

A Proposed Adaptive Bitrate Scheme Based on Bandwidth Prediction Algorithm for Smoothly Video Streaming

Dr. Saba Qasim Jabbar*

College of Engineering University of Baghdad
Baghdad, Iraq
E-Mail: shura2007515@yahoo.com

Dr. Dheyaa Jasim Kadhim

College of Engineering University of Baghdad
Baghdad, Iraq
E-Mail: deya_naw@yahoo.com

ABSTRACT

A robust video-bitrate adaptive scheme at client-aspect plays a significant role in keeping a good quality of video streaming technology experience. Video quality affects the amount of time the video has turned off playing due to the unfilled buffer state. Therefore to maintain a video streaming continuously with smooth bandwidth fluctuation, a video buffer structure based on adapting the video bitrate is considered in this work. Initially, the video buffer structure is formulated as an optimal control-theoretic problem that combines both video bitrate and video buffer feedback signals. While protecting the video buffer occupancy from exceeding the limited operating level can provide continuous video streaming, it may also cause a video bitrate oscillation. So the video buffer structure is adjusted by adding two thresholds as operating points for overflow and underflow states to filter the impact of throughput fluctuation on video buffer occupancy level. Then a bandwidth prediction algorithm is proposed for enhancing the performance of video bitrate adaptation. This algorithm's work depends on the current video buffer level, video bitrate of the previous segment, and iterative throughput measurements to predict the best video bitrate for the next segment. Simulation results show that reserving a bandwidth margin is better in adapting the video bitrate under bandwidth variation and then reducing the risk of video playback freezing. Simulation results proved that the playback freezing happens two times: firstly, when there is no bandwidth margin used and secondly, when the bandwidth margin is high while smooth video bitrate is obtained with moderate value. The proposed scheme is compared with other two schemes such as smoothed throughput rate (STR) and Buffer Based Rate (BBR) in terms of prediction error, QoE preferences, buffer size, and startup delay time, then the proposed scheme outperforms these schemes in attaining smooth video bitrates and continuous video playback.

Keywords: Video streaming, video buffer, overflow state, underflow state, adaptive bitrate algorithm.

*Corresponding author

Peer review under the responsibility of University of Baghdad.

<https://doi.org/10.31026/j.eng.2021.01.08>

2520-3339 © 2019 University of Baghdad. Production and hosting by Journal of Engineering.

This is an open access article under the CC BY4 license <http://creativecommons.org/licenses/by/4.0/>.

Article received: 2/9/2020

Article accepted: 7/10/2020

Article published: 1/1/2021



مخطط مقترح لمعدل البيانات التكيفي مبني على اساس خوارزمية توقع النطاق الترددي لتدفق الفيديو على نحو سلس

أ.م.د. ضياء جاسم كاظم
كلية الهندسة - جامعة بغداد

م.د. صبا قاسم جبار
كلية الهندسة - جامعة بغداد

الخلاصة

يعتبر دور المخطط التكيفي لمعدل بيانات الفيديو مهم في الحفاظ على جودة استمرارية مشاهدة الفيديو على جانب المستخدم . تتأثر جودة الفيديو عندما يكون المخزن المؤقت فارغ من المقاطع الجاهزة للمشاهدة . وعليه يتم في هذا الشغل تشكيل البنية الاساسية للمخزن المؤقت وربطها مع معدل بيانات الفيديو المتدفقة ومستوى المقاطع الفيديوية في المخزن. ولتجنب التذبذب في معدل بيانات الفيديو يتم التحكم بالبنية الاساسية للمخزن باضافة حدود قصوى لمستوى فراغ وامتلاء المخزن بمقاطع الفيديو . ثم اقتراح خوارزمية لتوقع النطاق الترددي لتنزيل المقطع القادم للفيديو و لتحسين أداء المخطط التكيفي لمعدل بيانات الفيديو. يعتمد عمل هذه الخوارزمية على مستوى المخزن المؤقت للفيديو الحالي ، ومعدل بتات الفيديو للمقطع السابق ، وقياس النطاق الترددي للمقطع . هدف الخوارزمية اختيار معدل بيانات افضل للمقطع القادم وتحسين جودة الفيديو. لقد تم مقارنه مخططنا المقترح مع مخططين لدينا مثل (STR) و (BBR) من حيث خطأ التنبؤ ومفردات QoE وحجم المخزن المؤقت ووقت تأخير بدء التشغيل ، وكان المخطط المقترح يتفوق على هذه المخططات في تحقيق معدل بيانات فيديو سلسة وتشغيل الفيديو المستمر .

الكلمات الرئيسية: تدفق الفيديو ، مخزن الفيديو، الحد الأدنى ، الحد الاعلى، خوارزمية البيانات التكيفية.

1. INTRODUCTION

Video content constitutes a dominant fraction of Internet traffic today. Further, to ride the Internet bandwidth requirements, existing video streaming systems pick out adaptive and scalable video coding techniques to simplify transmission (Ni P, 2011). Dynamic adaptive streaming over HTTP (DASH) (Stoekhammer T, 2011) is an HTTP-based technique in which each video is divided into segments encoded at multiple rates. This policy supplies TCP fairness in the network and efficient delivery of MPEG media over HTTP. The video client requests a segment encoded at a particular video rate with an HTTP GET message, depending on the network conditions. Adaptive bitrate (ABR) algorithms are responsible for selecting the video bitrates of the segments (ur Rahman W, 2015). ABR algorithms strive to maximize video quality to upgrade the user's viewing experience. At this point, algorithms seek to increase viewing quality by meeting conflicting video quality goals in different environments. Maximizing the video rate would risk rebuffering, whereas minimizing rebuffering can lead to low video quality. In an environment with variable throughput, selecting a higher feasible video rate may lead to a high frequency of video-rate switches (Liu C, 2011).

With this system, the bitrate of the video segment can be altered (i.e., quality) in an adaptive manner to maintain continuous operation and avoid large latency to start and stop due to network congestion (Kadhim DJ, 2017). On the other hand, choosing the correct adaptive decision to suit the user's conviction in viewing is an open path to study present scaling techniques for video permit adaptation in either the spatial or temporal domain. Downloaded segments are stored for a period of time before playback starts so that a short-term mismatch between bitrate and available bandwidth can be accommodated and can mitigate video interruptions. (Jabbar SQ, 2017). In any case, the mismatch's persistence will cause the buffer to be empty from the video segments, which affects the user's viewing quality due to constantly stopping playback.

On the other hand, adaptive tuning is prepared to video source quickly. The user can watch videos at the maximum bit rate allowing time, available bandwidth, and device resources to vary. (Liu C, 2011).



This basic issue will also affect the customer's ability to avoid buffer operation when bandwidth is suddenly low (**Jabbar SQ, 2018**). In short, the problem stems from the discrete nature of video bitrates. This makes it impossible to always match the video bit rate with network bandwidth, resulting in no network bandwidth subscription. Not subscribing is usually paired with a customer's download patterns upon checkout. Time distances are a source of uncertainty for the customer to understand their fair share of network bandwidth properly, thus preventing the customer from making precise decisions for adapting the bitrates (**Sun Y, 2016**). To solve this basic limitation, a method based on the "predict-and-adapt" principle" is picked up.

In this context, the authors (**Li Z, 2014**) suggested that TCP data rate be input only when it becomes a good indicator of fair share bandwidth. This usually happens when the number of subscribers and absence periods are exceeded. In the outages, the agreed algorithm continually checks the network bandwidth by increasing the transmission rate, and if it is congested, it will prepare to return (**Kadhim DJ, 2015**). This algorithm dynamically determines the level of the version that matches the user's available bandwidth. No further processing is needed to adapt the video rate to variable bandwidth since this algorithm diminishes processing costs when the video is encoded. In contrast, this way's drawback is the increased storage requirements. A more approximate splitting characterizes the adaptation because video bit rates can only belong to a discrete set of levels.

In (**Miller K., 2012**), an adaptive algorithm for HTTP video streaming has been proposed. This algorithm consists of two stages: a fast phase to raise the occupancy of segments in the buffer to a known threshold value and a steady phase to prevent the privilege from returning to a low-flow state; hence the number of quality switches can be dominated. In the work of (**Jabbar SQ, 2018**), the suggested algorithm is used to choose the dynamic version bitrate based on the network conditions. Then it was implemented as a platform-independent software library, after which they evaluated the prototype applied in real-world scenarios and found that it worked remarkably even under challenging network conditions. Moreover, it displays a stable and fair process if it is shared by many clients (**Abdulredah SH, 2020**).

In this work, video streaming will be maintained continuously with smooth bandwidth fluctuation, so a video buffer structure based on adapting the video bitrate is considered in this research. Initially, the video buffer structure is formulated as an optimal control-theoretic problem that combines both video bitrate and video buffer feedback signals. While protecting the video buffer occupancy from exceeding the limited operating level can provide continuous video streaming, it may also cause a video bitrate oscillation. So the video buffer structure is adjusted by adding two thresholds as operating points for overflow and underflow states to filter the impact of throughput fluctuation on video buffer occupancy level. Then a bandwidth prediction algorithm is proposed for enhancing the performance of video bitrate adaptation. This algorithm's work depends on the current video buffer level, video bitrate of the previous segment, and iterative throughput measurements to predict the best video bitrate for the next segment.

2. VIDEO STREAMING MODEL

In the DASH system, the original video is divided into miniature clips and are encoded at different levels in order to suit the network conditions of the user side and stored on server with $\{V_{R1} < V_{R2} < \dots < V_{Rm}\} \in V$ where V is the set of these rates and m different versions. The m encoded versions are further segmented into N_t individual segments, which can be decoded separately. These segments are common of equal and steady τ second duration. So the total length of the video is $N_t \tau$. The customer plays an essential role in altering the rate by choosing



and ordering the appropriate video version for each video clip. The rates for different video rendering levels are accessed to a customer by the server in the Media Presentation Data (MPD) stage. The size of i^{th} segment is $V_{Ri}\tau$ frames. The bitrate level of the already downloaded segment is notated as $L(i) \in [1, m]$. The client starts to download i^{th} segment at t_s^i and gets the last bits of the segment at t_r^i . If T_i is the network throughput during the period of downloading segment i^{th} . Assuming the video player waits for w_i (i.e., w is a short time not lead to interrupt the video) and starts to download the i^{th} segment at a time t_s^i . Then

$$\int_{t_s^i}^{t_r^i} T_i(t) dt = V_{Ri} \tau \tag{1}$$

The average throughput for download i^{th} segment is given by:

$$\tilde{T}_i = \frac{1}{N_i} \sum_{i=1}^{N_i} \frac{V_{Ri} \tau}{t_r^i - t_s^i} \tag{2}$$

The needed time for i^{th} segment to be downloaded is given by:

$$t_s^i = t_s^{i-1} + \frac{V_{R(i-1)} \tau}{\tilde{T}_{i-1}} + w_i \tag{3}$$

Where \tilde{T}_{i-1} is the average throughput during the downloaded of $(i-1)^{th}$ segment. Since video segments have different video bitrate levels in the DASH system, there is no relation between video buffer size and video buffer time. And then, to directly measure the buffer's length, the video buffer time is used, and the buffer process is mentioned in terms of time as $B(t)$. The video buffer behavior after the downloaded i^{th} segment is written as:

$$\begin{aligned} B(t_r^i) &= B(t_s^i) + \tau - (t_r^i - t_s^i) \\ &= B(t_s^i) + \tau \left(1 - \frac{V_{Ri}}{\tilde{T}_i} \right) \end{aligned} \tag{4}$$

where the consumed time is $(t_r^i - t_s^i)$ for i^{th} segment. The video buffer rate of change is the first derivative of $B(t)$ as:

$$B'(t) = \frac{B(t_r^i) - B(t_s^i)}{(t_r^i - t_s^i)} = \frac{\tilde{T}_i}{V_{Ri}} - 1, \quad t \in (t_s^i, t_r^i] \tag{5}$$

From Eq. (3) if the picked video level is more than \tilde{T}_i , the video buffer level decreases, and playback freezing status occurs; if the selected video level is lower, than \tilde{T}_i the video buffer level faces an overflow status. Hence the adaptation process for video bitrate should take into account these two statuses. To avoid buffer underflow, the video bitrate should satisfy the following condition to downloading i^{th} :



$$V_{Ri} \leq \tilde{T}_i + \frac{\tilde{T}_i}{\tau} B(t_i^s) = V_{Ri}^U \tag{6}$$

On the other side, to avoid buffer overflow, the video bitrate should satisfy the following condition:

$$V_{Ri} \geq \tilde{T}_i + \frac{\tilde{T}_i}{\tau} (B(t_i^s) - B_{max}) = V_{Ri}^L \tag{7}$$

where B_{max} is the maximum buffer size. V_{Ri}^U and V_{Ri}^L represent the upper bound and lower bound of video bitrate (V_R), respectively. However, if the lower bound is higher than the highest V_h video version or the upper bound is lower than the lowest V_L video version. There is no video bit rate in V that meets the inequality limitations (6) and (7).

3. QOE IMPROVEMENT PROBLEM

The main goal of video bitrate adaptation is to enhance the video quality on the user side. This paper tries to provide a flexible QoE formula, an active area of research (**Dobrian F, 2011 Aug 15**). The key metrics that impact QoE are:

1. Average Video bitrate (V_{Ravg}): The average video bitrate over all segments is equal to:

$$V_{Ravg} = \frac{1}{N_t} \sum_{i=1}^{N_t} (V_{Ri}) \tag{8}$$

2. Average video bitrate variations (D_{var}): This metric follows the magnitude of the changes in the video bitrate from one segment to another and is given as:

$$D_{var} = \frac{1}{N_t} \sum_{i=1}^{N_t} |V_{Ravg}(V_{Ri}) - V_{Ravg}(V_{Ri-1})| \tag{9}$$

3. Interruption duration (D_{int}): Stoppage in video occurs if the given bitrate version is above average throughput while downloading the meant segment as shown in expression (6). So, the total stoppage period for the i^{th} clip is:

$$D_{int} = \sum_{i=1}^{N_t} \left(\frac{V_{Ri} \tau}{\tilde{T}_i} - B(t_i) \right) \tag{10}$$

4. The startup delay time (ST): This factor has significant impacts on QoE, and the user may give up the video request if it takes a long time to start, according to (**Liu Y, 2013 Dec 12**), defining the startup delay time as:

$$ST = 3.2 \left(w_{(i-1)} + \frac{V_{Ri} \tau}{\tilde{T}_i} \right) \tag{11}$$

And $w_{(i-1)}$ as:

$$w_{(i-1)} = \left(\left(B(t_{i-1}) - \frac{V_{R(i-1)} \tau}{\tilde{T}_{i-1}} \right)_+ + \tau - B_{max} \right)_+ \tag{12}$$



where $(x)_+ = \max\{x, 0\}$ for ensuring the term is never being negative value. Accordingly, the QoE can be formulated as a weighted sum of the metrics mentioned above and is given by:

$$QoE = V_{Ravg} - \alpha_r D_{var} - \alpha_{int} D_{int} - \alpha_s ST \tag{13}$$

where α_r , α_{int} , and α_s are positive weighted parameters for average video bitrate variation, interruption duration, and startup time, respectively. For more clarity, a small α_r means the user does not deal with bitrate variation, while a large value means more work must do to reach a smooth video buffer level. The same thing for other parameters (i.e., α_{int} , α_s). To improve the video quality and ensure smooth bitrate adaptation, the main problem is defined as:

$$\begin{aligned} \max \quad & QoE \\ \text{s.t.} \quad & t_s^i = t_s^{i-1} + \frac{V_{R(i-1)}\tau}{\tilde{T}_{i-1}} + w_i \\ & B(t_r^i) = B(t_s^i) + \tau - (t_r^i - t_s^i) \\ & V_{Ri} \geq \max(V_{Ri}^L, V_l) \ \& \ V_{Ri} \leq \min(V_{Ri}^U, V_h) \\ & B(t_1) = ST, \ B(t_i) \in [0, B_{max}], \ \forall i \in [1, \dots, N_i] \end{aligned} \tag{14}$$

To maximally use the bandwidth and ensure improving video quality, constraints in (14) must be satisfied. In any case, the problem of predicting the bandwidth remains complex because sometimes there is a mistake in the prediction, and the conditions in (6) and (7) related to the overflow/underflow in the buffer can be inaccurate. This leads to fluctuations in the content level of the buffer due to changes in the bandwidth. In the following annotation, the proposed method is explained as a solution to this problem.

4. PROPOSED ADAPTIVE BITRATE SCHEME

Generally, preventing underflow/overflow in buffer level is simple for any adaptive algorithm to do it. But avoiding bitrate oscillation is not easy. In this section, two thresholds are added as the maximum and minimum level for the segments in the buffer and filter the effect of changing the buffer occupancy on the bit rate selection, and then analyzing the adaptive bitrate system. Also, a bandwidth prediction algorithm is proposed to enhance the performance of bitrate adaptation. Moreover, a small positive/negative bandwidth margin is conserved to repay the bandwidth prediction fault as explained below.

4.1 Video Buffer Modification Scheme

Firstly, the video buffer is modified by adding two thresholds b_{min} and b_{max} as the main operating points for controlling the video bitrate oscillation and stabilizing the buffer occupancy level. In detail, if the current video buffer of the last downloaded segment between the two thresholds (i.e. $b_{min} \leq B(t_s^i) \leq b_{max}$), in this case, the current bitrate version is kept the same as the previous requested version. i.e. $V_{Ri} = V_{Ri-1}$.

When the current video buffer occupancy is less than b_{min} i.e., $B(t_s^i) < b_{min}$ the operating threshold b_{min} is used to choose the bitrate version that pulls the current video buffer level until b_{min} . To



prevent underflow, the current video buffer must satisfy the following condition $B(t_r) \geq b_{\min}$ at the end of downloading i^{th} segment, hence:

$$V_{Ri} \leq \tilde{T}_i + \frac{\tilde{T}_i}{\tau} (B(t_i^s) - b_{\min}) \tag{15}$$

In this case, the video bitrate should be selected from a high version that satisfies expression (15) to make full use of the bandwidth while guaranteeing no buffer underflow happens. On the other side, if the current video buffer occupancy is more than b_{\max} i.e., the operating threshold b_{\max} is used to choose the bitrate version that pulls the current video buffer level until b_{\max} ensures no video buffer overflow occurs. The video bitrate must satisfy the following condition:

$$V_{Ri} \geq \tilde{T}_i + \frac{\tilde{T}_i}{\tau} (B(t_i^s) - b_{\max}) \tag{16}$$

The video bitrate should be select from the lowest version to satisfy Eq. (16) since any video version that fulfills the above condition will degrade the video buffer and ensure no buffer overflow. According to the modification in video buffer, the adaptive system for controlling the video bitrate can be represented as:

$$V_{Ri} = \left\{ \begin{array}{l} \tilde{T}_i + \frac{\tilde{T}_i}{\tau} (B(t) - b_x) \\ B'(t) = \frac{\tilde{T}_i}{V_{Ri}} - 1 \\ b_x = \left\{ \begin{array}{ll} b_{\min} & \text{if } B(t) < b_{\min} \\ b_{\max} & \text{if } B(t) > b_{\max} \end{array} \right\} \\ V_{Ri-1} & \text{if } b_{\min} \leq B(t) \leq b_{\max} \end{array} \right. \tag{17}$$

Eq. (17) is a non-linear style and unsuitable for analysis and implementation. For a linear system, the following procedure is done.

Let $B'(t) = 0$ around the operating point b_x the corresponding video bitrate be V_{Rx} , then:

$$B'(t) = 0 \Rightarrow V_{Rx} = \tilde{T}_i \tag{18}$$

The following are the output equations obtained for the entire system with feedback signals in term of bitrate and video buffer:

$$\left\{ \begin{array}{l} \delta(V_{Ri}) = \frac{\tilde{T}_i}{\tau} \delta(B(t)) \\ \delta(B'(t)) = -\frac{1}{\tilde{T}_i} \delta(V_{Ri}) \end{array} \right. \tag{19}$$

where $\delta(V_{Ri}) = V_{Ri} - V_{Rx}$ and $\delta(B'(t)) = B'(t) - b'_x$ Applying the Laplace transform to the differential equations, With some manipulation, the transfer function is gotten as:

$$H(s) = \frac{1}{\tau s + 1} \tag{20}$$

The system's equilibrium state is stable since the system's pole is $s_p = -\frac{1}{\tau} > 0$ the pole is located in the left phase plane.



4.2 Bandwidth Prediction Algorithm

In this subsection, a bandwidth prediction algorithm is proposed to stable and improve the performance of bitrate adaptation. The algorithm's workflow is described below, which can optimally combine both bandwidth prediction and buffer occupancy.

4.2.1 The basic Work of Algorithm

The algorithm consists of two phases the startup phase and the stable phase. Essentially the algorithm selects the video bitrate V_{Ri} through looking N steps ahead (i.e., N is a small duration of about ten seconds). Network conditions are stable for a short duration of time so, it is possible to have accurate bandwidth prediction for a short duration to the future (Zhang Y, 2001 Nov 1). Then the problem of improving the video quality of experience is solved with bandwidth prediction $\hat{T}_i(t) \in \{t_i, t_{i+N}\}$. Initially, feedback information is dependent for the first bitrate and the improvement process is repeated at each step i . At each iteration, the video player keeps a duration from i^{th} segment to $(i+N-1)^{\text{th}}$ and performs the three main steps in the prediction algorithm:

- a. **Expectation:** This part responsible for predicting the bandwidth for the next N segments $\hat{T}_{(t_i, t_{i+N})}$ using information about bandwidth predictors.
- b. **Improvement:** It is the main part of this algorithm. Video bitrate V_{Ri} is obtained based on the current video buffer occupancy, video bitrate of last downloaded segment V_{Ri-1} , and bandwidth prediction. Here it depends on which phase the video player is to find the optimal V_{Ri} for the next segment. In stable phase $V_{Ri} = f(V_{Ri-1}, B(t^i), \hat{T}_{(t_i, t_{i+N})})$ and implemented by solving the QoE expression (14). In the startup phase, the startup delay time ST is also considered $[V_{Ri}, ST] = f(V_{Ri-1}, B(t^i), \hat{T}_{(t_i, t_{i+N})})$ and implemented by solving the QoE expression (14).
- c. **Implement:** Start downloading i^{th} clip with V_{Ri} and move N proceedings forward. If the customer's player is at the startup phase, wait ST before starting the first clip.

The following steps show the implementation of the bandwidth prediction algorithm.

```

Initialize
For i = 1 to  $N_t$  do
  If startup phase = true
  then
  
```




$\hat{T}_{(t_i, t_{i+N})} = BW \text{ Pred}(T_{t_1, t_i})$
 $[V_{Ri}, ST] = f(V_{Ri-1}, B(t^i), \hat{T}_{(t_i, t_{i+N})})$
 after ST (seconds) begin to playback the video
 Else if *stable phase = true* then
 $\hat{T}_{(t_i, t_{i+N})} = BW \text{ Pred}(T_{t_1, t_i})$
 $V_{Ri} = f(V_{Ri-1}, B(t^i), \hat{T}_{(t_i, t_{i+N})})$
 end if
 Output
 download i^{th} segment with video bitrate V_{Ri}
 end for

4.2.2 Improve Bandwidth Prediction Algorithm

The above algorithm basically assumes the existence of an accurate bandwidth predictor. However, this accuracy may not be found in reality, especially with the network's poor status, for example, in cellular networks or at peak times when the Internet is crowded. Furthermore, the risk of freezing during video playing or buffer overflow and with no bandwidth margin comes with this issue since prediction error is inescapable. The improved bandwidth prediction algorithm enhances QoE's worst case assuming the actual bandwidth can take any value in a range $[\hat{T}^L, \hat{T}^H]$ where \hat{T}^L, \hat{T}^H the lower bound is and the higher bound of the bandwidth. The algorithm solves the following problem:

$$\max_{V_{Ri} \dots V_{Ri+N-1}} \min_{T \in [\hat{T}^L, \hat{T}^H]} QoE \tag{21}$$

s.t *constrains of expression (14)*

To obtain the video bitrate of i^{th} segment as:

$$V_{Ri} = f(V_{Ri-1}, B(t^i), [\hat{T}^L, \hat{T}^H]) \tag{22}$$

By taking the lowest value of the bandwidth as the input to the improved algorithm, to repay the error of bandwidth prediction and video buffer overflow / empty, a small positive / negative bandwidth margin is used, so:

$$V_{Ri} = f(V_{Ri-1}, B(t^i), \rho \hat{T}^L) \tag{23}$$

The worst case of QoE takes place when the bandwidth is at its lower bound \hat{T}^L . According to expressions (13) and (14), the QoE can be written as the sum of 3 main elements (f1: average video bitrate level, f2: average bitrate variation, and f3: total interruption duration), only the interruption duration element depends on bandwidth T, thus:



$$\begin{aligned}
 & \max_{V_R} \min_{T \in [T^L, T^H]} QoE \\
 & \triangleq \max_{V_R} \left(f_1(V_R) - \alpha_r f_2(V_R) - \max_{T \in [T^L, T^H]} \alpha_{in} f_3(V_R, T) \right) \\
 & \triangleq \max_{V_R} QoE(V_R, T^L)
 \end{aligned} \tag{24}$$

So, any decrease in throughput T will lead to high interruption duration; the minimum QoE is achieved at $T = T^L$. In simulation results, the improvement in the prediction algorithm is verified through changing the value of bandwidth margin. It explains the effect of prediction error on video bitrate and video buffer occupancy.

5. SIMULATION RESULTS

Here, the Adaptive Video Bitrate Scheme is appreciated based on bandwidth prediction at changeable network status. H.264/ AVC codec is used for encoding video, and five versions are prepared at the server $V = \{400kbps, 800kbps, 1200kbps, 1600kbps, 2500kbps\}$. The initial video buffer time is set to 40 seconds. The default values of b_{min} , b_{max} , τ , B_{max} , and ρ are 10, 70, 10, 80, and 3, respectively. The performance of our scheme is compared with other schemes, the smoothed throughput rate (STR) (Liu C, 2011 Feb 23) and Buffer Based Rate adaptation (BBR) schemes as in (Huang TY, 2015 Feb 25). For bandwidth trace and prediction, a combination of existing datasets for bandwidth measurements is used (ZTI Telecom).

Fig.1 shows the bandwidth margin's role in controlling the video buffer occupancy since underflow or overflow due to inaccuracy in bandwidth prediction. From the figure, the video freezing occurs twice, once in case there is no margin with bandwidth, while in case rising the margin value the stopping within playback video clips vanishes as from Fig.1(e) to Fig.1(g). Video bitrate becomes more smooth when $\rho=3$ than when $\rho =1$. The freezing in video playback appears again when the bandwidth margin is high, and the video bitrate oscillates as Fig.1(d), Fig.1(h). The huge space between the adjusted bandwidth and practical bandwidth is why this issue cannot be used for suitable video bitrate adaptation. While reserving appropriate bandwidth margins produces a relatively smooth video bitrate, some bitrate ripples as when $t = 2350s$ short term oscillations. The video buffer thresholds b_{min} and b_{max} are fixed.

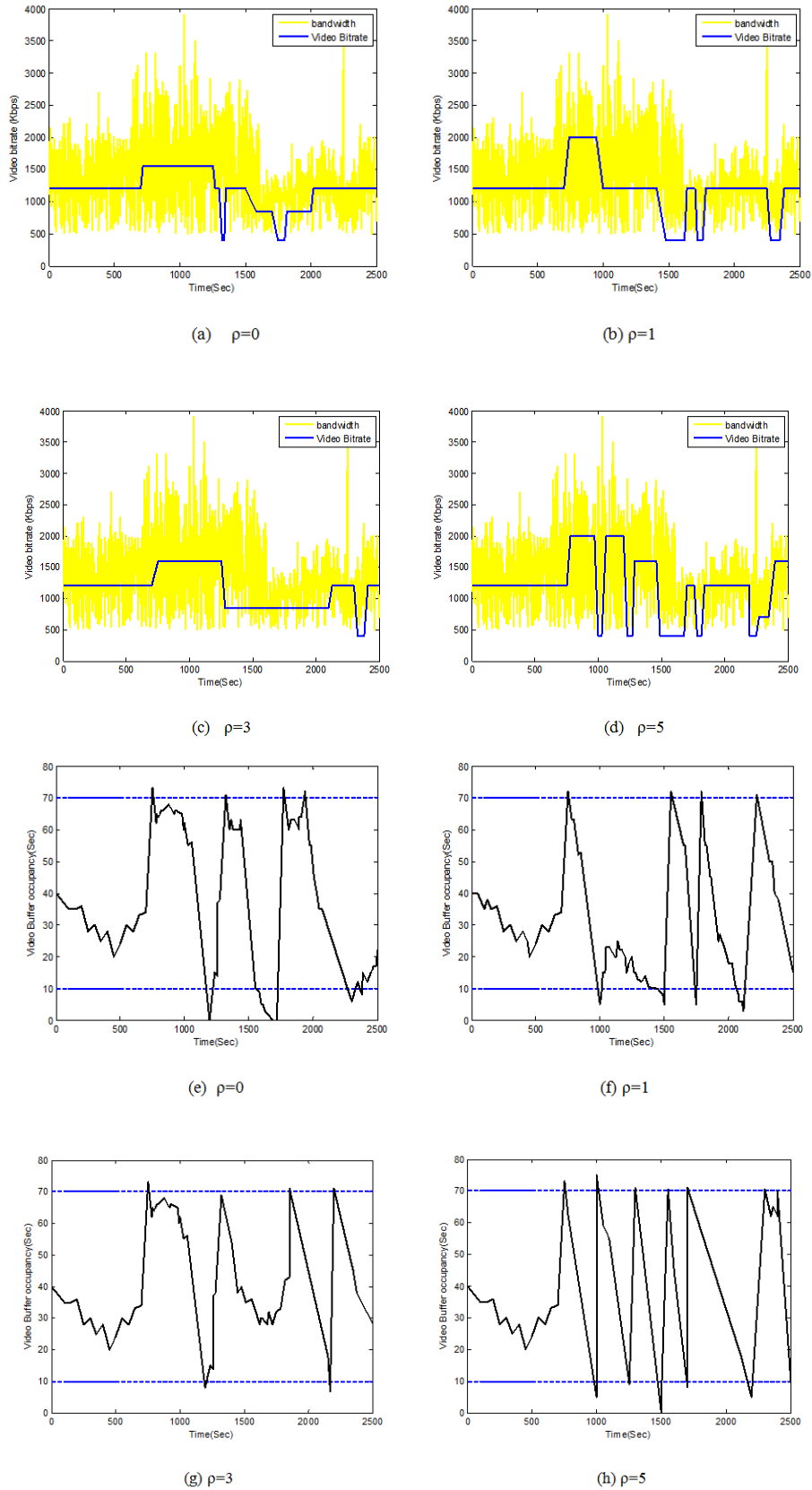


Figure 1. The proposed scheme under different bandwidth margins.

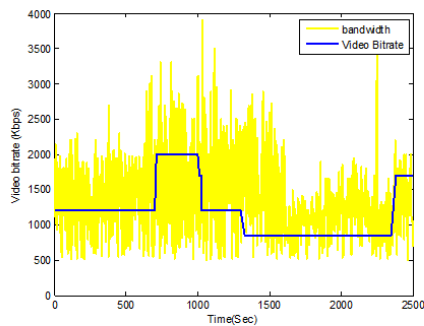


From Fig 2, when the timing space between b_{min} , b_{max} increases (i.e., b_{min} low and b_{max} high), the hazard of underflow/ overflow for buffer content increases, this is obvious in Fig.2(e). On the other side, when the timing space decreases (i.e., b_{min} increases and b_{max} decreases), the video bitrate oscillates drastically as in Fig.2(c). But when $b_{min} = 10$ and $b_{max} = 70$, the achievement turns better in terms of bitrate of video and buffer contents. Then the adjustment of b_{min} and b_{max} is dynamically by adding a change in video buffer expression (4) to satisfy the constraints in expressions (15) and (16) for video bitrates since there is some ripples in video bitrate when b_{min} , b_{max} are fixed. The condition for b_{min} , b_{max} be:

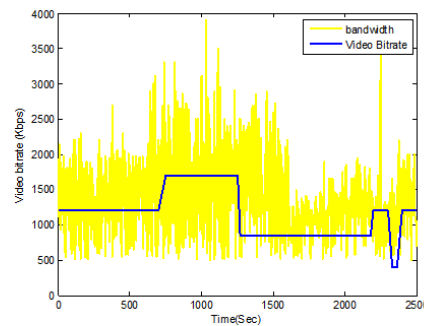
$$b_{min} \leq B(t_s^i) + \tau \left(1 - \frac{\beta_1 V_l}{T_i} \right) \quad \beta_1 \geq 1 \tag{25}$$

$$b_{max} \geq B(t_s^i) + \tau \left(1 - \frac{\beta_2 V_h}{T_i} \right) \quad \beta_2 \leq 1 \tag{26}$$

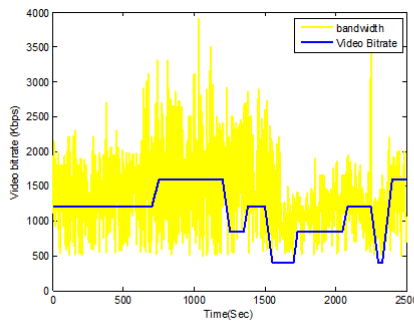
Where β_1, β_2 are weighted conservative parameters. From this figure, with dynamic thresholds, the achievement of bitrate is much better than static values. The diagrams in Fig.2 (d) and Fig.2 (h) show that freezing in playback and video buffer underflow/overflow never occurs.



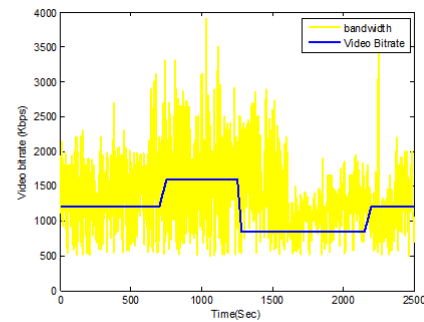
(a) $b_{min}=5, b_{max}=75$



(b) $b_{min}=10, b_{max}=70$



(c) $b_{min}=15, b_{max}=65$



(d) dynamic

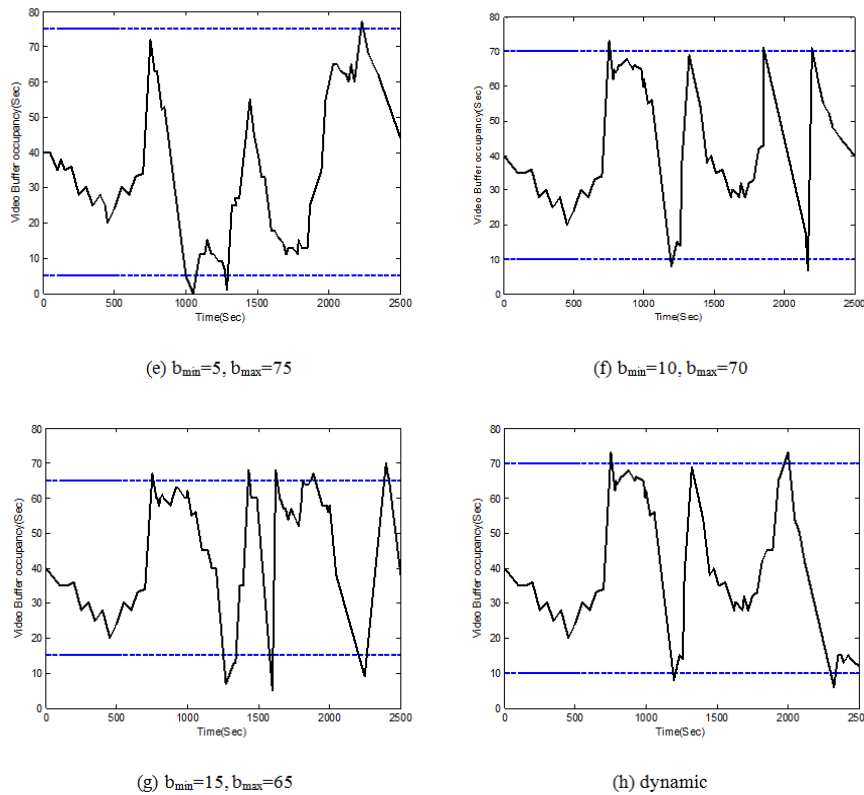


Figure 2. The proposed scheme under different video buffer thresholds.

A comparison between the proposed scheme with other schemes is made, the smoothed throughput rate (STR) and Buffer Based Rate adaptation (BBR) schemes. At time t_s^i when the bitrate of i^{th} segment is required, the simulation requests the proposed scheme based on bandwidth prediction algorithm to get V_{Ri} . The comparison is made in terms of prediction error, QoE favorites, and video buffer size, and startup delay time. Here a normalized QoE metric (QoE-N) is defined as the ratio between the online QoE achieved by proposed scheme with the selected video bitrate V_R based on current bandwidth predictions and the maximum QoE that can be achieved with perfect knowledge of future bandwidths over the entire duration.

Fig 3, examines the effect of prediction error on video bit rate conditioning. The mean level of error is used to describe the bandwidth predictor's performance and model the prediction output as a combination of real bandwidth with added random noise according to the average error level. As expected, BBR is unaffected as it does not use any transmission information. When the bandwidth predictions are accurate, the proposed system has a greater advantage over the BBR algorithm. However, with the prediction error exceeding 25%, which is the worst case, the improvement we proposed in improving the forecasted bandwidth may make our scheme less affected by the



prediction error because it considers the potential error and increases the worst-case for the quality of the experience. In contrast, STR performance decreases as prediction error rises .

The performance of the schemes under 3 sets of QoE weights, “Balanced” ($\alpha_r = 1; \alpha_{int} = \alpha_s = 3000$), “Avoid variation” ($\alpha_r = 3; \alpha_{int} = \alpha_s = 3000$), “Avoid interruption” ($\alpha_r = 1; \alpha_{int} = \alpha_s = 6000$) is compared.

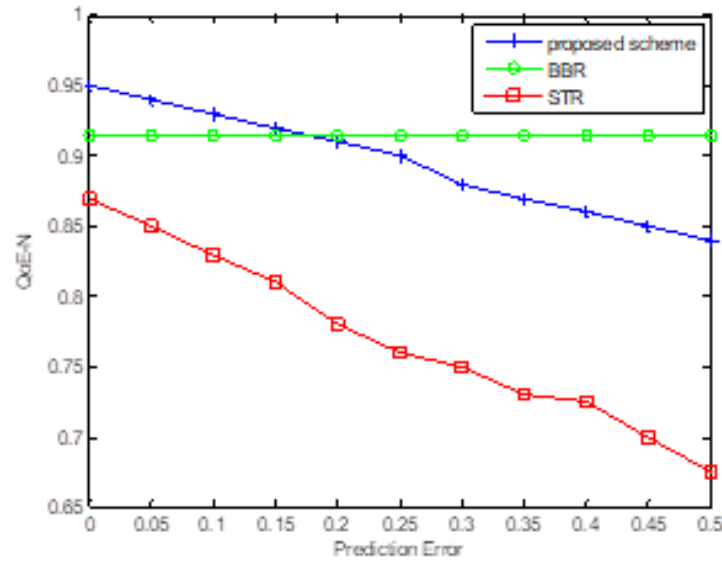


Figure 3. Prediction error comparison for proposed scheme with others.

In Fig. 4, as customers place more weights to change bitrate, the proposed scheme shows more advantage over STR and BBR. This is because the proposed scheme model is a video bit rate trade-off against the QoE function bit rate changes, while the STR and BBR decrease as the contrast increases. However, when outage time is a more important factor, the BBR scheme works similarly to this STR scheme because the BBR scheme maintains a minimal buffer level. The video player has a better chance of moving away from the lower bandwidth with less or no outage. Although our chart does a good job with a perfect prediction of bandwidth, it may suffer from a long outage due to the lack of a perfect average bandwidth predictor. Still, the performance of our scheme has been improved, such as keeping the level to a minimum and using an indicator for accuracy in bandwidth.

Fig. 5 explains the effect of video buffer size. At the beginning, the buffer size is small (playtime is less than 25 seconds); increasing the buffer size can improve all schemes' performance. The larger buffer size protects the player from interruption events and provides more freedom to optimize performance. When the buffer size reaches a certain level (playing time 25 seconds), even if the buffer size is further increased, the performance of all schemes will remain unchanged. Finally, STR is least affected by the buffer size because it does not consider the level of occupancy in the decision logic.

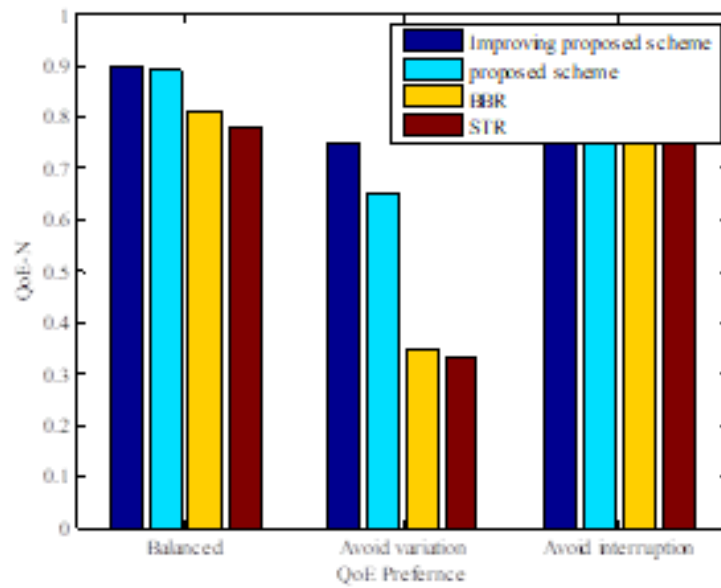


Figure 4. QoE Improvement Comparison of different schemes.

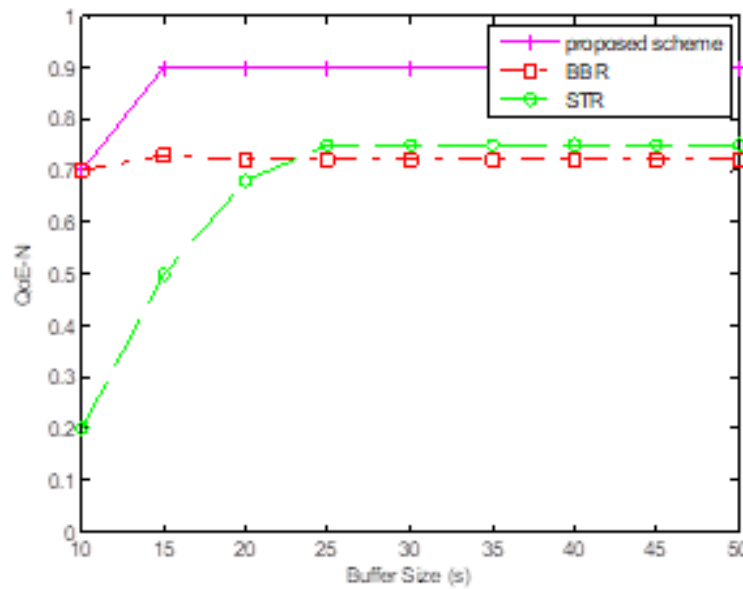


Figure 5. Buffer occupancy comparison of the proposed scheme with others.

From Fig. 6, the scheme automatically optimizes the startup delay; it analyzes how overall QoE would be affected if the startup delay were fixed. As shown in Fig.6, when the startup time increases, all schemes' performance enhances since the player of user collects more clips in the video buffer in the startup phase, which makes it easier to administrate interruption situations.

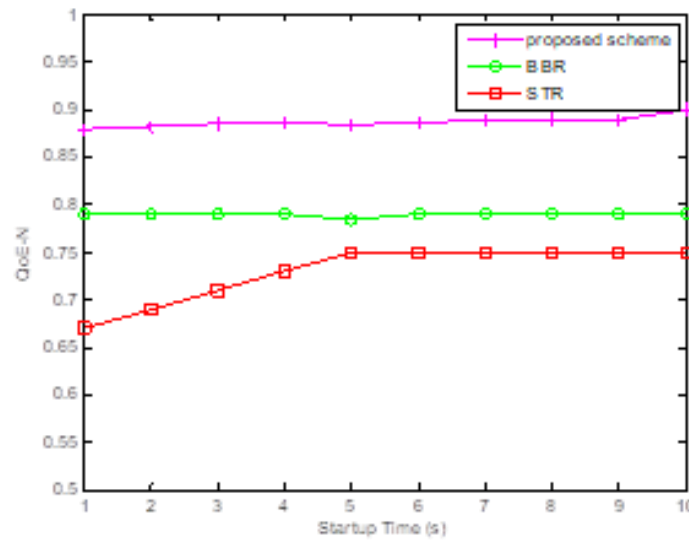


Figure 6. Startup time comparison of the proposed scheme with others.

6. CONCLUSIONS

In this paper, an adaptive bitrate scheme is suggested based on bandwidth prediction algorithms to ensure video playback continually. It was found that the conventional adaptive bitrate algorithms can provide continuous video playback but result in a serious rate fluctuation. At this point, to avoid buffer underflow/overflow and filter the impact of fluctuation in bandwidth, two thresholds are used as main points. When the timing space between these two thresholds becomes small, video bitrate fluctuates drastically. While it increases the timing space by about 60 sec, the video bitrate be stable. But when setting these thresholds dynamically, the performance becomes better for video buffer level. Then a bandwidth prediction algorithm is proposed to improve and stabilize the adaptation of video bitrate. The main elements that affect QoE Improvement are discussed. The work deduces that even the algorithm can provide a suitable video bitrate, but there is still the same fluctuation in video buffer occupancy, resulting in deteriorating the QoE. So the algorithm is improved using a small positive/negative bandwidth margin. The playback freezing happens twice, once in the case, there is no bandwidth margin used and second when the bandwidth margin is huge while smooth video bitrate is obtained with moderate value. The implementations of our scheme are compared with other schemes, the smooth throughput rate (STR) and adaptive schemes based on the buffering rate (BBR). The proposed scheme is outperformed these in achieving smooth rates and continuous playback.

**REFERENCES**

- Ni, P., Eg R, Eichhorn A, Griwodz C, and Halvorsen P., 2011. Flicker effects in adaptive video streaming to handheld devices. In Proceedings of the 19th ACM international conference on Multimedia Nov 28 (pp. 463-472). ACM.
- Stoekhammer, T., 2011. Dynamic adaptive streaming over HTTP-design principles and standards. In Proceedings of the Second Annual ACM Conference on Multimedia Systems (Vol. 2014, pp. 2-4). New York, USA: ACM.
- Ur Rahman W, and Chung K., 2015. Buffer-Based Adaptive Bitrate Algorithm for Streaming over HTTP. KSII Transactions on Internet and Information Systems (TIIS). Nov. 9 (11):4585-603.
- Liu C, Bouazizi I, and Gabbouj M., 2011 Rate adaptation for adaptive HTTP streaming. In Proceedings of the second annual ACM conference on Multimedia systems, Feb 23 (pp. 169-174). ACM.
- Kadhim, D.J., Yu, X., Jabbar, S.Q., Li, Y., and Luo, W., 2017. A New Scheme for QoE Management of Live Video Streaming in Cloud Environment. In Pacific-Rim Symposium on Image and Video Technology, pp. 150-161.
- Jabbar, S.Q., Kadhim, D.J., and Li, Y., 2017. December. A Proposed Adaptive Rate Algorithm to Administrate the Video Buffer Occupancy for Smooth Video Streaming. In International Conference on Internet and Distributed Computing Systems, pp. 37-49.
- Jabbar, S.Q., Kadhim, D.J., and Li, Y., 2018. Developing a video buffer framework for video streaming in cellular networks. Wireless Communications and Mobile Computing.
- Sun Y, Yin X, Jiang J, Sekar V, Lin F, Wang N, Liu T, and Sinopoli B., 2016 Cs2p: Improving video bitrate selection and adaptation with data-driven throughput prediction. In Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference, Aug 22 (pp. 272-285). ACM.
- Li Z, Zhu X, Gahm J, Pan R, Hu H, Begen AC, and Oran D., 2014 Probe and adapt: Rate adaptation for HTTP video streaming at scale. IEEE Journal on Selected Areas in Communications. Apr;32(4):719-33.
- Kadhim, D.J., and Hamad, O.A., 2014. Improving IoT Applications Using a Proposed Routing Protocol. Journal of Engineering, 20(11), pp.50-62.
- Miller K, Quacchio E, Gennari G, and Wolisz A., 2012. Adaptation algorithm for adaptive streaming over HTTP. In Packet Video Workshop (PV), 19th International 2012 May 10 (pp. 173-178). IEEE.
- Jabbar, S.Q., Kadhim, D.J., and Li, Y., 2018. February. An Adaptive Bitrate Algorithm Based on Estimation and Video Adaptation for Improving QoE in DASH. In 2018 International Conference on Computer Science, Electronics and Communication Engineering (CSECE 2018). Atlantis Press.
- Abdulredah, S.H. and Kadhim, D.J., 2020. New Approaches of Cloud Services Access using Tonido Cloud Server for Real-Time Applications. Journal of Engineering, 26(8), pp.83-99.



- Dobrian F, Sekar V, Awan A, Stoica I, Joseph D, Ganjam A, Zhan J, and Zhang H., 2011. Understanding the impact of video quality on user engagement. In ACM SIGCOMM Computer Communication Review, Aug. 15 (Vol. 41, No. 4, pp. 362-373). ACM.
- Liu Y, Dey S, Gillies D, Ulupinar F, and Luby M., 2013. User experience modeling for DASH video. Packet Video Workshop (PV), 2013 20th International, Dec. 12 (pp. 1-8). IEEE.
- Zhang Y, and Duffield N., 2011. On the constancy of Internet path properties. In Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement, Nov. 1 (pp. 197-211). ACM.
- Huang TY, Johari R, McKeown N, Trunnell M, and Watson M., 2015. A buffer-based approach to rate adaptation: Evidence from a large video streaming service. ACM SIGCOMM Computer Communication Review, Feb. 25; 44(4):187-98.
- ZTI Telecom, "IP Traffic-Test & Measure." Available: <http://www.zticommunications.com/>.