper can be highlighted as follows:

• First, we propose an analytical model that precisely captures all aspects of the playback process of a real-time video stream, including the packet generation process, the playback latency, packet dropping, and video interruptions. The proposed model also addresses the unreliable nature of wireless transmissions. Through Brownian approximation, we show that the playback process of real-time videos can be approximated by a two-sided reflected Brownian motion.
• Based on the proposed model and the approximation, we study the fundamental limits of the trade-off between the two most important QoE metrics: the playback latency and video interruptions, among all clients. Moreover, we characterize a necessary and sufficient condition for a set of QoE performance requirements to be feasible, given the reliabilities of wireless links.
• Next, we propose a simple policy that jointly determines the amount of playback latency of each client and the scheduling decision of each packet transmission. We show that this policy is able to satisfy any feasible set of QoE performance requirements, and hence we say that it is QoE-optimal.
• Through numerical simulations, we show that the proposed approximation approach can capture the original playback processes accurately, and the proposed scheduling policy indeed outperforms the other popular baseline policies.

The rest of the paper is organized as follows. Section 2 describes the system model and problem formulation. Section 3 discusses the characterization of the playback process. Section 4 presents the framework of Brownian approximation as well as the fundamental network properties. Section 5 presents the proposed scheduling policy and the proof of its QoE-optimality. Section 6 discusses the asymptotic results regarding playback latency. Simulation results are provided in Section 7, and Section 8 provides an overview of the related research. Finally, Section 9 concludes the paper.
2 SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we formally describe the wireless network model, the model for real-time video streaming, and the problem formulation.

2.1 Network Topology and Channel Model

We consider a wireless network with one AP that serves \( N \) video clients, each of which is associated with one packet stream of a real-time video generated by a video source. For ease of exposition, we assume that all the videos are streamed in downlink, i.e. from the AP to the clients. For temporary storage of the video content to be played, each video is associated with two video buffers: one buffer is on the client side, and the other is maintained by the AP. When the video source generates a packet, the packet is first forwarded to the AP and stored at the AP-side buffer. The AP then forwards the packet to the client to be stored at the client-side buffer. The packet is consumed by the client when the client plays the corresponding video frame, at which time the packet leaves the system. Since the bandwidth between the AP and the video source is usually much larger than the bandwidth at the edge, we also assume that latency between the AP and the source of video content is negligible. For each client \( n \), we use \( Q_n(t) \) and \( Q_n(t) \) to denote the number of available data packets in the client-side buffer and that in the AP-side buffer at time \( t \), respectively. Figure 1 shows an example of the AP-side and client-side video buffers with two clients.

Time is slotted, and the size of each time slot is chosen to be the total time required for one packet transmission. In each time slot, the AP can transmit one packet to exactly one of the video clients. If the AP chooses to transmit a packet to a client whose AP-side buffer is empty, then the AP will simply transmit a dummy packet. By using dummy packets, we can assume that the AP employs a causal work-conserving scheduling policy that always chooses a client to transmit in each time slot based on the past observed history. Let \( I_n(t) \) be the indicator of the event that client \( n \) is scheduled for a packet transmission at time slot \( t \). Under a work-conserving policy, we have \( \sum_{n=1}^{N} I_n(t) = 1 \), for all \( t \geq 0 \).

![Figure 1: An example of video buffers with two clients.](image)

Regarding wireless transmissions, we consider unreliable wireless packet transmissions that are subject to interference and collision from other neighboring networks. Since all links in the network experience a similar level of interference, we assume that all links have similar reliability. Specifically, each packet transmitted by the AP will be delivered successfully with probability \( p \in (0, 1) \). The AP will be instantly notified about the outcome of the transmission via the acknowledgment from the client and can choose to retransmit the packet in a later time slot if the current transmission fails.

2.2 The Model for Real-Time Video Streaming

Each client is watching a real-time video stream. The stream of client \( n \) generates one video packet every \( 1/\lambda_n \) slots, where \( 1/\lambda_n \) is a finite positive integer. Hence, the average video bitrate of client \( n \) is \( \lambda_n \) packets per time slot. For ease of exposition, for each client \( n \), time slots are further grouped into intervals, where each interval of client \( n \) consists of \( 1/\lambda_n \) consecutive time slots. We consider real-time video streams with a fixed playback latency of \( \ell_n \) intervals. Specifically, for each client \( n \), the video content generated at time slot \( t \) is forwarded immediately to the AP and is designated to be played by the client at time slot \( t + \ell_n/\lambda_n \). The playback latency is intended to reduce potential video rebuffering and hence achieve smoother playback of a real-time video while guaranteeing the freshness of the video contents. Moreover, to maintain a fixed playback latency, a packet that is not delivered to the client by its designated playback time will be dropped by the AP. When this happens, the client experiences video playback interruption due to the lack of video packets to play. For the rest of the paper, we call this event an interruption. For each client \( n \), we use \( D_n(t) \) to denote the total number of intervals in which video interruption occurs up to time \( t \), with \( D_n(0) = 0 \). Since a video interruption event occurs only when a video packet is dropped, \( D_n(t) \) also represents the total number of dropped packets up to time \( t \).

Consider an example of the real-time video playback process with \( \lambda_n = 1/2 \) (or equivalently one packet is played every 2 time slots), and \( \ell_n = 2 \) (or equivalently 4 slots), as illustrated in Figure 2. Since \( \ell_n = 2 \), we know there are two video packets (dubbed as packet 1 and packet 2 in Figure 2) available for transmission at the AP at \( t = 0 \). In particular, packet 1 and packet 2 are generated at \( t = -2 \) and \( t = 0 \), respectively. In this example, the client receives packets in time slots 1, 4, 8, and 9. The client plays packet 1 at the end of time slot 2 since it successfully receives packet 1. Similarly, the client plays packet 2 at the end of time slot 4 since it receives packet 2 within the playback latency. By contrast, as the client fails to receive packet 3 within the playback latency, video interruption begins at the end of time slot 6. Meanwhile, to maintain a fixed playback latency of \( \ell_n = 2 \), packet 3 is dropped by the AP at the end of time slot 6. At time 8, the video playback resumes as the client receives packet 4 by time slot 8. Note that the AP is able to deliver the packet 5 at time 9 since packet 5 is generated at time 6 and hence is already available for transmission.

![Figure 2: An example of real-time video playback process with \( \lambda_n = 1/2 \) and \( \ell_n = 2 \).](image)
2.3 Problem Formulation

To substantiate the trade-off between the playback latency and the video interruptions, we formally define the capacity region for QoE and introduce the notion of QoE-optimality as follows.

**Definition 2.1 (Capacity Region for QoE and QoE-Optimality).** A vector \( \delta = (\delta_0, \delta_1, \ldots, \delta_N) \) is said to be feasible if there exists a scheduling policy such that under \( \sum_{n=1}^{N} \tau_n \leq \delta_0 \), we have

\[
\limsup_{t \to \infty} \frac{D_n(t)}{t} \leq \delta_n, \quad (1)
\]

for every \( n \in \{1, \ldots, N\} \). Moreover, the capacity region for QoE is defined as the set of all feasible vectors. A scheduling policy is said to be QoE-optimal if it can achieve every point in the capacity region for QoE.

As addressed by this definition, we are interested in the long-term average video interrupt rates that can be achieved under a constraint of the total playback latency \( \sum_{n=1}^{N} \ell_n =: \ell_{\text{tot}} \). The main objective of this paper is to design a QoE-optimal policy that jointly makes scheduling decisions and determines the allocation of the latency budget among the clients.

3 CHARACTERIZATION OF THE BUFFERING AND PLAYBACK PROCESSES

In this section, we formally characterize the playback process of a real-time video with playback latency. As discussed in Section 2.1, each video is associated with two video buffers: one buffer is on the client side, and the other is maintained by the AP. Recall that \( B_n(t) \) and \( Q_n(t) \) denote the number of available data packets in the client-side buffer and that in the AP-side buffer at time \( t \), respectively. Given the fixed playback latency \( \ell_n \), we know that at any point of time, the amount of available and yet unplayed video data, which can be either in the AP-side buffer or in the client-side buffer, is exactly \( \ell_n \) packets. Therefore,

\[
Q_n(t) + B_n(t) = \ell_n, \quad \forall t \geq 0. \quad (2)
\]

Then, both \( Q_n(t) \) and \( B_n(t) \) are non-negative integers with \( 0 \leq Q_n(t) \leq \ell_n \) and \( 0 \leq B_n(t) \leq \ell_n \), for all \( t \geq 0 \). Suppose that the client-side buffer is initially empty, i.e. \( B_n(0) = 0 \), for all \( n \). By (2), we thereby know \( Q_n(0) = \ell_n \), for all \( n \). Note that the data packets stored in the AP-side buffers at time 0 are essentially generated by the content provider during time \([-(\ell_n - 1)/\lambda_n, 0]\).

As described in Section 2.1, if the AP chooses to transmit a packet to client \( n \) at time \( t \) with \( Q_n(t) = 0 \) (i.e. the AP-side buffer for client \( n \) is empty), the AP will simply transmit a dummy packet to client \( n \). Let \( U_n(t) \) be the number of dummy packets delivered by the AP to the client \( n \) by time \( t \), with \( U_n(0) = 0 \). Let \( A_n(t) \) be the number of data packets received by client \( n \) up to time \( t \), with \( A_n(0) = 0 \). Upon the designated playback time of each video packet, client \( n \) either consumes a packet from the client-side buffer if \( B_n(t) \geq 1 \), or experiences video interruption if \( B_n(t) = 0 \). Let \( S_n(t) \) be the number of video packets that have been played by client \( n \) by time \( t \), with \( S_n(0) = 0 \). Then, we have

\[
B_n(t) = A_n(t) - S_n(t). \quad (3)
\]

Since a video packet is dropped only when \( t \) is an integer multiple of \( 1/\lambda_n \) and \( B_n(t) = 0 \), we have

\[
D_n(t + 1) = \begin{cases} 
D_n(t) + 1, & \text{if } B_n(t) = 0 \text{ and } t \in \{k/\lambda_n, k \in \mathbb{N}\} \\
D_n(t), & \text{otherwise.}
\end{cases} \quad (4)
\]

Therefore, we know that \( B_n(t) = 0 \) if \( D_n(t + 1) - D_n(t) = 1 \). Define

\[
Z_n(t) := (A_n(t) + U_n(t)) - (D_n(t) + S_n(t)). \quad (5)
\]

Note that \( A_n(t) + U_n(t) \) is the total number of delivered packets, and \( S_n(t) + D_n(t) \) is the number of packets that the client \( n \) should have played if there is no video interruption. Therefore, \( Z_n(t) \) loosely reflects the status of the client-side buffer, with dummy packets included. By the definitions of \( B_n(t) \) and \( Z_n(t) \) in (3) and (5), we can rewrite \( B_n(t) \) as

\[
B_n(t) = Z_n(t) - U_n(t) + D_n(t). \quad (6)
\]

We summarize the important properties of \( B_n(t) \) as follows. For any \( t \geq 0 \), we have

\[
B_n(t) = (Z_n(t) - U_n(t)) + D_n(t) \geq 0, \quad (7)
\]

\[
D_n(t + 1) - D_n(t) \in \{0, 1\}, \quad D_n(t) = 0, \quad (8)
\]

\[
B_n(t)(D_n(t + 1) - D_n(t)) = 0. \quad (9)
\]

Now we turn to the AP-side buffer. Recall that \( U_n(t) \) denotes the number of dummy packets received by the client \( n \) by time \( t \). As a dummy packet is transmitted to the client \( n \) only if the AP-side buffer of the client \( n \) is empty, we know \( U_n(t) \) can be updated as

\[
U_n(t + 1) = \begin{cases} 
U_n(t) + 1, & \text{if } Q_n(t) = 0 \text{ and a packet is delivered to client } n \text{ at time } t, \\
U_n(t), & \text{otherwise.}
\end{cases} \quad (10)
\]

Similar to (7)-(9), we summarize the useful properties of \( Q_n(t) \) as follows: for any \( t \geq 0 \),

\[
Q_n(t) = \left( \ell_n - (Z_n(t) + D_n(t)) \right) + U_n(t) \geq 0, \quad (11)
\]

\[
U_n(t + 1) - U_n(t) \in \{0, 1\}, \quad U_n(0) = 0, \quad (12)
\]

\[
Q_n(t)(U_n(t + 1) - U_n(t)) = 0. \quad (13)
\]

where (11) follows directly from (2) and (6). Note that the stochastic processes \( D_n(t), U_n(t), B_n(t), Q_n(t), A_n(t), S_n(t), \) and \( Z_n(t) \) are right-continuous with left limits for every sample path since all of them change values only at integer time. By (7)-(9) and (11)-(13), we are able to connect \( Z_n(t) \) with \( D_n(t) \) and \( U_n(t) \) as follows.

**Theorem 1.** For any \( Z_n(t) \), there exists a unique tuple of processes \( (D_n(t), U_n(t), B_n(t), Q_n(t)) \) that satisfies (7)-(9) and (11)-(13), for every sample path.

**Proof.** We prove this by the two-sided reflection mapping. Specifically, we take \( Z_n(t) \) as the process of interest and let \( 0 \) and \( \ell_n \) be the lower and upper barrier, respectively. Since \( Z_n(t) \) is right-continuous with left limits, the uniqueness of \( D_n(t), U_n(t), B_n(t), Q_n(t) \) follows directly from [31, Theorem 14.8.1]. \( \square \)

In addition to Theorem 1, we further show that \( D_n(t) \) and \( U_n(t) \) can be uniquely characterized by a pair of recursive equations.
To analyze video interruption, we start by introducing dummy packets included. Recall that (7)-(9) are satisfied. Based on the same argument, we also have that (11) holds if and only if (11)-(13) are satisfied.

By combining Theorem 1 and 2, we have the following:

**Corollary 1.** For any \( Z_n(t) \), there exists a pair of non-decreasing processes \( (D_n(t), U_n(t)) \) that uniquely satisfies (14)-(15), for every sample path.

**Remark 1.** The two-sided reflection mapping \( \{Z_n(t), D_n(t), U_n(t)\} \) is also called double Skorokhod mapping in the literature [21].

**Remark 2.** From (14)-(15), it is easy to check that under any fixed sample path of \( Z_n(t) \), a larger \( \ell_n \) will lead to smaller \( D_n(t) \) and \( U_n(t) \). This fact manifests the fundamental trade-off between the playback latency and the video interruptions.

**Remark 3.** One special case of the model for real-time video streaming is the widely-used real-time wireless network model [RT] [13, 19, 20, 24]. Specifically, by selecting \( \ell_n = 1 \) for all \( n \) and \( \lambda_n = \lambda_m \) for any pair \( n, m \), the conventional RT model can be recovered by our model for real-time video streaming. This also manifests that the RT model does not address the effect of playback latency on network performance. Moreover, different from the conventional studies on real-time wireless scheduling, our goal is to propose a scheduling policy that jointly optimizes overall video interruption and playback latency, instead of simply achieving the required packet delivery ratios.

For convenience, we summarize the key notations in Table 1.

### 4 THE BROWNIAN-APPROXIMATION FRAMEWORK

In this section, we formally introduce the Brownian-approximation framework for real-time video playback processes.

#### 4.1 Fundamental Network Properties

To analyze video interruption, we start by introducing \( Z(t) \) as

\[
Z(t) := \sum_{n=1}^{N} \frac{Z_n(t)}{p} = \sum_{n=1}^{N} \left( A_n(t) + U_n(t) \right) - \left( C_n(t) + S_n(t) \right) \tag{16}
\]

Note that \( Z(t) \) is right-continuous with left limits since \( Z_n(t) \) is right-continuous with left limits, for all \( n \). Moreover, \( Z(0) = 0 \) as \( Z_n(0) = 0 \), for all \( n \). As \( Z(t) \) is a weighted sum of \( Z_n(t) \), \( Z(t) \) loosely reflects the network-wide buffer status on the clients’ side, with dummy packets included. Recall that \( \ell_{\text{tot}} := \sum_{n=1}^{N} \ell_n \). By Corollary

| Notation | Description |
|----------|-------------|
| \( \lambda_n \) | average video bitrate of client \( n \) (packet/slot) |
| \( \ell_n \) | playback latency of client \( n \) (in interval) |
| \( \ell_{\text{tot}} \) | total playback latency of all clients (\( \sum_{n=1}^{N} \ell_n \)) |
| \( p \) | channel reliability of the network |
| \( A_n(t) \) | number of packets received by client \( n \) up to \( t \) |
| \( S_n(t) \) | number of frames that client \( n \) plays up to \( t \) |
| \( D_n(t) \) | amount of video interruption of client \( n \) up to \( t \) |
| \( U_n(t) \) | number of dummy packets received by client \( n \) up to \( t \) |
| \( B_n(t) \) | \( A_n(t) - S_n(t) \) (number of data packets in the client-side buffer at time \( t \)) |
| \( Q_n(t) \) | \( \ell_n - B_n(t) \) (number of data packets for client \( n \) in the AP-side buffer at time \( t \)) |
| \( Z_n(t) \) | \( A_n(t) + U_n(t) - (D_n(t) + S_n(t)) \) (reflects client-side buffer status, including dummy packets) |
| \( \tilde{Z}_n(t) \) | \( A_n(t) + U_n(t) - \lambda_n \tau \) |
| \( \bar{Z}(t) \) | sum of \( Z_n(t)/p \) over all clients |
| \( \bar{\tilde{Z}}(t) \) | sum of \( \tilde{Z}_n(t)/p \) over all clients |
| \( \overline{Z}_n(t), \tilde{Z}(t) \) | fluid limits of \( Z_n(t) \) and \( Z(t) \) |
| \( \bar{Z}_n(t), \bar{Z}(t) \) | diffusion limits of \( Z_n(t) \) and \( Z(t) \) |
| \( D'_n(t), U'_n(t) \) | Brownian-approximated version of \( D_n(t), U_n(t) \) |
| \( D(t), U(t) \) | the unique pair of processes that satisfy the two-sided reflection mapping from \( D'_n(t) \) under the total playback latency \( \ell \) |
| \( D(t), U(t) \) | the unique pair of processes that satisfy the two-sided reflection mapping from \( Z(t) \) |
| \( \mathcal{H}_t \) | system history up to time \( t \) |

1, we know that given the process \( Z(t) \), there exists a unique pair of non-decreasing processes \( (D(t), U(t)) \) that satisfies

\[
D(t) = \sup_{0 \leq \tau \leq t} \left( -Z(\tau) + U(\tau) \right)^+, \tag{17}
\]

\[
U(t) = \sup_{0 \leq \tau \leq t} \left( Z(\tau) + D(\tau) - \frac{\ell_{\text{tot}}}{p} \right)^+, \tag{18}
\]

where \((\cdot)^+ = \max\{0, \cdot\} \). Note that as \( Z(0) = 0 \), we also have \( D(0) = 0 \) and \( U(0) = 0 \). Next, we describe an important property of \( D(t) \) and \( D_n(t) \) that holds regardless of the employed policy.

| Theorem 3. Under any scheduling policy, we have |
| \( D(t) \leq \frac{1}{p} \sum_{n=1}^{N} D_n(t) \) \tag{19} |

for all \( t \geq 0 \) and for every sample path.

**Proof.** Due to the space limitation, the proof is provided in Appendix A.1. □
### 4.2 Brownian Approximation For Real-Time Video Delivery

In this section, we are ready to apply Brownian approximation to characterize the behavior of playback interruption.

#### 4.2.1 Approximation Through the Fluid Limit and the Diffusion Limit

We first provide an outline of the approximation approach as follows: consider the fluid limit and diffusion limit of \( Z_n(t) \) as

\[
Z_n(t) := \lim_{k \to \infty} \frac{Z_n(kt)}{k},
\]

\[
\tilde{Z}_n(t) := \lim_{k \to \infty} \frac{Z_n(kt) - k\tilde{Z}_n(t)}{\sqrt{k}},
\]

respectively. Generally speaking, the fluid limit and the diffusion limit and the diffusion limit are meant to capture the evolution of a stochastic process based on the Strong Law of Large Numbers (SLLN) and the Central Limit Theorem (CLT), respectively [7]. For ease of exposition, we will focus on ergodic scheduling policies under which \( Z_n(t + 1) - Z_n(t) \) forms a positive recurrent Markov chain. In this case, both limits in (20)-(21) exist [31, Section 4.4], and the fluid limit can be further written as \( Z_n(t) \rightarrow Z^*(t) \). We consider the following approximation for \( Z_n(t) \) [7, Section 6.5]:

\[
Z_n(t) \overset{d}{=} \tilde{Z}_n(t) + Z^*(t) \overset{d}{=} Z_n(t),
\]

where \( \overset{d}{=} \) means that the two stochastic processes are approximately equal in distribution. By (22), we also know \( Z_n(t) \) is right-continuous with left limits, for every sample path. By Corollary 1, we know that given the process \( Z_n(t) \), there exists a unique pair of non-decreasing processes \( D_n(t), U_n(t) \) that satisfies

\[
D_n(t) = \sup_{0 \leq r \leq t} \left( -Z_n(r) + U_n(r) \right)^+, \quad (23)
\]

\[
U_n(t) = \sup_{0 \leq r \leq t} \left( Z_n(r) + D_n(r) - \ell_n \right)^+. \quad (24)
\]

Subsequently, based on Theorem 2 and (22)-(24), we consider the following approximation for \( D_n(t) \) and \( U_n(t) \):

\[
D_n(t) \overset{d}{=} D^*_n(t), \quad U_n(t) \overset{d}{=} U^*_n(t). \quad (25)
\]

Similar to (20)-(21), define the fluid limit and diffusion limit of \( Z(t) \)

\[
\bar{Z}(t) := \lim_{k \to \infty} \frac{Z(kt)}{k},
\]

\[
\hat{Z}(t) := \lim_{k \to \infty} \frac{Z(kt) - k\bar{Z}(t)}{\sqrt{k}},
\]

Again, under an ergodic scheduling policy, \( \{Z(t + 1) - Z(t), t \geq 0\} \) forms a positive recurrent Markov chain, and hence we know both limits in (26)-(27) exist [31, Section 4.4]. We will explicitly characterize \( \bar{Z}(t) \) and \( \hat{Z}(t) \) in Section 4.2.2. Similar to (22), we consider the following Brownian approximation for \( Z(t) \) as

\[
Z(t) \overset{d}{=} \bar{Z}(t) + \hat{Z}(t) \overset{d}{=} Z^*(t),
\]

Next, we further define two processes \( D^*(t; \ell_{tot}) \) and \( U^*(t; \ell_{tot}) \) as

\[
D^*(t; \ell_{tot}) = \sup_{0 \leq r \leq t} \left( -Z^*(r) + U^*(r; \ell_{tot}) \right)^+, \quad (29)
\]

\[
U^*(t; \ell_{tot}) = \sup_{0 \leq r \leq t} \left( Z^*(r) + D^*(r; \ell_{tot}) - \ell_{tot} \right)^+/p. \quad (30)
\]

Since \( Z^*(t) \) is right-continuous with left limits, by Theorem 2 we know that \( D^*(t; \ell_{tot}) \) and \( U^*(t; \ell_{tot}) \) can be uniquely characterized by (29)-(30). Note that here we use the notations \( D^*(t; \ell_{tot}) \) and \( U^*(t; \ell_{tot}) \) to make explicit their dependence on the total playback latency. Figure 3 summarizes the general recipe of the Brownian approximation framework considered in this paper. Up to this point, we have discussed how to construct the approximation of interest with the help of the fluid and diffusion limits as well as the two-sided reflection mapping. As suggested by Figure 3, we shall proceed to characterize \( Z^*(t) \) and \( D^*(t; \ell_{tot}) \) (Section 4.2.2) as well as derive \( Z_n(t) \) and \( D^*_n(t) \) under the proposed policy (Section 5).

![Figure 3: The Brownian approximation framework considered in this paper.](image-url)
where (36) follows from the fact that the AP transmits to exactly one client in each time slot. Note that (36) always holds, for all work-conserving scheduling policies. Hence, for any \( t \geq 0 \),
\[
\mathbb{E}[(\tilde{Z}(t + 1) - \varepsilon(t + 1)) - (\tilde{Z}(t) - \varepsilon t)|\mathcal{H}_t] = 0,
\]
and the process \( (\tilde{Z}(t) - \varepsilon t, t \geq 0) \) is said to be a martingale. As \( \tilde{Z}(t) \) and \( \tilde{Z}(t) \) have the same fluid limit, by the Functional SLLN for martingales, we can establish the fluid limit of \( Z(t) \) as
\[
\overline{Z}(t) = \lim_{k \to \infty} \frac{Z(kt)}{k} = \lim_{k \to \infty} \frac{\tilde{Z}(kt)}{kt} = ct,
\]
almost surely, for any work-conserving scheduling policy. Moreover, as \( Z(t) \) and \( \tilde{Z}(t) \) have the same diffusion limit, we can establish the diffusion limit of \( Z(t) \) as
\[
\tilde{Z}(t) = \lim_{k \to \infty} \frac{Z(kt) - k\tilde{Z}(t)}{\sqrt{k}} = \lim_{k \to \infty} \frac{\tilde{Z}(kt) - k\tilde{Z}(t)}{\sqrt{k}},
\]
where the last equality follows directly from (38). Define \( \alpha_n := \lim_{k \to \infty} \mathbb{E}[\sum_{\ell=1}^{kn} I_n(t)] \). Note that \( \alpha_n \) is the average proportion of time in which client \( n \) is scheduled. For any work-conserving scheduling policy, we have \( \sum_{n=1}^{N} \alpha_n = 1 \). By the Functional CLT for martingales, we know that \( \tilde{Z}(t) \) is a Brownian motion with zero drift and variance \( \sigma^2 \), where \( \sigma^2 \) can be derived as [6, 15]:
\[
\sigma^2 = \lim_{t \to \infty} \frac{1}{t} \sum_{s=0}^{t-1} \mathbb{E} \left[ \tilde{Z}(s+1) - \tilde{Z}(s) - \varepsilon \right]^2 |\mathcal{H}_s] = 0,
\]
\[
= \lim_{t \to \infty} \frac{1}{t} \sum_{s=0}^{t-1} \mathbb{E} \left[ \tilde{Z}(s+1) - \tilde{Z}(s) \right]^2 |\mathcal{H}_s] - \varepsilon^2
\]
\[
= \frac{1}{\sum_{n=1}^{N} \alpha_n (1 - \frac{1}{\lambda_n \ell_n})^2 + \alpha_n (1 - \frac{1}{\lambda_n \ell_n})^2) - \varepsilon^2
\]
\[
\sigma^2 = \sum_{n=1}^{N} \alpha_n \left( \frac{1}{\lambda_n \ell_n} - 1 \right)
\]
\[
\sigma^2 = \sum_{n=1}^{N} \alpha_n \left( \frac{1}{\lambda_n \ell_n} - 1 \right)
\]
In other words, for any \( t, \Delta t \geq 0 \), we know \( (\tilde{Z}(t+\Delta t) - \tilde{Z}(t)) \) follows a Gaussian distribution with zero mean and variance \( \Delta t \cdot \sigma^2 \). Based on (28) and the above discussion on \( \tilde{Z}(t) \) and \( \tilde{Z}(t) \), we know that \( Z^*(t) \) is a Brownian motion with drift \( \varepsilon \) and variance \( \sigma^2 \).

Remark 4. Note that if there is only one client served by the AP (i.e., no scheduling involved), then \( Z^*(t) \) is simply \( Z^* \) with a pre-factor and hence can readily capture the playback behavior of this client. In this case, as we already know \( Z^*(t) \) is a Brownian motion, \( D_n(t) \) and \( U_n(t) \) of this client can be readily approximated by a two-sided reflected Brownian motion (with drift and variance explicitly characterized). For the case of multiple clients, as will be discussed in Section 5, we show that the playback process of each individual client can still be explicitly characterized by a two-sided reflected Brownian motion under the proposed policy.

Remark 5. By mimicking (7), we can define
\[
B^*(t; \ell_{tot}) := Z^*(t) - U^*(t; \ell_{tot}) + D^*(t; \ell_{tot}).
\]
By Theorems 1 and 2 and (29)-(30), we know \( B^*(t; \ell_{tot}) \in [0, \ell_{tot}/p] \) and that \( B^*(t; \ell_{tot}), U^*(t; \ell_{tot}), \) and \( D^*(t; \ell_{tot}) \) satisfy the same set of equations as (7)-(9). As we already know \( Z^*(t) \) is a Brownian motion, by [3, Proposition 5.1], we further know that \( B^*(t; \ell_{tot}) \) satisfies the ergodic property, i.e. \( B^*(t; \ell_{tot}) \) admits a unique stationary distribution. As \( D^*(t; \ell_{tot}) \) is directly related to the event that \( B^*(t; \ell_{tot}) \) hits zero, such ergodic property implies that \( D^*(t; \ell_{tot}) \) grows linearly with time at a fixed rate on average. Hence, we can define the long-term average growth rate of \( D^*(t; \ell_{tot}) \) as
\[
d^*(\ell_{tot}) := \lim_{t \to \infty} \frac{D^*(t; \ell_{tot})}{t}.
\]
Note that given \( \ell_{tot} > 0 \), both \( D^*(t; \ell_{tot}) \) and \( d^*(\ell_{tot}) \) are well-defined, regardless of the policy. Moreover, it is easy to check that \( d^*(\ell_{tot}) \) is a decreasing function of the total playback latency \( \ell_{tot} \). This fact also manifests the trade-off between playback latency and video interferences. In Section 6, we will further discuss some asymptotic results on quantifying the video interrupt rates.

As will be formally shown in Section 6, the asymptotic behavior of \( D_n^*(t) \) with respect to the playback latency is largely determined by the value of \( \varepsilon \). To prepare for the subsequent analysis, here we highlight the three major regimes regarding the value of \( \varepsilon \):

- **Heavy-traffic regime**: This regime represents the case where \( \varepsilon = 1 - \sum_{n=1}^{N} (\lambda_n / p) = 0 \). Therefore, \( Z^*(t) \) is a driftless Brownian motion with finite variance \( \sigma^2 \). Note that \( \lambda_n / p \) can be viewed as the equivalent workload of client \( n \) as \( 1 / p \) is the expected number of required transmissions for each successful packet delivery. Hence, this regime corresponds to the case where the total channel resource equals the total video bitrate.

- **Under-loaded regime**: In this regime, \( \varepsilon = 1 - \sum_{n=1}^{N} (\lambda_n / p) > 0 \), and therefore (28) suggests that \( Z^*(t) \) is a Brownian motion with positive drift. This regime corresponds to the case where the total channel resource is strictly larger than the total video bitrate. Therefore, it is intuitively feasible to have \( B_n(t) \) close to \( \lambda_n \ell_n \) for most of the time by properly scheduling each client based on its video bitrate and channel reliability. In Section 6, we will see that this effect also manifests itself in the fast-decaying behavior of \( D_n^*(t) \) with respect to the playback latency.

- **Over-loaded regime**: This regime corresponds to that \( \varepsilon < 0 \). If \( \varepsilon < 0 \), then there must exist one client \( n \) that suffers from \( B_n(t) = 0 \) and hence excessive video interruption for most of the time, regardless of the scheduling policy.

The over-loaded regime is generally not the case of interest in the design of scheduling policies. Therefore, in this paper we focus mainly on the heavy-traffic and under-loaded regimes, i.e. \( \sum_{n=1}^{N} \lambda_n / p \leq 1 \).

4.3 Capacity Region for QoE Under Brownian Approximation
Recall from Definition 2.1 that the capacity region for QoE is defined based on the feasible video interrupt rates \( \lim_{t \to \infty} D_n^*(t) / t \) under a playback latency budget. Moreover, recall from (25) that we propose to use \( D_n^*(t) \) to approximate the original processes \( D_n(t) \). Therefore, subsequently we proceed by considering the approximation \( \lim_{t \to \infty} D_n^*(t) / t \approx \lim_{t \to \infty} D_n^*(t) / t \) and thereby study the set of feasible vectors of \( \lim_{t \to \infty} D_n^*(t) / t \).
To quantify \( \lim_{t \to \infty} D_n^*(t)/t \), we propose to use \( D^*(t; \ell_{\text{tot}}) \) and the corresponding \( d^*(\ell_{\text{tot}}) \) defined in (45) as the reference measure for the following reasons: (i) as the distribution of \( Z^*(t) \) does not depend on the employed scheduling policy, by (29)-(30) we know that both \( D^*(t; \ell_{\text{tot}}) \) and \( U^*(t; \ell_{\text{tot}}) \) also have invariant distributions under a given \( \ell_{\text{tot}} \) across all scheduling policies; (ii) there is an inherent connection between \( D_n^*(t) \) and \( D^*(t; \ell_{\text{tot}}) \) based on the two-sided reflection mappings in (23)-(24) and (29)-(30).

To formally compare the two stochastic processes \( D^*(t; \ell_{\text{tot}}) \) and \( D_n^*(t) \), we first introduce the notion of stochastic ordering for stochastic processes as follows.

**Definition 4.1 (Stochastic Ordering [30]).** Let \( G_1 \) and \( G_2 \) be two real-valued random variables. We say that \( G_1 \leq_{st} G_2 \) if
\[
P[G_1 \geq x] \leq P[G_2 \geq x], \quad \forall x \in \mathbb{R}. \tag{46}
\]

Now we are ready to present an important property which connects \( D^*(t; \ell_{\text{tot}}) \) with \( D_n^*(t) \). Specifically, we show that the inequality in Theorem 3 still holds under the approximation as follows.

**Theorem 4.** Under any \( \ell_{\text{tot}} > 0 \) and any scheduling policy,
\[
D^*(t; \ell_{\text{tot}}) \leq \sum_{n=1}^{N} \frac{1}{P} D_n^*(t), \quad \forall t \geq 0. \tag{47}
\]

**Proof.** Due to the space limitation, the proof is provided in Appendix A.2. \( \square \)

**Remark 6.** To get some intuition of (47), consider a special case where \( \ell_n = \infty \), for all \( n \). This coincides with the on-demand video scenario, i.e. the AP already has the complete video for each client at time 0. In this degenerate case, (24) becomes \( U_n^*(t) = 0 \) and therefore (23) can be simplified as
\[
D_n^*(t) = \sup_{0 \leq \tau \leq t} (-Z^*_n(\tau))^+. \tag{48}
\]
Similarly, (30) becomes \( U^*(t) = 0 \) and (29) can be simplified as
\[
D^*(t; \ell_{\text{tot}}) = \sup_{0 \leq \tau \leq t} (-Z^*(\tau))^+. \tag{49}
\]
By combining (48)-(49), it is easy to verify that (47) indeed holds after applying the basic properties of supremum. Note that a similar result for this degenerate case (i.e. on-demand videos) has been derived in [15]. Different from [15], the proof of (47) for the general cases (i.e. finite playback latency \( \ell_n \)) requires more involved analysis due to the recursion in (14)-(15) and (29)-(30).

Based on Theorem 4, under the Brownian approximation, we can obtain a necessary condition of a feasible vector as follows.

**Theorem 5.** Let \( \delta = (\delta_0, \delta_1, \ldots, \delta_N) \) be a feasible vector under the Brownian approximation with \( \delta_0 > 0 \) and \( \delta_n \geq 0 \), for all \( n = 1, \ldots, N \). Then, \( \delta \) must satisfy \( \frac{1}{P} \sum_{n=1}^{N} \delta_n \geq d^*(\delta_0) \).

**Proof.** Recall from the beginning of Section 4.3 that under the Brownian approximation, the vector \( \delta \) is feasible if under the condition that \( \ell_{\text{tot}} \leq \delta_0, \limsup_{t \to \infty} D_n^*(t)/t \leq \delta_n \), for all \( n \). Given the fact that \( D_n^*(t) \) and \( D^*(t; \ell_{\text{tot}}) \) are non-decreasing processes in \( t \), we divide both sides of (47) by \( t \) and take the limit superior to get
\[
d^*(\ell_{\text{tot}}) = \lim_{t \to \infty} D^*(t; \ell_{\text{tot}}) / t \leq \limsup_{t \to \infty} \frac{1}{P} \sum_{n=1}^{N} D_n^*(t) / t \leq \frac{1}{P} \sum_{n=1}^{N} \delta_n. \tag{50}
\]
Since \( d^*(\cdot) \) is a decreasing function by Remark 5, we know (50) and that \( \ell_{\text{tot}} \leq \delta_0 \) clearly imply \( \frac{1}{P} \sum_{n=1}^{N} \delta_n \geq d^*(\delta_0) \). \( \square \)

### 5 A QoE-Optimal Scheduling Policy

In this section, we present a QoE-optimal scheduling policy for real-time video streams. Recall that in Section 4.1, we define the capacity region for QoE and provide a necessary condition of feasible vectors in Theorem 5. In this section, we further show that the condition provided in Theorem 5 is also sufficient.

#### 5.1 Scheduling Policy

To begin with, we formally present the weighted largest deficit policy (WLD) as follows.

**Weighted Largest Deficit Policy (WLD):**

Let \( \{\beta_n\}_{n=1}^{N} \) be the predetermined positive weight factors.

1. During initialization, the AP configures the playback latency of each client \( n \) as \( \ell_n = \frac{\beta_n}{\sum_{m=1}^{N} \beta_m} \ell_{\text{tot}} \).

2. In each time slot \( t \), the AP schedules the client \( n \) with the largest \( -\tilde{Z}_n(t)/\beta_n \), with ties broken arbitrarily.

**Remark 7.** Recall that \( \tilde{Z}_n(t) = A_n(t) + U_n(t) - \lambda_n^* t \). As the video bitrate \( \lambda_n^* \) is usually predetermined and can be treated as hyper-parameters, the WLD policy is able to make scheduling decisions based on \( A_n(t) \) and \( U_n(t) \), which can be updated based on the acknowledgments from the clients. Moreover, the WLD policy does not require any information about the channel reliability.

#### 5.2 Proof of QoE-Optimality

To show that WLD is QoE-optimal, we first present the following state-space collapse property.

**Theorem 6.** For any given weight vector \( \beta = (\beta_1, \ldots, \beta_N) \) with \( \beta_n > 0 \), for all \( n \), and for any \( \{\lambda_n\} \) and \( p \) such that \( \sum_n \lambda_n / p \leq 1 \), the WLD policy achieves
\[
\frac{1}{\beta_n} Z_n(t) = \frac{1}{\beta_m} Z_m(t), \tag{51}
\]
for all pairs \( n, m \). Moreover, we have
\[
Z_n^*(t) = \frac{p \beta_n}{\sum_{m=1}^{N} \beta_m} Z^*(t), \quad \forall n. \tag{52}
\]

**Proof.** We prove this result by constructing \( N \) auxiliary stochastic processes as follows:
\[
W_n(t) := -\frac{\tilde{Z}_n(t)}{\beta_n} + \frac{\sum_{m=1}^{N} \tilde{Z}_m(t)}{\sum_{m=1}^{N} \beta_m}, \tag{53}
\]
where the second term is a weighted sum of \( \tilde{Z}_m(t)/\beta_m \). Note that \( \sum_{n=1}^{N} \beta_n W_n(t) = 0 \). Without loss of generality, suppose that
Stochastic processes are positive recurrent. The Lyapunov function by the definition in (53). Moreover, (54) implies that $W(t) \geq 0$. Define the one-step difference of $W(t)$ as $\Delta W_n(t) := W_n(t+1) - W_n(t)$, for all $n$. For ease of notation, we also define the one-step difference of $Z_n(t)$ as $\Delta Z_n(t) := Z_n(t+1) - Z_n(t)$, for all $n$. Next, we consider a quadratic Lyapunov function and show that the auxiliary stochastic processes are positive recurrent. The Lyapunov function is defined as

$$L(t) = \sum_{n=1}^{N} \frac{\beta_n}{\sum_{m=1}^{N} \beta_m} W_n(t)^2.$$  

(55)

Recall that $\mathcal{H}_t$ denotes the system history up to time $t$. The conditional Lyapunov drift can be calculated as

$$\mathbb{E}[L(t+1) - L(t) | \mathcal{H}_t] \leq \sum_{n=1}^{N} \frac{\beta_n}{\sum_{m=1}^{N} \beta_m} W_n(t) \cdot \Delta W_n(t) + B_0,$$  

(56)

where $B_0$ is some finite constant since $|\Delta Z_n(t)| \leq 1$, for all $n$ and for all $t \geq 0$. Note that given the assumption of (54), client 1 is scheduled at time $t$ under the WLD policy. Therefore, we know $\Delta Z_1(t) = p - \lambda_1$ and $\Delta Z_2(t) = -\lambda_2$, for all $n \neq 1$. Recall that $\epsilon = 1 - \frac{\sum_{n=1}^{N} \lambda_n}{p}$. We can rewrite (56) as

$$\mathbb{E}[L(t+1) - L(t) | \mathcal{H}_t] \leq B_0 + \frac{\beta_1}{p} W_1(t) \left( p - \frac{\lambda_1}{\beta_1} + 1 - \frac{\sum_{m=1}^{N} \lambda_m}{\sum_{m=1}^{N} \beta_m} \right) + \sum_{n=2}^{N} \frac{\beta_n}{\sum_{m=1}^{N} \beta_m} W_n(t) \left( 1 - \frac{\lambda_n}{\beta_n} + 1 - \frac{\sum_{m=1}^{N} \lambda_m}{\sum_{m=1}^{N} \beta_m} \right)$$  

(57)

$$= B_0 - \epsilon W_1(t) + \sum_{n=2}^{N} \frac{\lambda_n}{\sum_{m=1}^{N} \beta_m} W_n(t) - \sum_{n=1}^{N} \frac{\epsilon \cdot \beta_n}{\sum_{m=1}^{N} \beta_m} W_n(t)$$  

(58)

$$= B_0 - \epsilon W_1(t) + \sum_{n=2}^{N} \frac{\lambda_n}{\sum_{m=1}^{N} \beta_m} (W_n(t) - W_1(t)) + \sum_{n=1}^{N} \frac{\epsilon \cdot \beta_n}{\sum_{m=1}^{N} \beta_m} (W_n(t) - W_1(t))$$  

(59)

$$= B_0 + \sum_{n=2}^{N} \frac{\lambda_n}{\sum_{m=1}^{N} \beta_m} (W_n(t) - W_1(t)) + \sum_{n=1}^{N} \frac{\epsilon \cdot \beta_n}{\sum_{m=1}^{N} \beta_m} (W_n(t) - W_1(t))$$  

(60)

$$= B_0 + \sum_{n=1}^{N} \frac{\lambda_n}{\sum_{m=1}^{N} \beta_m} + \frac{\epsilon \cdot \beta_n}{\sum_{m=1}^{N} \beta_m} \left( W_n(t) - W_1(t) \right)$$  

(61)

$$\leq B_0 + \frac{\lambda_N}{\sum_{m=1}^{N} \beta_m} + \frac{\epsilon \cdot \beta_N}{\sum_{m=1}^{N} \beta_m} \left( W_n(t) - W_1(t) \right).$$  

(62)

By the assumption that $W_n(t)$ are in decreasing order, we further know $\beta_1 W_1(t) + \sum_{n=2}^{N} \beta_n W_n(t) \leq \sum_{n=1}^{N} \beta_n W_n(t) = 0$ and hence $W_N(t) \leq -\frac{\beta_1}{\sum_{n=1}^{N} \beta_n} W_1(t)$. Therefore, we have

$$W_N(t) - W_1(t) \leq \left( 1 + \frac{\beta_1}{\sum_{n=2}^{N} \beta_n} \right) W_1(t).$$  

(63)

(65)

By (64) and (65), we know

$$\mathbb{E}[L(t+1) - L(t) | \mathcal{H}_t] \leq B_0 - B_1 W_1(t),$$  

(66)

where $B_1 = \left( 1 + \frac{\beta_1}{\sum_{n=2}^{N} \beta_n} \right) > 0$, for any $\epsilon \in [0, 1)$. By the Foster’s Criterion [28], we know that $(W_n(t))$ is positive recurrent, for all $n$, in both heavy-traffic and under-loaded regimes. We define the fluid limit and the diffusion limit of $W_n(t)$ as

$$\tilde{W}_n(t) := \lim_{k \to \infty} \frac{W_n(k)}{k},$$  

(67)

$$\tilde{W}_n(t) := \lim_{k \to \infty} \frac{W_n(t) - k \tilde{W}_n(t)}{\sqrt{k}}.$$  

(68)

Since $(W_n(t))$ is positive recurrent, we thereby know that $\tilde{W}_n(t) = 0$ and $\tilde{W}_n(t)$ is 0, almost surely, for all $n$. This implies that $\frac{Z_n(t)}{p} = \tilde{Z}_n(t)$, and hence $\frac{Z_n(t)}{p} = \frac{Z_n(t)}{p}$. For any pair of $n, m$. Moreover, as $\alpha^*(t) = \sum_{n=1}^{N} \frac{Z_n(t)}{p}$ by definition, we therefore have $\alpha^*(t) = \sum_{n=1}^{N} \frac{Z_n(t)}{p}$ for all $n$.

Recall from Section 4.2.2 that $\alpha^*(t)$ is a Brownian motion with drift $\epsilon$ and variance $\sigma^2$. By (52) in Theorem 6, we know $\alpha_m^*(t)$ is also a Brownian motion with positive drift $\epsilon_n$ and variance $\sigma_n^2$ under the WLD policy, where

$$\epsilon_n = \frac{\epsilon \cdot \beta_n}{\sum_{m=1}^{N} \beta_m},$$  

(69)

$$\sigma_n = \left( \frac{\beta_n}{\sum_{m=1}^{N} \beta_m} \right)^2 \sigma^2.$$  

(70)

By Theorem 6, we are ready to show that WLD policy achieves every point in the capacity region for QoE.

**Theorem 7.** For any feasible vector $\delta = (\delta_0, \delta_1, \ldots, \delta_N)$, under the WLD policy with $\beta_n/\delta_n = \beta_m/\delta_m$ for every pair $n, m$ and $\epsilon_n = \delta_0/\sum_{m=1}^{N} \beta_m$ for every client $n$, we have

$$\lim_{t \to \infty} \frac{D_n^*(t)}{t} = \frac{\delta_n p}{\sum_{m=1}^{N} \delta_m} d^*(\delta_0) \leq \delta_n.$$  

(71)

**Proof.** For ease of notation, define $\eta_n := \beta_n/\sum_{m=1}^{N} \beta_m$, for all $n$. By substituting (52) into (23)-(24), we have

$$D_n^*(t) = \sup_{0 \leq \tau \leq t} \left( -\eta_n Z^*(\tau) + U_n^*(\tau) \right),$$  

(72)

$$U_n^*(t) = \sup_{0 \leq \tau \leq t} \left( \eta_n Z^*(\tau) + D_n^*(\tau) - \epsilon_n \right).$$  

(73)

Since $\epsilon_n = \delta_0/\sum_{m=1}^{N} \beta_m$ for every $n$, we know $\epsilon_n = \delta_0$. By comparing (72)-(73) with (29)-(30), it is easy to verify that $D_n^*(t) =$
\[ \eta_n D^*(t; \delta_n) \text{ and } U_n^*(t) = \eta_n U^*(t; \delta_n) \] is the unique solution to (72)-(73). Therefore, for each \( n \), we can obtain the limit of \( D_n^*(t)/t \) as

\[
\lim_{t \to \infty} \frac{D_n^*(t)}{t} = \lim_{t \to \infty} \frac{1}{\sum_{m=1}^{N} \beta_m} D^*(t; \delta_n) = \frac{\delta_n}{\sum_{m=1}^{N} \delta_m} d^*(\delta_0) \leq \delta_n,
\]

for all \( n \), under the Brownian approximation, the vector \( \delta \) is feasible if and only if

\[ \frac{1}{\sum_{n=1}^{N} \delta_n} \geq d^*(\delta_0) \]

By Theorem 7, we know the necessary condition provided by Theorem 5 is also sufficient. We summarize this result as follows.

**Theorem 8.** For any vector \( \delta = (\delta_0, \delta_1, \ldots, \delta_N) \) with \( \delta_n > 0 \), for all \( n \), under the Brownian approximation, the vector \( \delta \) is feasible if and only if

\[ \frac{1}{\sum_{n=1}^{N} \delta_n} \geq d^*(\delta_0). \]

**Remark 8.** Note that in Theorem 8, we only consider the case where \( \delta_n > 0 \), for every client \( n \). Despite this, from an engineering perspective, we can get arbitrarily close to \( \delta_n = 0 \) by simply assigning an extremely small \( \beta_n \) to client \( n \).

### 5.3 Choosing \( \beta_n \) for WLD Policy: Examples of Network Utility Maximization for QoE

In this section, we discuss how to properly choose weights \( \{\beta_n\} \) for the WLD policy. In practice, the optimal \( \{\beta_n\} \) can be determined by solving a network utility maximization (NUM) problem, which encodes the relative importance of the QoE performance of the clients. To demonstrate the connection between NUM and WLD, we briefly discuss the following examples of NUM for QoE:

**Example 1 (Max-Min Fairness):** Suppose the AP follows the WLD policy with a predetermined latency budget \( \ell_{\text{tot}} \) and is configured to minimize a network-wide QoE penalty function \( f(D_n^*(t)) = \max_{1 \leq n \leq N} \left( \text{sup}_{t \to \infty} D_n^*(t)/t \right) \). By Theorem 7, this NUM can be converted into another equivalent optimization problem as

\[
\min_{\delta} \delta = \text{feasible, } \delta_n = \epsilon_n, \quad n = 1, \ldots, N
\]

Note that \( \delta_n \) is a standard NUM for max-min fairness with a constraint induced by the capacity region for QoE. Therefore, it is easy to verify that the optimal solution to \( \delta_n \) is \( \delta_n = \delta_n = (\epsilon_n - \epsilon_{\text{tot}})/N, \) for every \( n \). Moreover, by plugging this solution into Theorem 7, we know that \( f(D_n^*(t)) \) is minimized when \( \sup_{t \to \infty} D_n^*(t)/t = \frac{\epsilon_n}{\epsilon_{\text{tot}}}/N \), for all \( n \). Therefore, WLD can achieve the optimal QoE penalty by choosing \( \beta_n = \beta_m \), for any pair of \( n, m \), as suggested by Theorem 7. Moreover, under the total playback latency budget \( \ell_{\text{tot}} \), \( \beta_n = \beta_m \) suggests that we choose \( \ell_n = \ell_m \) (or equivalently \( \ell_n = \ell_{\text{tot}}/N \)).

**Example 2 (Weighted Sum of Monomial Penalty):** Let \( \zeta_n > 0 \) be the importance weight of each client \( n \). The AP follows WLD policy with a predetermined latency budget \( \ell_{\text{tot}} \) and is configured to minimize a network-wide QoE penalty function \( f(D_n^*(t)) = \sum_{n=1}^{N} \zeta_n \left( \text{sup}_{t \to \infty} D_n^*(t)/t \right)^{\kappa} \), with some constant \( \kappa > 1 \). By Theorem 7, we can convert this NUM into an equivalent problem:

\[
\min_{\delta} \delta = \text{feasible, } \delta_n = \epsilon_n, \quad n = 1, \ldots, N
\]

It is easy to verify that for any \( \kappa > 1 \), the optimal solution to (77) is

\[ \delta_n = \left( \frac{\zeta_n}{\ell_{\text{tot}}} \right)^\kappa \left( \frac{1}{\sum_{m=1}^{N} \zeta_m} \right)^\kappa, \]

for every \( n \). Again, by Theorem 7, WLD can achieve the optimal network utility by choosing \( \beta_n = \left( \frac{\zeta_n}{\ell_{\text{tot}}} \right)^\kappa \left( \frac{1}{\sum_{m=1}^{N} \zeta_m} \right)^\kappa \). Regarding the playback latency, WLD simply assigns \( \ell_n = \beta_n \cdot \ell_{\text{tot}}, \) for each \( n \).

Based on these two examples, we know that the WLD policy can be easily configured to solve a broad class of NUM problems for QoE given the flexibility provided by the WLD policy.

### 6 ASYMPTOTIC RESULTS WITH RESPECT TO PLAYBACK LATENCY

In this section, we discuss the asymptotic behavior of \( D_n^*(t)/t \) with respect to the playback latency under the WLD policy. Recall that in Remark 5, we discuss the ergodic property of the two-sided reflected Brownian motion. Based on Theorem 7, we know that the video interrupt rates under approximation (i.e. \( \lim_{t \to \infty} D_n^*(t)/t \)) exists and depends on the playback latency \( \ell_n \). To begin with, we consider the heavy-traffic regime, i.e. \( \sum_{n=1}^{N} \lambda_n/p = 1 \). The following theorem shows that the video interrupt rate is inversely proportional to the playback latency in heavy-traffic.

**Theorem 9.** In the heavy-traffic regime, under the WLD policy, we have

\[
\lim_{t \to \infty} D_n^*(t)/t = \left( \frac{\sigma_n^2}{2\zeta_n} \right), \quad \text{as } \ell_n \to \infty. \tag{78}
\]

**Proof.** This result can be directly obtained by plugging the variance of \( Z_n^*(1) \) into [3, Theorem 12.1].

Next, we turn to the under-loaded regime, where \( \sum_{n=1}^{N} \lambda_n/p < 1 \). The following theorem shows that the video interrupt rate under approximation decreases exponentially fast with the playback latency in the under-loaded regime.

**Theorem 10.** In the under-loaded regime, under the WLD policy, we have

\[
\lim_{t \to \infty} D_n^*(t)/t = c \exp \left( -\frac{2\eta_n}{\sigma_n^2} \ell_n \right), \quad \text{as } \ell_n \to \infty, \tag{79}
\]

where \( c \) is some constant that does not depend on \( \ell_n \).

**Proof.** By [3, Theorem 3.1], this result can be directly obtained by finding the root \( \gamma \) of the Lundberg equation \( \mathbb{E}[\exp(\gamma Z_n^*(1))] = 1 \). As \( Z_n^*(1) \) is a Gaussian random variable with mean \( \ell_n \) and variance \( \sigma_n^2 \) (defined in (69)-(70)), it is easy to verify that \( \gamma = -2\eta_n/\sigma_n^2 \). □

**Remark 9.** Note that a one-dimensional one-sided reflected Brownian motion with negative drift has a stationary distribution, which is actually exponential [7, Theorem 6.2]. In the under-loaded regime, as suggested by Theorem 10, a two-sided reflected Brownian motion also exhibits a similar behavior as that of the one-sided reflected counterpart.

### 7 NUMERICAL SIMULATIONS

In this section, we present the simulation results of the proposed policy. Throughout the simulations, we consider a network of one
AP and 5 video clients. All the simulation results presented below are the average of 50 simulation trials.

7.1 Accuracy of the Approximation

We start by evaluating the accuracy of the proposed approximation under the proposed WLD policy. We consider a fully-symmetric network of 5 video clients, where \( \ell_n = \ell_{tot}/5 \), for every \( n \). In this case, WLD shall choose \( \ell_n = 1/5 \), for every client. We consider three heavy-traffic scenarios with \( p = 1/2, 1/3, 5/7 \) and \( \lambda_n = 1/10, 1/15, 1/7 \), respectively. First, Figure 4(a) shows the total amount of video interruptions (i.e. \( \sum_{n=1}^{N} D_n(t) \)) under \( p = 0.5 \) and different playback latency budgets in the heavy-traffic regime. We observe that video interruptions grow roughly linearly with time, as suggested by Remark 5. To further verify the accuracy of the approximation, Figure 4(b)-(d) show the total video interrupt rates under different playback latency budgets and different channel reliabilities in the heavy-traffic regime. Note that both the x-axis and y-axis are in log scale. We also plot the theoretical estimates of the total video interrupt rates based on Theorem 9 (by (43), we know \( \sigma^2 = 1, 2 \), and 0.4 for \( p = 1/2, 1/3, \) and 5/7, respectively). It can be observed that the empirical rates are very close to the theoretical estimates, and the difference shrinks with the playback latency budget. This is consistent with the asymptotic results in Theorem 9.

Next, we turn to the under-loaded case. We consider three slightly under-loaded scenarios with \( p = 0.52, 0.3467, 0.7428 \) and \( \lambda = 1/10, 1/15, 1/7 \), respectively. In all three scenarios, \( \sum_{n=1}^{N} r_n/p = 25/26 \) (and hence \( e = 1 - \sum_n \lambda_n/p = 1/26 \)). Figure 5(a) shows the total video interrupt rates under different \( \ell_{tot} \) and channel reliabilities (note that the y-axis is in log scale and the x-axis is in linear scale). We can observe that the dependency of empirical rates on \( \ell_{tot} \) is roughly log-linear, as suggested by Theorem 10. To further verify the accuracy of the theoretical estimates provided by Theorem 10, Figure 5(b) plots the ratio between the empirical total interrupt rate and the asymptotic term in (79), i.e. \( \frac{\sum_{n=1}^{N} D_n(t)}{(N \exp(-2\pi \ell_n/\sigma^2))} \), under different channel reliabilities. We observe that under different \( \ell_{tot} \), this ratio stays at around 0.01, 0.005, and 0.05 under \( p = 0.52, 0.3467, \) and 0.7428, respectively. Hence, Figure 5(b) verifies the accuracy of the approximation in the under-loaded regime.

In summary, all the above results suggest that the approximation \( D_n(t) - D_{n,tot}(t) \) is rather accurate in both heavy-traffic and under-loaded regimes, even with small to moderate latency budgets.

7.2 Comparison With Other Policies

We evaluate the proposed WLD policy against four baseline policies, namely Weighted Random (WRand), Weighted Round Robin (WRR), Earliest Deadline First (EDF), and the Delivery-Based Largest-Debt-First (DBLDF). Under the WRand policy, in each time slot, the AP simply schedules each client \( n \) with probability \( \lambda_n/\sum_{m=1}^{N} \lambda_m \). Under the WRR policy, the AP groups multiple time slots into a frame and schedules the clients in a cyclic manner within each frame. Specifically, in each frame, each client \( n \) is scheduled for exactly \( K \lambda_n/\sum_{m=1}^{N} \lambda_m \) times, where \( K \) is chosen to be the smallest positive integer such that \( K \lambda_n/\sum_{m=1}^{N} \lambda_m \) is an integer, for all \( n \). Under the EDF policy, the AP schedules the video packet with the smallest absolute deadline among all the video packets in the AP-side buffers, with ties broken randomly. The EDF policy is widely used in real-time systems given its strong theoretical guarantee for deadline-constrained tasks [22]. Under DBLDF, the AP schedules the client with the largest delivery debt, which is defined as \( \lambda_n t - A_n(t) \). Different from WLD, DBLDF tracks only the delivery of data packets and is completely oblivious to the dummy packets. Note that the delivery-debt index was proposed and analyzed in [13] for the frame-synchronized real-time wireless networks. We evaluate the WLD policy as well as the four baseline policies in both heavy-traffic and under-loaded regimes.

To showcase the performance of the proposed policy, we start with the following heavy-traffic scenario: The 5 video clients are
Figure 6: QoE and video interruptions with \( p = 0.5 \), \( \ell_{\text{tot}} = 55 \), and a quadratic penalty function in the heavy-traffic regime.

Figure 7: QoE and video interruptions with \( p = 0.52 \), \( \ell_{\text{tot}} = 55 \), and a quadratic penalty function in the under-loaded regime.

8 RELATED WORK

In this section, we review the existing works that are most relevant to this paper. The first category includes the existing works of QoE-driven scheduling in video streaming. To optimize the QoE in video delivery, Joseph et al. [18] consider the joint optimization of network resource allocation and video quality adaption in scheduling and propose NOVA, an online scheduling algorithm that is proved to be asymptotically optimal. A similar formulation and discussion can be found in [12]. Cicalò et al. [8] consider the fairness in QoE and develop a resource allocation algorithm that maximizes the overall video quality under the QoE fairness constraint. Anand et
al. [1] extend the scope of QoE with consideration of mean flow delay; a scheduling policy that achieves asymptotically optimal delay-based QoE is proposed. In its follow-up paper [2], the non-linear relationship between a user’s QoE and flow delays as well as the balancing technique has been discussed. Li et al. [25] especially focus on the design of rate control and scheduling algorithm in multi-cast wireless networks. While these studies have discussed wireless scheduling in video streaming, the settings they consider are very different from ours. They consider that video streaming is on-demand, i.e., all data of the video to be transmitted are already available before the streaming process starts. In contrast, we address the scheduling problem in real-time video streaming, where video contents are continuously generated in real time. This leads to new research challenges such as the playback latency constraint, occurrence of video interruptions, and the dropping of packets that are not delivered within the required playback latency.

The second category consists of the research of scheduling algorithms for real-time wireless networks. In [13], Hou et al. propose a framework for optimizing the deadline-constrained wireless scheduling with delivery ratio requirements. In [17], Jaramillo et al. investigate the resource allocation fairness in wireless scheduling under strict packet deadlines. In [24], Li et al. specifically discuss the time-varying issues such as time-varying traffic and channel conditions in distributed real-time scheduling. In [20], Kim et al. consider the multi-cast scheduling problem for transmitting real-time flows under strict deadlines over unreliable wireless channels. In its follow-up work [19], the model is extended to account for delayed round-trip feedback. Based on the theoretical analysis established in [13] for a special traffic pattern (namely, the frame-synchronized setting), Deng et al. [9] extend the analysis to more general traffic patterns. The above works discuss real-time wireless scheduling, usually with an aim to optimize delivery ratios. Their objectives are different from the one in this paper, whereas our goal is to tackle the fundamental trade-off between video interruptions and the playback latency in real-time video delivery.

The formulation of the real-time video delivery problem in this paper is also closely related to the finite buffer capacity settings that have been widely used in many other contexts, e.g., packet switched networks [10], single-hop and multi-hop wireless networks [23, 33, 34], multi-cast wireless networks [26, 27], and transportation networks [11, 32]. Compared to the above prior works, our work has two salient features: First, our goal is to handle the critical trade-off between the playback latency and video interruptions, while the focus of those studies is to achieve throughput optimality. Second, we leverage Brownian approximation, which considers both the first-order (fluid limit) and second-order behavior (diffusion limit) and therefore can accurately capture the short-term performance of the scheduling policies, while the prior works focus on long-term average throughput performance.

Several very recent works have proposed the idea of utilizing Brownian approximation as a tool for analyzing problems in wireless scheduling [4, 5, 14–16]. Inspired by the above works, we propose to leverage Brownian approximation to analyze the real-time video delivery problem. Our model is different from those works in two aspects: First, we consider the two-sided reflection mapping in characterizing the real-time video playback processes, while in those works, only one-sided reflection mapping is taken into account. Second, the above prior works focus on network optimization in the on-demand scenarios, i.e. all data to be transmitted are available before transmission. In contrast, we consider the streaming scenarios where data is continuously generated in real time with a playback latency requirement. These features lead to new research challenges such as the playback latency constraint, video interruptions, and the dropping of packets that violate the playback latency requirements. Instead of using a simple adaptation of the existing methods for the on-demand scenarios, we present a novel framework to tackle these critical challenges in real-time video delivery.

9 CONCLUSION

This paper studies the critical trade-off between playback latency and video playback interruption, which are the two most critical QoE metrics for real-time video streaming. With the proposed analytical model and the Brownian approximation scheme, we study the fundamental limits of the latency-interruption trade-off and thereby design a QoE-optimal scheduling policy. Through both rigorous analysis and extensive simulations, we show that the proposed approximation framework can capture the original playback processes very accurately and hence thoroughly address the interplay between playback latency and video interruption for real-time video delivery.

REFERENCES

[1] Arjun Anand and Gustavo de Veciana. 2017. Measurement-based scheduler for multi-class QoE optimization in wireless networks. In IEEE INFOCOM 2017-IEEE Conference on Computer Communications. IEEE, 1–9.
[2] Arjun Anand and Gustavo de Veciana. 2018. A Whittle’s index based approach for QoE optimization in wireless networks. Proceedings of the ACM on Measurement and Analysis of Computing Systems 2, 1 (2018), 15.
[3] Lars Norvang Andersen, Søren Asmussen, Peter W Glynn, and Mats Pihlsgård. 2015. Levy processes with two-sided reflection. In Levy Matters V. Springer, 67–182.
[4] Hana Baili. 2018. Optimal scheduling and power allocation in wireless networks with heavy traffic: the infinite time horizon case. International Journal of Systems Science: Operations & Logistics 5, 3 (2018), 211–226.
[5] Hana Baili. 2019. Ergodic optimization of stochastic differential systems in wireless networks. Multidimensional Systems and Signal Processing 30, 1 (2019), 413–449.
[6] Bruce M Brown et al. 1971. Martingale central limit theorems. The Annals of Mathematical Statistics 42, 1 (1971), 59–66.
[7] Hong Chen and David D Yao. 2001. Fundamentals of Queueing Networks: Performance, Asymptotics, and Optimization. Vol. 46. Springer Science & Business Media.
[8] Sergio Cicalò, Nesrine Changuel, Velo Tralli, Bessem Sayadi, Frederic Faucheux, and Sylvaine Kerboruf. 2016. Improving QoE and fairness in HTTP adaptive streaming over LTE network. IEEE Transactions on Circuits and Systems for Video Technology 26, 12 (2016), 2284–2298.
[9] Lei Deng, Chih-Chun Wang, Minghua Chen, and Shizhen Zhao. 2017. Timely wireless flows with general traffic patterns: Capacity region and scheduling algorithms. IEEE/ACM Transactions on Networking 25, 6 (2017), 3475–3486.
[10] Paolo Giacone, Emilio Leonardi, and Devavrat Shah. 2007. Throughput region of finite-buffered networks. IEEE Transactions on Parallel and Distributed Systems 18, 2 (2007), 251–263.
[11] Jean Gregoire, Xiaojianjun Qian, Emilio Frazzoli, Arnaud De La Fortelle, and Tichakorn Wongpiromsarn. 2015. Capacity-aware backpressure traffic signal control. IEEE Transactions on Control of Network Systems 2, 2 (2015), 164–173.
[12] Lijun He and Guizhong Liu. 2018. QoE-driven joint optimization of segment adaptation and resource allocation for DASH clients over LTE networks. IEEE Transactions on Vehicular Technology 67, 11 (2018), 11035–11048.
[13] IH Hou, V Borkar, and PR Kumar. 2009. A Theory of QoS for Wireless. Proc. of IEEE INFOCOM (2009), 486–494.
[14] I-Hong Hou. 2016. On the modeling and optimization of short-term performance for real-time wireless networks. In Proc. of IEEE INFOCOM 1–9.
\[ \sup_{0 \leq t \leq T} \left( -Z(t) + \sum_{n=1}^{N} \sup_{0 \leq s \leq T} \left( \frac{Z_n(s) + U_n(s)}{p} - \ell_n \right) \right) \geq \sup_{0 \leq t \leq T} \left( -Z(t) + \sum_{n=1}^{N} \sup_{0 \leq s \leq T} \left( Z(s) + \frac{1}{p} D_n(s) - \frac{N}{n} \ell_n \right) \right), \]

where (80) follows from (14) and the fact that \( Z_n(0) = 0 \) and \( U_n(0) = 0 \), (81) follows from the definition of supremum, (82) is a direct result of (15), and (83) again follows directly from the definition of supremum. Now, we are ready to prove the theorem by contradiction. Let us fix one sample path and define

\[ t_s := \inf \left\{ t : \sum_{n=1}^{N} \frac{1}{p} D_n(t) < D(t) \right\}. \]

Suppose that \( t_s < \infty \), which implies that we have

\[ \sum_{n=1}^{N} \frac{1}{p} D_n(t_s) \geq D(t_s), \]

for any \( t \) with \( 0 \leq t \leq t_s - 1 \). At time \( t_s \), due to right continuity of \( D(t) \) and \( D_n(t) \), we have

\[ \sum_{n=1}^{N} \frac{1}{p} D_n(t_s) < D(t_s). \]

Since both \( D_n(t) \) and \( D(t) \) are non-decreasing, we must have

\[ D(t_s) - D(t_s - 1) > 0. \]

By the definition of \( D(t) \) and \( U(t) \) in (17)-(18), we also know that \( D(t) \) and \( U(t) \) cannot increase simultaneously. Therefore, we have

\[ U(t_s) = U(t_s - 1) = 0. \]

Next, we have

\[ D(t_s) = \sup_{0 \leq t \leq T} \left( -Z(t) + U(t) \right) \]

\[ = \max_{0 \leq t \leq T} \left( -Z(t) + U(t) - Z(t_s) + U(t_s - 1) \right) \]

\[ \leq -Z(t_s) + U(t_s - 1) \]

\[ = -Z(t_s) + \sup_{0 \leq t \leq t_s - 1} \left( Z(s) + D(s) - \sum_{n=1}^{N} \frac{\ell_n}{p} \right) \]

\[ \leq -Z(t_s) + \sum_{0 \leq t \leq t_s - 1} \left( Z(s) + \sum_{n=1}^{N} \frac{1}{p} D_n(s) - \frac{N}{n} \ell_n \right) \]

\[ \leq -Z(t_s) + \sum_{0 \leq t \leq t_s} \left( Z(s) + \sum_{n=1}^{N} \frac{1}{p} D_n(s) - \frac{N}{n} \ell_n \right) \]

\[ \leq \sup_{0 \leq t \leq T} \left( -Z(t) + \sum_{0 \leq s \leq T} \left( Z(s) + \sum_{n=1}^{N} \frac{1}{p} D_n(s) - \frac{N}{n} \ell_n \right) \right), \]

where (89) follows from (17) and the fact that \( U(0) = 0 \) and \( Z(0) = 0 \), (90) is a direct result of (88), (91) follows from (87), (92) follows from the definition of \( U(t) \), (93) holds due to (85), and (94)-(95) are obtained by taking supremum over a longer horizon. By (86) and
for all \( t \geq 0 \), for every sample path. Next, we consider the limits of \( H_n^{(k)}(t) \) and \( H^{(k)}(t) \). By letting \( k \to \infty \) in (98)-(99), we have
\[
\lim_{k \to \infty} H_n^{(k)}(t) = \hat{Z}_n(t) + tZ_n = Z_n(t),
\]
\[
\lim_{k \to \infty} H^{(k)}(t) = \hat{Z}(t) + ct = Z^*(t),
\]
almost surely. Since the two-sided reflection mapping is a continuous mapping \([31, \text{Section } 5.4]\), then by continuous mapping theorem \([31, \text{Theorem 3.4.3}]\) along with (23)-(24) and (29)-(30), we know that as \( k \to \infty \), \( D_n^{(k)}(t) \) and \( D^{(k)}(t) \) converge in distribution to \( D_n^*(t) \) and \( D^*(t; t_{\text{tot}}) \), respectively. Finally, by combining (105) and the convergence results of \( D_n^{(k)}(t) \) and \( D^{(k)}(t) \) as well as using the argument of the Baby Skorohod Theorem \([29, \text{Theorem } 8.3.2]\), we conclude that \( D^*(t; t_{\text{tot}}) \leq \sum_{n=1}^N \frac{1}{p} D_n^{(k)}(t) \). □

\section*{A.2 Proof of Theorem 4}

Proof. We prove this result by constructing a sequence of processes based on the scaling approach outlined in [31, Chapter 5.4] and leverage the continuous mapping theorem to establish this inequality \([31, \text{Theorem 3.4.3}]\). Recall that we suppose \( Z_n(t) \) can be written as \( \hat{Z}_n(t) = iZ_n(t) \) and we already have \( \hat{Z}(t) = et \) in (38). For all \( k \in \mathbb{N} \), define the following scaled processes
\[
H_n^{(k)}(t) := \frac{Z_n(\lfloor kt \rfloor) - \lfloor kt \rfloor Z_n}{\sqrt{k}},
\]
\[
H^{(k)}(t) := \frac{Z(\lfloor kt \rfloor) - \lfloor kt \rfloor e}{\sqrt{k}},
\]
Moreover, by the definition of \( Z(t) \), it is easy to verify that for all \( k \),
\[
H^{(k)}(t) = \sum_{n=1}^N \frac{1}{p} H_n^{(k)}(t).
\]
We can observe that both \( H_n^{(k)}(t) \) and \( H^{(k)}(t) \) are step functions and hence are right-continuous, for all \( k \). Next, by following the similar argument as Corollary 1, for all \( k \in \mathbb{N} \), define \( (D_n^{(k)}(t), U_n^{(k)}(t)) \) as
\[
D_n^{(k)}(t) = \sup_{0 \leq \tau \leq t} \left( -H_n^{(k)}(\tau) + U_n^{(k)}(\tau) \right)^+, \quad (101)
\]
\[
U_n^{(k)}(t) = \sup_{0 \leq \tau \leq t} \left( H_n^{(k)}(\tau) + D_n^{(k)}(\tau) - \ell_n \right)^+. \quad (102)
\]
Similarly, define \( (D^{(k)}(t), U^{(k)}(t)) \) as
\[
D^{(k)}(t) = \sup_{0 \leq \tau \leq t} \left( -H^{(k)}(\tau) + U^{(k)}(\tau) \right)^+, \quad (103)
\]
\[
U^{(k)}(t) = \sup_{0 \leq \tau \leq t} \left( H^{(k)}(\tau) + D^{(k)}(\tau) - \ell_{\text{tot}} \right)^+. \quad (104)
\]
Again, by Corollary 1, we know \( (D_n^{(k)}(t), U_n^{(k)}(t)) \) and \( (D^{(k)}(t), U^{(k)}(t)) \) are unique, given \( H_n^{(k)}(t) \) and \( H^{(k)}(t) \). Since \( H_n^{(k)}(t) \) and \( H^{(k)}(t) \) are step functions which change values only when \( kt \) is an integer, then by using the same argument as that in the proof of Theorem 3, we know that for all \( k \in \mathbb{N} \),
\[
D^{(k)}(t) \leq \sum_{n=1}^N \frac{1}{p} D_n^{(k)}(t), \quad (105)
\]