CrossMark

Design of new resource allocation scheme for symbiosis of DASH clients and non-DASH clients

Hyun Jun Kim[†] ⓘ, Ye Seul Son[†] and Joon Tae Kim[*]

Abstract

In this paper, we propose a new resource allocation scheme for symbiosis of DASH clients and non-DASH clients (RASS) to provide seamless video streaming service to DASH clients and to guarantee fairness among non-DASH clients. In RASS, all the LTE downlink resource blocks are divided into two chunks by a KKT-conditions-based Resource-Division Controller (KCRDC). With the two divided chunks, the proposed scheme leads PFRA and KKT-conditions-based resource allocation scheme (KCRA) to allocate each chunk to non-DASH clients and DASH clients, respectively. PFRA allocates resource blocks with maintaining good trade-off between fairness and spectrum efficiency. On the other hand, KCRA allocates more resource blocks to DASH clients in the risk of re-buffering events to preferentially reduce the re-buffering-time ratios of the DASH clients. Finally, simulations are completed using the NS-3 (network simulator). The obtained results show that the proposed scheme shows a better synthetic performance than the existing resource allocation schemes in consideration of the re-buffering-time ratios and QoE value of DASH clients and the fairness among non-DASH clients.

Keywords: Resource allocation scheme, Karush-Kuhn-Tucker (KKT)-conditions, Rate-adaptation scheme, Dynamic adaptive streaming over http, Long-Term Evolution (LTE)

1 Introduction

Recently, due to the development of the wireless communication network and mobile devices, mobile-data traffic has increased rapidly. According to the Cisco Global Mobile Data Traffic Forecast [1], mobile-data traffic grew 18-fold from 2011 to 2016, while mobile-video traffic accounted for 60% of the total mobile-data traffic in 2016. Moreover, it is expected that mobile-video traffic will increase 9-fold between 2016 and 2021 to account for up to 78% of the total mobile-data traffic. Accordingly, the Hypertext Transfer Protocol (HTTP)-based Dynamic Adaptive Streaming Over HTTP (DASH) is emerging as a new standard for video streaming for the efficient processing of the tremendous mobile-video traffic in mobile networks. DASH [2–7] is the adaptive-streaming standard that provides seamless video-streaming services to DASH clients in time-varying network conditions. By using the

HTTP/TCP protocol, DASH can overcome the firewall problem in contrast to the previously used RTP/UDP. In other words, firewall does not block HTTP packets due to the usage of well-known port number (80) while it might block RTP packets due to the usage of registered port number (5004). It is also cost-effective due to the employment of standard HTTP servers. In DASH, media content is encoded into various versions at different bit-rates and is divided into multiple segments, and these can be played for seconds or tens of seconds. After the division, the segments are stored in the HTTP servers. To obtain the utmost quality of experience (QoE), DASH clients act accordingly by requesting segments that are appropriate under the variable network conditions.

In the Long-Term Evolution (LTE) network, base stations allocate radio resource blocks, i.e., physical resource blocks (PRBs), to the LTE clients to communicate with them simultaneously. One of the well-known resource

*Correspondence: jtkim@konkuk.ac.kr
[†]Hyun Jun Kim and Ye Seul Son equally contributed to this work.
Department of Electronic Engineering, Konkuk University, Seoul, Republic of Korea

allocation scheme named as Proportional Fair Resource Allocation (PFRA) [8, 9] maintains good grade-off between the fairness and spectrum efficiency. Also, there are several other resource scheduling schemes such as Maximum Throughput, Blind Equal Throughput, Token Bank Fair Queue, and Priority Set schemes that serve mobile clients with their own mechanism effectively [10]. These schemes, however, are not appropriate to provide high QoE to DASH clients, since they do not consider whether the traffic is video traffic or not. In this context, with the fact that the average video quality, variance of video quality, and re-buffering-time ratio critically affect QoE [11, 12] (here, the re-buffering-time ratio affects the QoE most significantly), Base Station Optimization (BSOP) scheme is proposed in [13]. BSOP is based on the Lagrange dual approach, and it provides optimal resource allocation solutions by considering the DASH-clients' playback buffer levels to reduce the re-buffering-time ratios of video services. Also, several resource allocation schemes are proposed to provide utmost QoE to multimedia clients [13–18]. Like this, plenty of resource allocation schemes have been proposed, but none of these schemes consider DASH clients and non-DASHs simultaneously. In the resource allocation process, it is important to consider DASH and non-DASH clients simultaneously since LTE system eventually receives payment from all clients.

In this paper, we propose a new resource allocation scheme for symbiosis of DASH clients and non-DASH clients (RASS) to provide seamless video streaming service to DASH clients and to guarantee fairness among non-DASH clients. In RASS, all the LTE downlink resource blocks are divided into two chunks by Karush-Kuhn-Tucker (KKT)-conditions [19, 20]-based Resource-Division Controller (KCRDC). One is for PFRA scheme and the other is for KKT-conditions-based resource allocation scheme (KCRA). KCRDC provides optimal solutions by considering the number of DASH clients, the number of non-DASH clients, the sum of the DASH-client buffer levels, and the channel conditions of every client. With the divided chunks, the proposed scheme leads PFRA and the KCRA to allocate each chunk to the non-DASH clients and the DASH clients, respectively. PFRA considers the achievable rates and the past average throughputs of non-DASH clients to maintain a good trade-off between the spectrum efficiency and the fairness. Alternatively, KCRA considers other factors such as the maximum achievable rates, buffer levels, and past average throughputs of the DASH clients, and it solves the optimization problem by using the KKT-conditions to preferentially reduce the re-buffering-time ratios of the DASH clients. As re-buffering events affect the QoE most significantly, it is expected to increase QoE of DASH clients by reducing re-buffering events.

Finally, simulations are completed using the NS-3 network simulator [21]. It is assumed that all the clients are in the edge of the LTE network, i.e., only one base station can serve the clients. This is for the verification that in such harsh network environment, RASS can provide best QoE to DASH clients by reducing re-buffering events and guarantees fairness among non-DASH clients. The role of Client-Side Rate-Adaptation Scheme (CSRAS) is also important. CSRAS is an application level algorithm applied in DASH clients and helps DASH clients to request optimal quality of video segments. Since each CSRAS behaves differently and exhibits different performance, there is a possibility of gap between performances of resource allocation schemes depending on the type of CSRAS. So, in the simulations, various CSRAS are used to show that the proposed scheme operates effectively regardless of the type of CSRAS. The obtained results show that the proposed scheme shows a more effective performance than those of two selected reference schemes (one from the schemes that do not consider video streaming (PFRA) and one from the schemes that consider video streaming (BSOP)) through an overall consideration of the QoE values of the DASH clients and the fairness among the non-DASH clients.

To sum up, the main contributions of this paper are summarized as follows:

- The proposal of a new resource allocation scheme for symbiosis of DASH clients and non-DASH clients (RASS) that allocates radio resource blocks to DASH clients and non-DASH clients for its purpose. The proposed scheme implements a control process to divide all the available PRBs into two chunks, which are used by PFRA and KCRA, and let PFRA and KCRA allocate PRBs to non-DASH clients and DASH clients, respectively.
- Various simulations with multiple client side rate adaptation schemes, reference resource allocation schemes, and network environments.
- The demonstration that the proposed scheme can increase the QoE value of DASH clients by reducing re-buffering events, especially in harsh network conditions.
- The demonstration that the proposed scheme can guarantee the fairness of using radio resource blocks among non-DASH clients.

This paper is structured as follows. In Section 2, a review of the recent studies that are related to our work is presented. Section 3 outlines design principle and evaluation items, while Section 4 presents the optimization process of the KCRDC which is one of the components of RASS. Section 5 then describes the KCRA, and Section 6 presents the simulation results. Finally, Section 7 concludes this paper.

2 Related works

Several rate-adaptation and resource allocation schemes have been proposed to handle tremendous video traffic and provide a higher QoE to DASH clients. In Section 2.1, several resource allocation schemes are described. In Section 2.2, several CSRASs are outlined.

2.1 Resource allocation schemes

In [22], the authors proposed a scheme that maximizes the throughput and minimizes the delay of real-time traffic to fulfill the QoS criteria. This scheme consists of the following two resource allocation schemes: (1) PFRA scheme and (2) Maximum-Throughput Resource Allocation (MTRA) scheme. Real-time traffic is scheduled by the proportional fair scheduler, whereas non-real-time traffic is scheduled by the maximum throughput scheduler.

The authors of [13] developed a formulation to maximize the QoE with a combined CSRAS–BSOP scheme. The key factors that are considered for client side rate-adaptation scheme are as follows: (1) average video quality, (2) variance of video quality, and (3) start-up delay. Meanwhile, the key factors that are considered in BSOP are the re-buffering-time ratios of DASH clients and the achievable rate of PRBs.

In [16], QoE-based resource allocation for adaptive HTTP video delivery was proposed to enhance the user QoE in multi-user over-the-top (OTT) DASH. This scheme uses information such as buffer levels of streaming users and channel quality of mobile users. With the information, a proxy-based method is used to optimize the video rate that is requested by streaming users.

A bandwidth-efficient multi-path transport layer protocol for high-quality real-time video was proposed in [17]. This transport layer protocol uses priority-aware data scheduling and adaptive forward error correction (FEC) coding algorithm to mitigate the packet loss rate.

In [18], the authors proposed a QoE-driven cross-layer optimization method to provide improve QoE of DASH clients. The proposed method is a proxy-based scheme that eNodeB changes the HTTP Get Request according to the channel condition and other DASH-related information of DASH clients. By using this scheme, DASH clients with good channel conditions and buffer capacity can request enhancement layers of downloaded video segments so that DASH clients can be provided higher QoE.

2.2 Client-side rate-adaptation schemes

The agile smooth-video adaptation algorithm (SVAA) [23] uses the buffer occupancy and the buffer trend to select the segment version that is appropriate for the current network condition and the buffer state of the DASH client.

Also, SVAA applies a buffer cap to prevent the buffer overflow that occurs when the buffer is already full and receives another data.

In [24], the authors proposed the segment-fetching time method (SFTM) rate-adaptation scheme for the serial and parallel segment-fetching methods in the content distribution network (CDN). The proposed scheme uses the ratio of the expected segment-fetch time and the measured segment-fetch time to detect the network capacity. Also, the scheme uses a stepwise switch-up method and a multistep switch-down method that are based upon the proposed rate-adaptation metric. In addition, an idling method is used to prevent buffer overflows, and the prioritized optimum segment-fetch time is suggested for the newly joined DASH clients to solve the fairness problem among the DASH clients.

Also, the fuzzy-logic-based rate-adaptation scheme (FDASH) presented in [25] adapts the segment bit-rate by using the fuzzy-logic controller (FLC). The FLC uses the following inputs: (1) buffer level and (2) differential of the buffer level. Also, the scheme controls the bit-rate of the next segment by using the throughput-estimation value.

## 3 Methods and evaluation items

RASS is designed to reduce probabilities of re-buffering events of multi-media services and to guarantee fair services among non-DASH clients. The design of RASS is shown in Fig. 1. The overall system consists of the client-side architecture and the base-station-side architecture, and the details of the client-side architecture and the base-station-side architecture are represented in Sections 3.1 and 3.2, respectively.

3.1 Client-side architecture

In the client-side architecture, DASH clients receive the media-content segments that are then stored in their Playout Buffers in the received order. The DASH clients then play the accumulated video segments one by one. During the video playback, the information of the buffer levels of the DASH clients is extracted from the Playout Buffers and is included in the *HTTP Get Request* [26]. The format of the *HTTP Get Request* used in RASS is described in Fig. 2. The left side of the format is elliptical, which is an original format of the *HTTP Get Request* and includes the necessary DASH information such as resolution id and segment id. The buffer states of DASH clients are added to the original format with 32-bit size. Then, the *HTTP Get Requests* are transferred to the base station via the uplink data channel (PUSCH) [27] of the LTE system.

3.2 Base-station-side architecture

The LTE system that is used in the proposed network model uses the Frequency Division Duplexing (FDD)

Fig. 1 System architecture
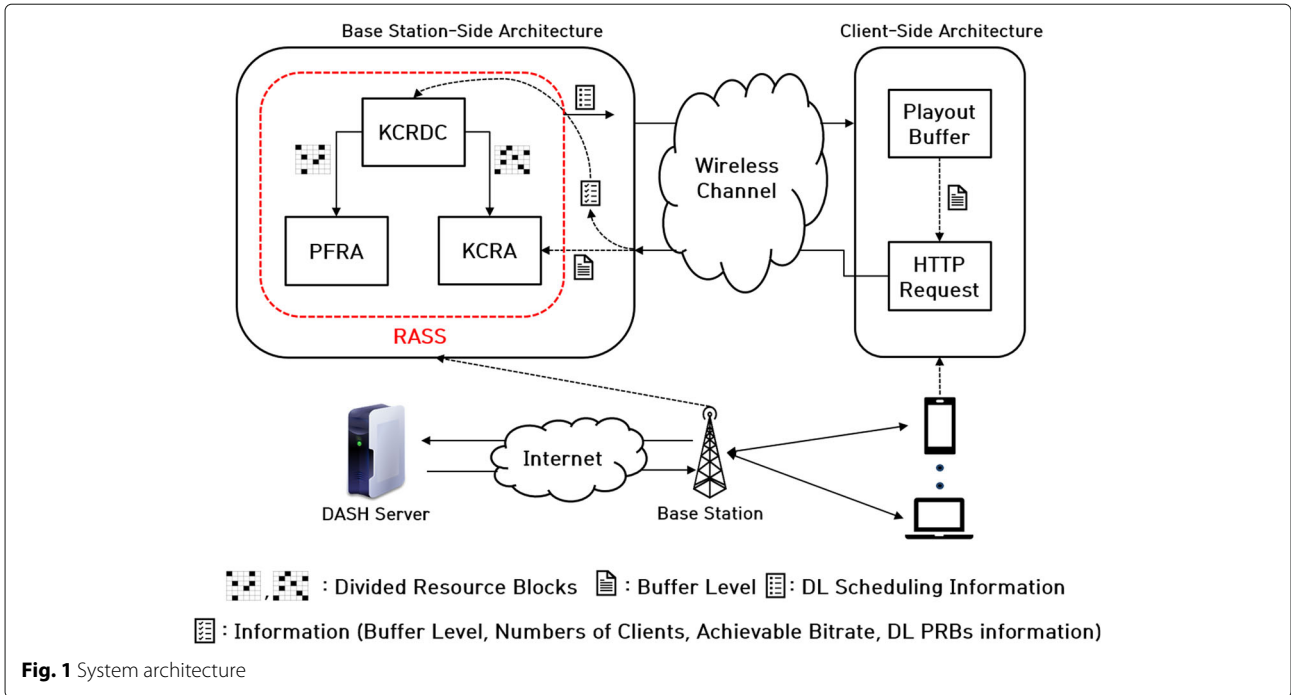
scheme [28]. In the FDD scheme, since uplink and downlink transmission use different frequency bands, clients send and receive packets simultaneously. Also, multiple clients communicate with a base station simultaneously due to resource (sub-channels, power) allocation in every transmission time interval (TTI). At every TTI (i.e., every 1 ms), the clients send Channel Quality Information (CQI) [29] to their base station. By using the CQI, a base station can calculate the achievable rate that a user can achieve with the allocated PRBs, as follows:

$$R_{\text{achieve}} = \frac{TB \, (bytes)}{1 \, (ms)} \qquad (1)$$

where *TB* is the size of a transport block that is determined by a Modulation and Coding Scheme (MCS) and the number of allocated PRBs. Additionally, the MCS is determined by the predefined table that informs the selectable MCS based on the given CQI [30]. At the base-station side, the link-layer model that is proposed in [31] is used. In this model, the data rate of the user *i* is denoted as R_i and the maximum achievable rate that the user *i*

can achieve if all the available PRBs are allocated to the user *i* is denoted as $R_{max,i}$. The R_i for all of the *N* users is denoted as:

$$R_i = \alpha_i R_{max,i}, \quad 0 \leq \alpha_i \leq 1, \, \forall i \qquad (2)$$

where α_i denotes the proportion of the total PRBs that the user *i* can use and this value will be calculated in our proposed scheme. An $R_{max,i}$ for each user is updated in each optimization round, i.e., in each TTI, based on its channel condition that is informed by the user. As an expansion of the client-side architecture, the base station receives the *HTTP Get Request*, and this is followed by the extraction of the information about the buffer level from the *HTTP Get Request* that is sent to the RASS. The other two control factors, i.e., the achievable bit-rate and the numbers of DASH and non-DASH clients, as well as the buffer levels of the DASH clients, are included in the RASS. The numbers of DASH and non-DASH clients can be achieved by the control message from DASH clients. If a DASH client starts a streaming service, then it sends a control message to its base station through control channel so that the base station can identify this client as a DASH client and increases the number of DASH clients. On the other hand, if a DASH client stops playing streaming service, it sends a control message again so that the base station can erase this client from the DASH client list and decreases the number of DASH clients. After receiving all the information, KCRDC starts a control process using the KKT-Conditions. The details of these factors are described in Section 4. The KCRDC optimally divides the

Fig. 2 *HTTP Get Request* Format
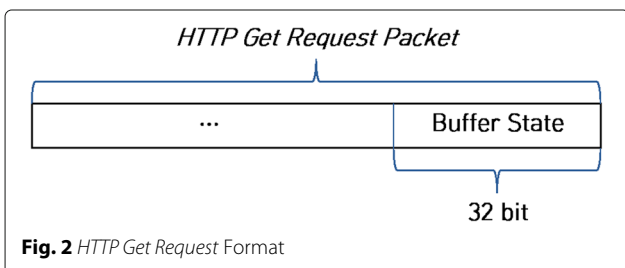
total PRBs into two chunks by using the KKT-conditions. One chunk is for PFRA scheme and the other is for the KCRA scheme. Then, PFRA scheme allocates its resource chunk to the non-DASH clients according to the following equation:

$$i_k(t) = arg \max_{j=1,...,N} \frac{R_j(k,t)}{T_j(t)}, \quad \forall j, t, k, \quad (3)$$

where $i_k(t)$ denotes the chosen user for the transmission of the kth PRB at the time t, $R_j(k,t)$ is the achievable rate for the kth PRB at t, and $T_j(t)$ is the past average throughput of the user j. On the other hand, the other PRB chunk is allocated to the DASH clients by the KCRA scheme. The details of the KCRA scheme are described in Section 5.

To sum up, RASS receives (1) numbers of DASH and non-DASH clients, (2) achievable bit-rate of all the clients, (3) playback buffer levels of DASH clients, and (4) downlink PRBs information as input items. Then, KCRDC, KCRA, and PFRA make downlink scheduling information as an output item of RASS.

3.3 Evaluation items

The evaluation items that are used to validate the proposed scheme are as follows: (1) average re-buffering-time ratios of the DASH clients; (2) *average received data* and *standard deviation of the received data* of the non-DASH clients; and (3) average QoE value of the DASH clients. The following equations are used to calculate (1) and (2):

$$Reb(i) = \frac{I_{total,i}}{\ell_i}, \quad \forall i, \quad (4)$$

$$m = \frac{1}{N_{non-DASH}} \sum_{i=1}^{N_{non-DASH}} R(i), \quad (5)$$

$$Stad = \sqrt{\frac{1}{N_{non-DASH}} \sum_{i=1}^{N_{non-DASH}} (R(i) - m)^2}, \quad (6)$$

where $Reb(i)$, $I_{total,i}$, and ℓ_i denote the re-buffering-time ratio of the ith DASH client, the total playback-interruption time of the ith DASH client, and the total video-playback duration, respectively, and m denotes the *average received data* of the non-DASH clients that are the results of the division operation of the sum of all of the non-DASH clients' received data [$R(i)$] and the number of DASH clients ($N_{non-DASH}$). Additionally, *Stad* denotes the *standard deviation of the received data* of the non-DASH clients that represents the degree of the fairness among the non-DASH clients.

In addition, a QoE model [11, 13, 32] is needed to prove that the QoE of the DASH clients is actually improved by the application of the proposed scheme; the employed QoE model was proposed in [13] for this reason. The following three factors are considered for the QoE model

that is used in the present paper: (1) average video quality; (2) variance of video quality; and (3) re-buffering-time ratio. The average video quality and the variance of video quality are represented as:

$$m_{avq}(i) = \frac{1}{N_{seg,i}} \sum_{k=1}^{N_{seg,i}} q(\Re_i[k]), \quad \forall i, \quad (7)$$

$$Var(i) = \frac{1}{N_{seg,i}} \sum_{k=1}^{N_{seg,i}} (q(\Re_i[k]) - m_{avq})^2, \quad \forall i, \quad (8)$$

$$q(\Re_i[k]) = \vartheta \log \Re_i[k]) + \iota_i, \quad (9)$$

where $m_{avq}(i)$ is the average video quality of the ith DASH client; $N_{seg,i}$ is the number of the segments that are received by the ith DASH client; $q(R_i[k])$ is the form of video quality that is proposed in [33], which can be controlled by the parameters ϑ_i and ι_i for different DASH clients; and $Var(i)$ is the variance of the video quality of the ith DASH client. With (4), (7), and (8), the QoE can be defined as

$$QoE(i) = m_{avq}(i) - \theta Var(i) - \xi Reb(i), \quad \forall i, \quad (10)$$

In the simulations of this study, ϑ_i and ι_i are set as 10 and 0 for all of the users, respectively. Also, θ and ξ are set as 0.2 and 300, respectively, since the re-buffering-time ratio is more critical to the QoE than the variance of video quality [34].

4 KKT-conditions-Based Resource-Division Controller

In this section, the description of the KCRDC is given. The control factors for the controlling of the size of two PRB chunks are presented in Section 4.1, and the KCRDC control process is described in Section 4.2.

4.1 Control factors of the KCRDC

In the proposed KCRDC, the following three factors for the effective division of the PRBs are considered:

1. The number of DASH clients and the number of non-DASH clients to allocate more PRBs to the group with the higher number of clients.
2. The sum of the maximum achievable bit-rates of the DASH clients and the sum of the maximum achievable bit-rates of the non-DASH clients to consider the effectiveness of the PRB usage.
3. Buffer levels of the DASH clients to allocate a larger number of PRB chunks to the KCRA for any of the DASH clients that are at risk of the re-buffering events.

The objective of this controller is the maximization of the effectiveness function $R(\alpha)$ that is given by

$$\mathcal{R}(\alpha) = \sum_{i=1}^{2} \hbar_i(t) C(\alpha_i), \tag{11}$$

$$\hbar_1(t) = \frac{N_{DASH} \sum_{i=1}^{N_{DASH}} \log(R_{max,i} + a)}{\sum_{i=1}^{N_{DASH}} e^{\frac{B_i[t] + \varepsilon}{B_{max,i}}}}, \quad \forall i, t, \tag{12}$$

$$\hbar_2(t) = N_{non-DASH} \sum_{i=1}^{N_{non-DASH}} \log(R_{max,i} + a), \quad \forall i, t, \tag{13}$$

where $\alpha = [\alpha_1, \alpha_2]$, $C(\alpha_i) = -(\alpha_i - 1)^2$ is a convex function for the optimization, N_{DASH} is the number of DASH clients, $N_{non-DASH}$ is the number of non-DASH clients, $B_i[t]$ is the buffer level of the user i at the time t, $B_{max,i}$ is the maximum buffer level that the user i can use, ε is a small constant value, and a is a small positive constant value with an $a > 1$ to ensure that the $\log(R_{max,i} + a) > 0$. Additionally, $\hbar_1(t)$ is the control function for the DASH clients that represents how much of downlink resource blocks will be taken from DASH clients. If the $B_i[t]$ is decreased, the number of PRBs that are allocated to the KCRA becomes larger to prevent the re-buffering events of the DASH clients. In the same manner, if the $\sum_{i=1}^{N_{DASH}} \log(R_{max,i} + a)$ is increased, the number of PRB chunks that are allocated to the KCRA will become larger, since it is important to consider not only the re-buffering-time ratios of the DASH clients but also the effectiveness of the PRB usage. Also, it is reasonable to allocate a greater number of PRBs if the N_{DASH} is increased, since a higher N_{DASH} means that the necessity of the PRBs has increased. Also, we have used log function for achievable rate ($R_{max,i}$) and exponential function for buffer levels ($B_i[t]$) of DASH clients. It is because we want buffer levels to be more influential than achievable rate. In other words, by using the characteristic that exponential values changes rapidly than log values with same amount of input changes, KCRA can get PRBs sufficiently and allocates them to DASH clients who are at risk of buffer underflow.[1]

Likewise, $\hbar_2(t)$ is the control function of the non-DASH clients that stands for the amount of PRBs that is needed for the non-DASH clients. Also, like $\hbar_1(t)$, the amount of the PRB chunk is controlled by the $\sum_{i=1}^{N_{non-DASH}} \log(R_{max,i} + a)$ and the $N_{non-DASH}$.[2]

4.2 Control process

In the control process of the KCRDC, the size of chunks that are allocated to PFRA and the KCRA are controlled with the use of the KKT-conditions. With (11), (12), and (13), it is possible to formulate the optimization problem for the KCRDC as (14), (15), and (16) and the Lagrangian of this problem is represented by (17), as follows:

$$\alpha = arg \max_{\alpha_1, \alpha_2} \mathcal{R}(\alpha), \tag{14}$$

$$s.t. \quad g(\alpha) = \sum_{i=1}^{2} \alpha_i - 1 \leq 0, \tag{15}$$

$$f_i(\alpha_i) = \alpha_i \geq 0, \quad \forall i, \tag{16}$$

$$L(\alpha, \lambda) = \mathcal{R}(\alpha) - \lambda_1 g(\alpha) + \sum_{i=1}^{2} \lambda_{i+1} f_i(\alpha_i), \tag{17}$$

As described earlier in Section 4.1, the objective of this optimization problem is the finding of the proportion set α that maximizes the effectiveness function $R(\alpha)$. Obviously, the optimization problem is constrained by (15) and (16), since the sum of proportion values ($alpha_i$) must be less than or equal to 1, and each proportion value must be greater than or equal to zero. The KKT-conditions for (14), (15), and (16) are given by:

$$L_{\alpha_j} = \sum_{i=1}^{2} \hbar_i(t) \nabla_{\alpha_j} C(\alpha_i) - \lambda_1 \nabla_{\alpha_j} g(\alpha) \\ + \sum_{i=1}^{2} \lambda_{i+1} \nabla_{\alpha_j} f_i(\alpha_i) = 0, \quad \forall j, \tag{18}$$

$$L_{\lambda_j} = \sum_{i=1}^{2} \hbar_i(t) \nabla_{\lambda_j} C(\alpha_i) - \nabla_{\lambda_j} \lambda_1 g(\alpha) \\ + \sum_{i=1}^{2} \nabla_{\lambda_j} \lambda_{i+1} f_i(\alpha_i) = 0, \quad \forall j, \tag{19}$$

$$\lambda_1 g(\alpha) = 0, \quad \lambda_{i+1} f_i(\alpha_i) = 0, \quad \forall i, \tag{20}$$

$$\lambda_k \geq 0, \quad \forall i, \tag{21}$$

By using the conditions of (18), (19), (20), and (21), this problem can be formulated into a matrix form, as follows:

$$A \times x = B, \tag{22}$$

$$A = [a_1, \ldots, a_5]^T, \tag{23}$$

$$B = [b_1, \ldots, b_5]^T, \tag{24}$$

$$x = [\alpha_1, \alpha_2, \lambda_1, \lambda_2, \lambda_3]^T \tag{25}$$

where a_i is the vector that contains coefficient of α_i and λ_i in equation L_{α_i} and L_{λ_i}. Also, b_i is the constant value of

equations L_{α_i} and L_{λ_i}. For instance, if $L_{\alpha_1} = 10\alpha_1 + 5\alpha_2 + 0\lambda_1 - \lambda_2 + 10\lambda_3 = 7$, then $a_1 = [10 \quad 5 \quad 0 \quad -1 \quad 10]$ and $b_1 = 7$

$$x = A^{-1} \times B, \qquad (26)$$

If the KKT conditions are not satisfied with the obtained results from (26), some of the conditions are deactivated and the vector x is recalculated. This process is repeated until all the KKT conditions are satisfied. With the values α_1 and α_2 that are finally obtained from Eq. (26), two PRB chunks that are represented as follows can be obtained:

$$N_{\text{PRBs}-\text{DASH}} = Tr(\alpha_1 N_{\text{PRBs}}), \qquad (27)$$

$$N_{\text{PRBs}-\text{non}-\text{DASH}} = Tr(\alpha_2 N_{\text{PRBs}}), \qquad (28)$$

where N_{PRBs} is the total number of PRBs and $Tr()$ is a truncation function. The sum of $N_{\text{PRBs}-\text{DASH}}$ and $N_{\text{PRBs}-\text{non}-\text{DASH}}$, however, will not be N_{PRBs} since $\alpha_1 N_{\text{PRBs}}$ and $\alpha_2 N_{\text{PRBs}}$ are truncated into the integer. In our work, the remaining PRBs are allocated to one of the two schemes which has bigger decimal value.

5 KKT-conditions-based resource allocation for DASH clients

The KCRA scheme for DASH clients is described in this section. With the resource chunk that is allocated by the KCRDC, the KCRA scheme only needs to allocate resource chunk to the DASH clients to prevent the re-buffering events. Although the KCRA is based on the KKT-conditions same as the KCRDC, its weight factors are different from the KCRDC control factors, since the KCRA objective is different from that of the KCRDC. In Section 5.1, the KCRA weight factors are presented, and the optimization process is described in Section 5.2.

5.1 Weight factors of the KCRA
To effectively allocate the PRBs, it is important for the KCRA scheme to consider the following factors:

1. The maximum achievable bit-rates of the DASH clients to maximize the effectiveness of the PRB usage.
2. The buffer levels of the DASH clients to prevent the corresponding re-buffering events.
3. The past average throughputs of the DASH clients to achieve a certain level of fairness among the DASH clients.

The objective of the KCRA is the maximization of the effectiveness function that is given by (29), as follows:

$$\mathcal{F}(\beta) = \sum_{i=1}^{N_{DASH}} \phi_i(t) C(\beta_i), \qquad (29)$$

$$\phi_i(t) = \frac{R_{max,i} w_i(t)}{T_i(t)}, \quad \forall i, t, \qquad (30)$$

$$w_i(t) = \frac{\gamma_i(t)}{\sum_{i=1}^{N} \gamma_i(t)}, \quad \forall i, t, \qquad (31)$$

where $\gamma_i(t) = \log\left(\frac{B_{i,max}}{B_i[t] + \eta}\right)$ for all of the t values [13], η is a constant, $C(\beta_i) = -(\beta_i - 1)^2$ is a convex function for the optimization, and $\phi_i(t)$ is the weight function that consists of the previously described three factors. Also, $\beta = [\beta_1, ..., \beta_{N_{DASH}}]$ and β_i represents the percentage of total downlink resources that ith DASH client will use. The DASH client with the larger $\phi_i(t)$ acquires a larger portion of the PRBs, i.e., the DASH client whose buffer level is lower than those of the others (i.e., a large $w_i(t)$), the DASH client whose $R_{max,i}$ is larger than those of the others, and the DASH client who previously received less data, i.e., a low $T_i(t)$, can use more PRBs than the other DASH clients. [3]

5.2 KCRA optimization process
With (29), (30), and (31), the performance of an optimization process that is similar to the control process of the KCRDC is possible. First, a formulation of the optimization problem as (32), (33), and (34) are necessary, followed by a representation of the Lagrangian of this problem as (35), as follows:

$$\beta = arg \max_{\beta_1, ..., \beta_{N_{DASH}}} \mathcal{F}(\beta) \qquad (32)$$

$$s.t. \quad \varphi(\beta) = \sum_{i=1}^{N_{DASH}} \beta_i - 1 \leq 0, \qquad (33)$$

$$\mu_i(\beta_i) = \beta_i \geq 0, \quad \forall i, \qquad (34)$$

$$L(\beta, \nu) = \mathcal{F}(\beta) - \nu_1 \varphi(\beta) + \sum_{i=1}^{N_{DASH}} \nu_{i+1} \mu_i(\beta_i), \quad (35)$$

In common with Section 4.2, the KCRA optimization problem is constrained by (33) and (34), and with these constraints, the KKT-conditions for (32), (33), and (34) are given by:

$$
\begin{aligned}
L_{\beta_j} = &\sum_{i=1}^{N_{DASH}} \phi_i(t) \nabla_{\beta_j} C_i(\beta_i) - \nu_1 \nabla_{\beta_j} \varphi(\beta) \\
&+ \sum_{i=1}^{N_{DASH}} \nu_{i+1} \nabla_{\beta_j} \mu_i(\beta_i) = 0, \quad \forall j,
\end{aligned}
\qquad (36)
$$

$$
\begin{aligned}
L_{\nu_j} = &\sum_{i=1}^{N_{DASH}} \phi_i(t) \nabla_{\nu_j} C_i(\beta_i) - \nabla_{\nu_j} \nu_1 \varphi(\beta) + \sum_{i=1}^{N_{DASH}} \nabla_{\nu_j} \nu_{i+1} \mu_i(\beta_i) \\
&= 0, \quad \forall j,
\end{aligned}
$$
$$(37)$$

Table 1 Simulation parameters

DASH parameters	
Client-side rate-adaptation scheme	SVAA, SFTM, FDASH
Base-station-side resource allocation scheme	PFRA, BSOP, RASS
Number of representations	13
Segment size	2 s
Frame rate	40 fps, 50 fps
LTE parameters	
Downlink carrier frequency	2.12 GHz
Number of PRBs	25
Bandwidth per PRB	180 KHz
Fading model	EVA-60 kmph, ETU-3 kmph
Pathloss model	Log-Distance Propagation Loss model
Link capacity between PGW and DASH Server	1 Gbps
Simulation parameters	
Number of DASH clients	3,6,9
Number of non-DASH client	10,20,30
$B_{i,max}$	35 s
η	5
T	20 s
Simulation time	500 s

$$v_1\varphi(\beta) = 0, \quad v_{i+1}\mu_i(\beta_i) = 0, \quad \forall i, \tag{38}$$

$$v_k \geq 0, \quad for\, k = 1,\ldots,N_{\text{DASH}} + 1, \tag{39}$$

By using the conditions of (36), (37), (38), and (39), just like the KCRDC process, this problem can be formulated into a matrix form, as follows:

Table 2 Average re-buffering ratios of 3 DASH clients with 50 fps video contents

Resource Allocation Algorithm	Number of non-DASH client	Re-buffering ratio (%)					
		FDASH		SVAA		SFTM	
		EVA	ETU	EVA	ETU	EVA	ETU
BSOP	10	6.368	9.225	6.755	7.414	2.668	8.182
	20	11.219	7.01	5.117	26.362	2.959	7.712
	30	4.51	12.046	15.229	31.836	9.146	3.385
PFRA	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	1.431	0	1.432	0	1.431	0
RASS	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	0	0	0	0	0	0

Table 3 Average re-buffering ratios of 6 DASH clients with 50 fps video contents

Resource Allocation Algorithm	Number of non-DASH client	Re-buffering ratio (%)					
		FDASH		SVAA		SFTM	
		EVA	ETU	EVA	ETU	EVA	ETU
BSOP	10	0.7	0.188	2.156	2.598	0.896	5.030
	20	4.243	2.315	2.234	3.229	0.828	5.054
	30	2.925	3.017	3.459	3.194	2.237	5.248
PFRA	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	5.389	0.134	5.375	0.146	5.368	0.193
RASS	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	0	0	0	0	0	0

$$D \times y = E, \tag{40}$$

$$D = [d_1,\ldots,d_{2N_{\text{DASH}}+1}]^T, \tag{41}$$

$$E = [e_1,\ldots,e_{2N_{\text{DASH}}+1}]^T, \tag{42}$$

$$y = [\beta_1,\ldots,\beta_{N_{\text{DASH}}}, v_1,\ldots,v_{N_{\text{DASH}}+1}]^T, \tag{43}$$

where d_i is the vector that contains coefficient of β_i and v_i in equation L_{β_i} and L_{v_i} same as the a_i in Eq. (23). Also, e_i is the constant value of equations L_{β_i} and L_{v_i} same as the b_i in Eq. (24).

$$y = D^{-1} \times E, \tag{44}$$

The iterative process for obtaining optimal results described in Section 4.2 is also done in this optimization process. After the optimization process, the proportion set β is obtained using Eq. (44). By using the set β, every

Table 4 Average re-buffering ratios of 9 DASH clients with 50 fps video contents

Resource Allocation Algorithm	Number of non-DASH client	Re-buffering ratio (%)					
		FDASH		SVAA		SFTM	
		EVA	ETU	EVA	ETU	EVA	ETU
BSOP	10	1.836	0.765	2.277	1.15	1.650	3.599
	20	1.628	1.653	1.658	1.308	1.132	4.52
	30	2.832	1.642	0.701	1.952	1.071	6.028
PFRA	10	0	0	0	0	0	0
	20	0.070	0	0.068	0	0.070	0
	30	13.787	0.725	14.172	0.507	16.367	4.975
RASS	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	2.117	0	1.715	0	0.412	0

user i receives the amount of resource blocks that is determined by

$$iNumPRBs(i) = Tr(\beta_i N_{\text{PRBs-DASH}}), \quad (45)$$

The sum of $iNumPRBs(i)$, i.e., $\sum_{i=1}^{N} iNumPRBs(i)$, however, will not be $N_{\text{PRBs-DASH}}$, since $\beta_i N_{\text{PRBs-DASH}}$ is truncated into the integer. Therefore, to allocate the remaining PRBs to the clients, the Allocating Remaining PRBs (ARP) algorithm is applied.

Algorithm 1 describes the ARP algorithm. At first, the decimal of $\alpha_i N_{\text{PRBs}}$ is represented as $fNumPRBs(i)$. Then, the array $fNumPRBs$ is sorted into a descending order. Subsequently, with the confirmation that the PRB is available, the DASH clients obtain the PRBs one by one in the order of the $fNumPRBs$. If the $index$ is N_{DASH}, then the $index$ is set to 1 again to allocate more resources to the clients with a larger decimal.

Algorithm 1. ARP Algorithm
1:    for($int\, i = 0\, li < N_{DAHS}; i++)$
2:       $fNumPRBs(i) = \alpha_i N_{PRBs} - Tr(\alpha_i N_{PRBs})$
3:    end for
4:    $sort(fNumPRBs)$
5:    $int\, index = 1;$
6:    for($int\, k = 0; k < N_{PRBs}; k++)$
7:      if($PRBsMap(k) == false$)
8:       $PRBsMap(k) = true;$
9:       $allocate(index, k);$
10:      $index++;$
11:      if($index == N_{DASH}$)
12:       $index = 1;$
13:      end if
14:     end if
15:   end for
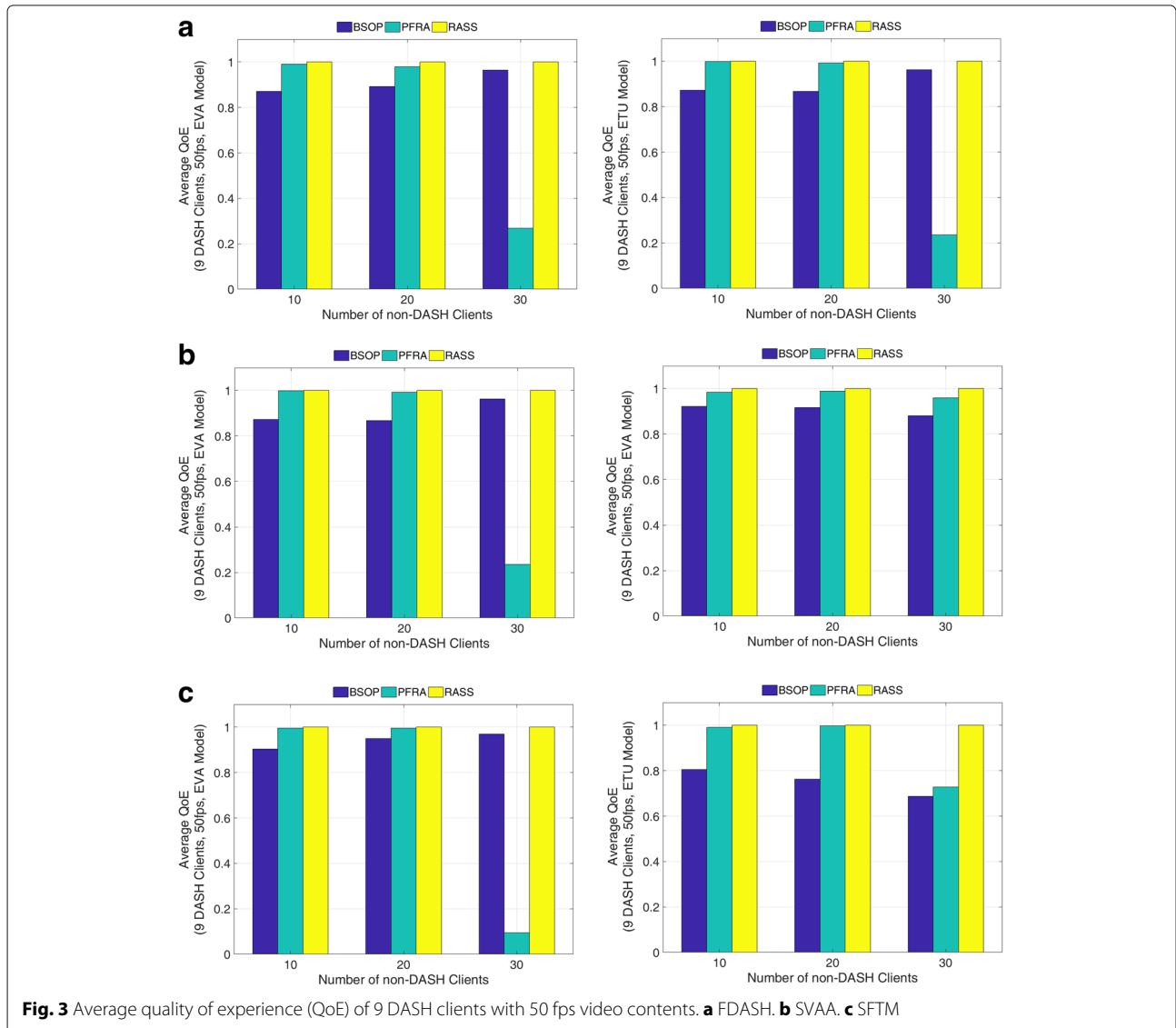
Fig. 3 Average quality of experience (QoE) of 9 DASH clients with 50 fps video contents. **a** FDASH. **b** SVAA. **c** SFTM

6 Results and discussion

To validate the proposed scheme, simulations are implemented in various LTE networks using the NS-3. In Section 6.1, an outline of the simulation parameters and evaluation items is given. In Section 6.2, the simulation results that are obtained in the LTE networks with the application of RASS, BSOP, and PFRA in the base station are introduced. In addition to the base-station application, one of three available CSRASs, SVAA, SFTM, and FDASH, is applied to all of the DASH clients in every simulation. With the obtained results, the following two facts are verified: (1) RASS shows more favorable performances than other reference resource allocation schemes upon the overall consideration of QoE of DASH clients and the fairness among non-DASH clients. (2) RASS operates effectively irrespective of the CSRAS type that is applied.

6.1 Simulation setup

To show that the proposed scheme works effectively in various fading environments, the simulation results in Section 6.2 are obtained in the LTE networks using the following two fading models: (1) Extended Vehicular A model—60 km/h, and (2) Extended Typical Urban model—3 km/h. The path loss model of the LTE network in the simulation is the log-distance path loss model. This path loss model can be expressed as

$$PL = PL_0 + 10n \log \left(\frac{d}{d_0} \right), \quad (46)$$

where PL is the path loss, PL_0 is the path loss at the reference distance, n is the path-loss exponent, and d_0 is the reference distance (set as 1 m in this simulation). In the simulation, the n is set as 3. Above this, each video
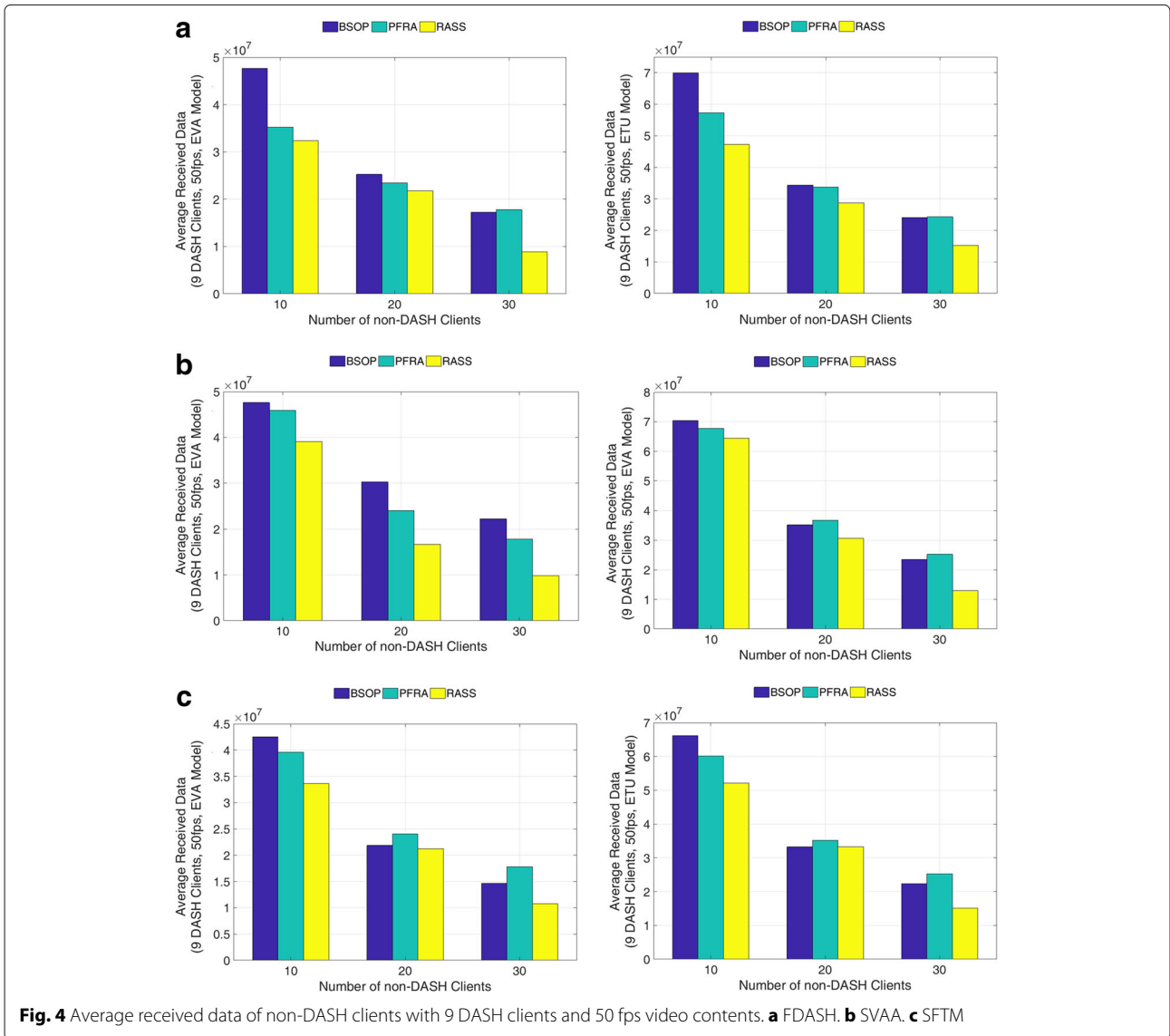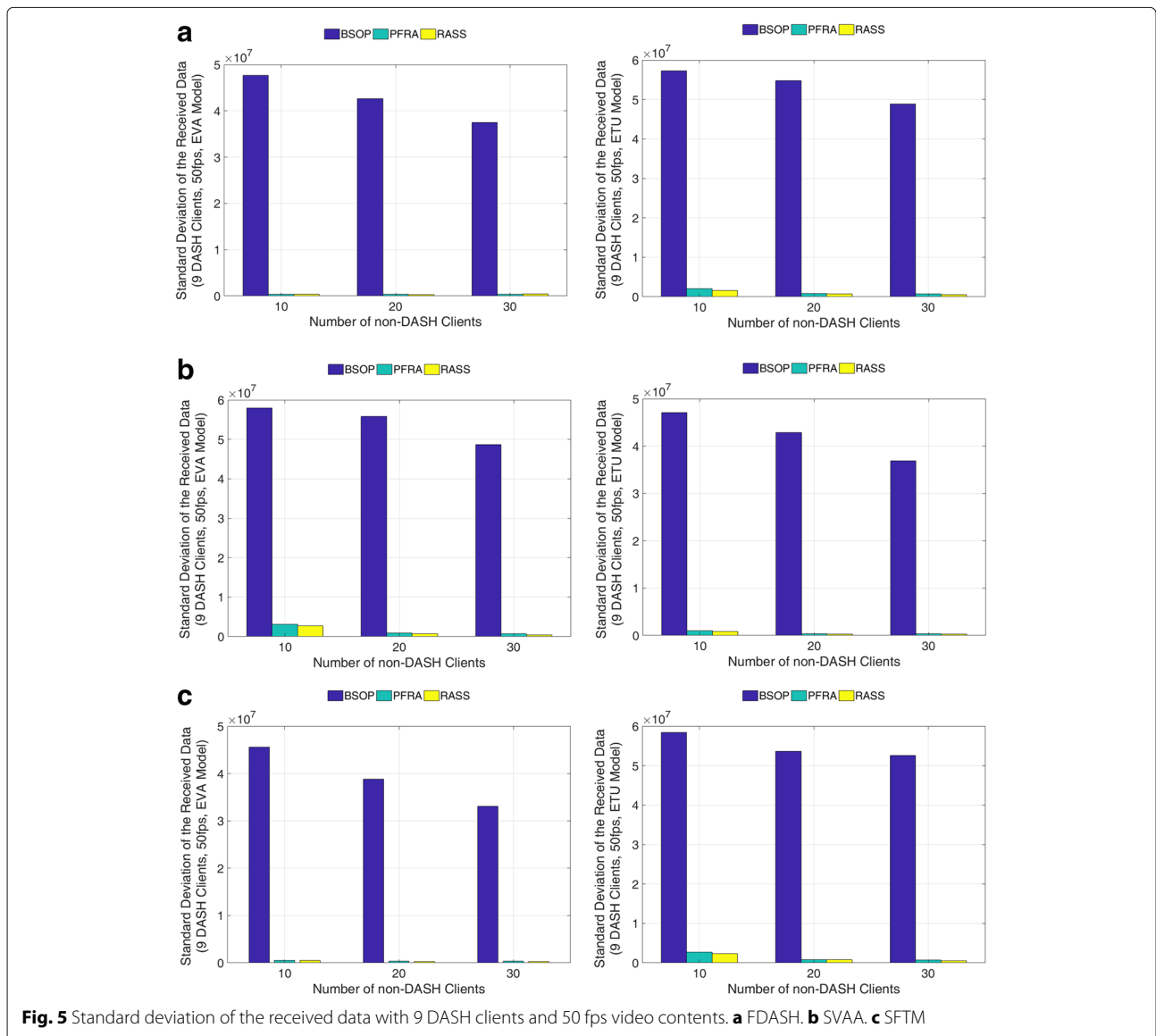

Fig. 4 Average received data of non-DASH clients with 9 DASH clients and 50 fps video contents. **a** FDASH. **b** SVAA. **c** SFTM

is encoded into 13 different bit-rate versions, as follows: $\Re$ = [278 Kbps, 384 Kbps, 522 Kbps, 791 Kbps, 1.033 Mbps, 1.245 Mbps, 1.547 Mbps, 2.134 Mbps, 2.484 Mbps, 3.079 Mbps, 3.527 Mbps, 3.840 Mbps, and 4.22 Mbps]. Also, to simplify the simulation, it is assumed that the traffic that the non-DASH clients received is non-real-time traffic (e.g., FTP, Web) [35]. The rest of the simulation parameters are described in Table 1.

6.2 Simulation results with three CSRAS and three resource allocation schemes

Tables 2, 3, and 4 show the re-buffering-time ratios of the DASH clients with the 50 frames per second (fps) video contents. As shown in the tables, BSOP scheme shows the poorest performance in all the simulation environments. As BSOP scheme uses information of buffer level

which only exists in DASH client, BSOP cannot operate effectively in the case that DASH and non-DASH clients exist in network simultaneously. Therefore, it is no wonder that BSOP shows poor and irregular performance even if there are few non-DASH clients in the LTE network (i.e., less harsh network). PFRA scheme shows more favorable performance than BSOP scheme since it considers fairness and allocates more resource blocks to users who have not gotten enough radio resource blocks. However, as the number of non-DASH clients increase, PFRA scheme should allocate more resource blocks to non-DASH clients. Therefore, it is natural that DASH clients experienced more re-buffering events as the number of non-DASH clients increases. On the other hand, with the operation of KCRDC and KCRA, RASS scheme shows no re-buffering events in almost every case. RASS

Fig. 5 Standard deviation of the received data with 9 DASH clients and 50 fps video contents. **a** FDASH. **b** SVAA. **c** SFTM

experiences A when there are 9 DASH clients and 30 non-DASH clients (2.117%, 1.715%, 0.412%) due to the large number of DASH clients and non-DASH clients. Nevertheless, RASS still shows the lowest re-buffering ratios among the three resource allocation schemes.

Figure 3 represents the simulation results within the QoE framework[4]. The QoE values are normalized by largest value in each case. BSOP provides lowest QoE to DASH clients in almost every case except three cases such as (a)—EVA, (b)—EVA, and (c)—EVA in Fig. 3. In those three cases, PFRA provides lowest QoE to DASH clients when there are 30 non-DASH clients since PFRA have to provide fairness to all the clients. RASS, in constrast with other two schemes, always provides highest QoE value since it manages well the re-buffering events of DASH clients as shown in Tables 2, 3, and 4. With these results, it is verified that by carefully managing re-buffering events of DASH clients, it is possible to increase QoE of DASH clients in harsh network environment.

Figures 4 and 5 show the *average received data* and the *standard deviation of the received data* of the non-DASH clients with in various environments. As shown in those figures, BSOP shows the highest *standard deviation of the received data*, i.e., it performed poorly in terms of the fairness. In contrast, RASS provides almost the same level of *standard deviation* with PFRA, i.e., it provides great fairness to non-DASH clients. Another point to note in the case of RASS is that the non-DASH clients received a lesser amount of data compared with the other resource allocation schemes. It is because RASS allocates more PRBs to the DASH clients to prevent the experiencing of the expected re-buffering events. However, it is okay for the non-DASH clients to receive data in a slightly more-slower manner since non-real-time traffic is much more tolerant than real-time traffic (i.e., video traffic) [36].

Furthermore, to show that RASS operates effectively regardless of the CSRAS type, the previous results need to be reviewed. If only RASS results in Table 2 and Figs. 3, 4, and 5 are considered, it is noticeable that the DASH clients experience the lowest number of re-buffering events regardless of the type of CSRAS. Furthermore, it is obvious that RASS provides a similar level of fairness to the non-DASH clients regardless of the type of CSRAS. However, there might be a question about the differences among the performances of RASS depending on the type of CSRAS. The differences among the performances are occurred because the segment bit-rates that are requested at a certain time and the timing of the request of a certain segment are different depending on the type of CSRAS. For example, let us assume that RASS provides more data to non-DASH clients in the case

**Table 5** Average re-buffering ratios of 3 DASH clients with 40 fps video contents

Resource Allocation Algorithm	Number of non-DASH client	Re-buffering ratio (%)					
		FDASH		SVAA		SFTM	
		EVA	ETU	EVA	ETU	EVA	ETU
BSOP	10	19.5144	14.544	16.056	18.535	9.904	10.599
	20	16.84	17.181	25.923	36.097	11.093	12.74
	30	15.979	24.399	21.835	38.679	22.868	10.854
PFRA	10	0	0	0	0	0	0
	20	0	0	0	0	0	1.093
	30	0	0	0	0	0	0
RASS	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	0	0	0	0	0	0

Table 6 Average re-buffering ratios of 6 DASH clients with 40 fps video contents

Resource Allocation Algorithm	Number of non-DASH client	Re-buffering ratio (%)					
		FDASH		SVAA		SFTM	
		EVA	ETU	EVA	ETU	EVA	ETU
BSOP	10	5.274	3.839	4.145	4.358	1.753	2.603
	20	2.687	2.623	6.497	2.194	3.619	2.278
	30	2.978	2.457	4.385	2.335	3.573	3.091
PFRA	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	0	0	0	0	0	0.262
RASS	10	0	0	0	0	0.696	0
	20	0	0	0	0	0	0
	30	0	0	0	0	0	0

Table 7 Average re-buffering ratios of 9 DASH clients with 40fps video contents

Resource Allocation Algorithm	Number of non-DASH client	Re-buffering ratio (%)					
		FDASH		SVAA		SFTM	
		EVA	ETU	EVA	ETU	EVA	ETU
BSOP	10	1.566	1.8322	2.648	1.976	0.849	1.601
	20	1.734	1.711	1.546	1.799	1.506	1.861
	30	1.626	1.387	1.36	1.526	2.885	2.22
PFRA	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	0.061	0	0.043	0	4.612	0.698
RASS	10	0	0	0	0	0	0
	20	0	0	0	0	0	0
	30	0	0	0	0	0	0

of SVAA compared with the case of SFTM. In this context, two cases exist. First, SFTM chooses the segments of the higher bit-rate compared with SVAA. At the cost of a higher bit-rate, SFTM maintains lower DASH-client buffer levels than SVAA. Second, SFTM maintains higher DASH-client buffer levels than SVAA with similar segment bit-rates. In both cases, the SFTM clients use more PRBs than the SVAA clients, i.e., the DASH clients use different amounts of PRBs depending on the type of CSRAS. Therefore, the only focal point is that RASS provides a similar level of the *average received data* amount and a similar level of the *standard deviation of the received data* for the non-DASH clients regardless of the type of CSRAS.

Not only the simulation results with 50 fps video contents, but also simulation results with 40 fps video contents have been made. Tables 5, 6, and 7 show the re-buffering ratioss of DASH clients with 40 fps video

contents. As shown in those tables, BSOP shows highest re-buffering ratios similarly with the results with 50 fps video contents. However, unlike the results with 50 fps video, PFRA shows low re-buffering ratios because 40 fps video requires 1 frame per 25 msec while 50 fps video requires 1 frame per 20 msec. In other words, re-buffering events occur frequently with higher fps video contents due to the higher number of frames in 1 sec.

The *average received data* and the *standard deviation of the received data* of the non-DASH clients in the simulations with 40 fps video contents are outlined in Figs. 6, 7, and 8. Same as the Figs. 3, 4, and 5, only the results with 9 DASH clients are represented in those figures. The three resource allocation schemes shows similar tendency compared to the results in Figs. 4 and 5. In other words, BSOP provides lowest QoE to DASH clients except a few cases that PFRA provides lowest QoE. Also,

Fig. 6 Average quality of experience (QoE) of 9 DASH clients with 40 fps video contents. **a** FDASH. **b** SVAA. **c** SFTM
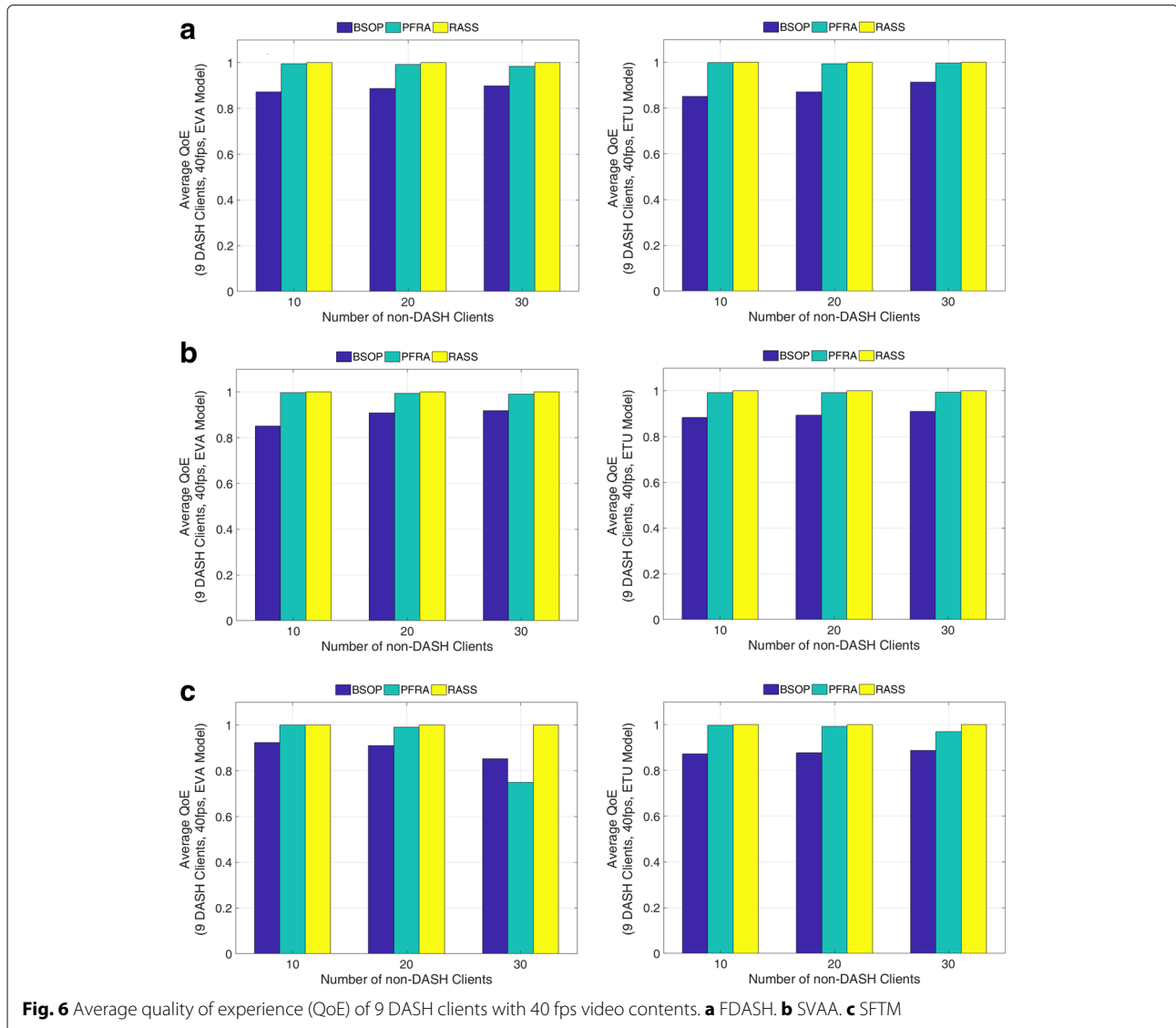
non-DASH clients under RASS received less data than BSOP and PFRA due to the mechanism to prioritize the decrease of re-buffering ratios. Also, RASS shows low *standard deviation* while BSOP shows bad performance in *standard deviation.*
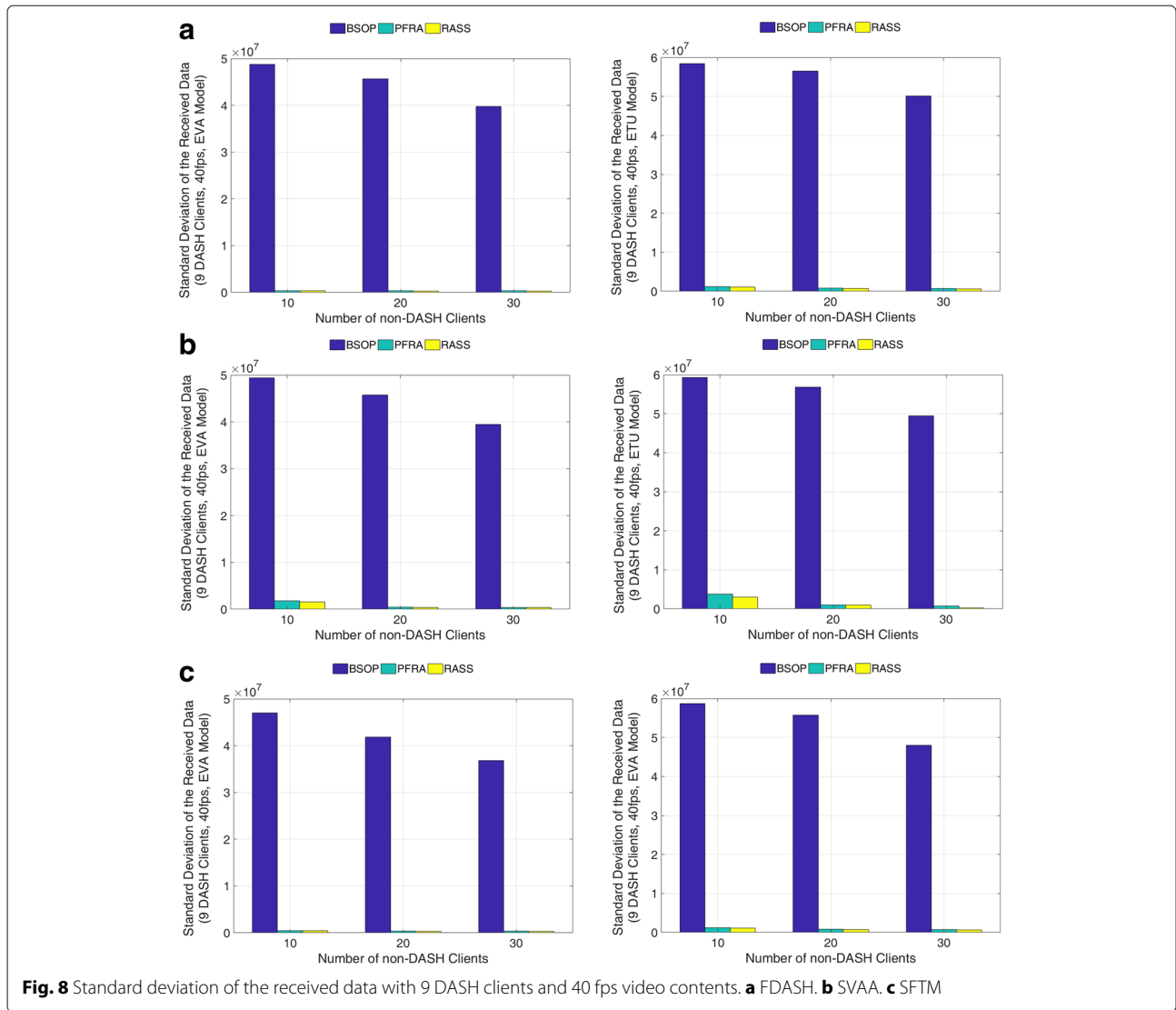
7   Conclusion

In this paper, we proposed a new resource allocation scheme for symbiosis of DASH clients and non-DASH clients (RASS) to provide seamless video streaming service to DASH clients and to guarantee fairness among non-DASH clients. KCRDC divides the total number of PRBs by considering various factors such as the number of DASH and non-DASH clients, the DASH-client buffer levels, and the maximum achievable bit-rates of all clients. Likewise, KCRA allocates the resource blocks to DASH

clients in consideration of the buffer levels, maximum achievable bit-rates, and past average throughput of the DASH clients.

A comparison between the proposed scheme and other resource allocation schemes was performed in various LTE environments with two fading models and a path-loss propagation. With the simulation results, it was verified that RASS scheme provides higher QoE to DASH clients than other schemes and shows almost same fairness compared to PFRA scheme. Also, RASS scheme transparency was proved by a demonstration of a similar tendency regarding the fairness of non-DASH clients and the re-buffering ratios of DASH clients. In summary, the obtained results showed that the proposed scheme operates very effectively in various LTE environments regardless of the type of client-side rate-adaptation scheme.

Fig. 7 Average received data of non-DASH clients with 9 DASH clients and 40 fps video contents. **a** FDASH. **b** SVAA. **c** SFTM
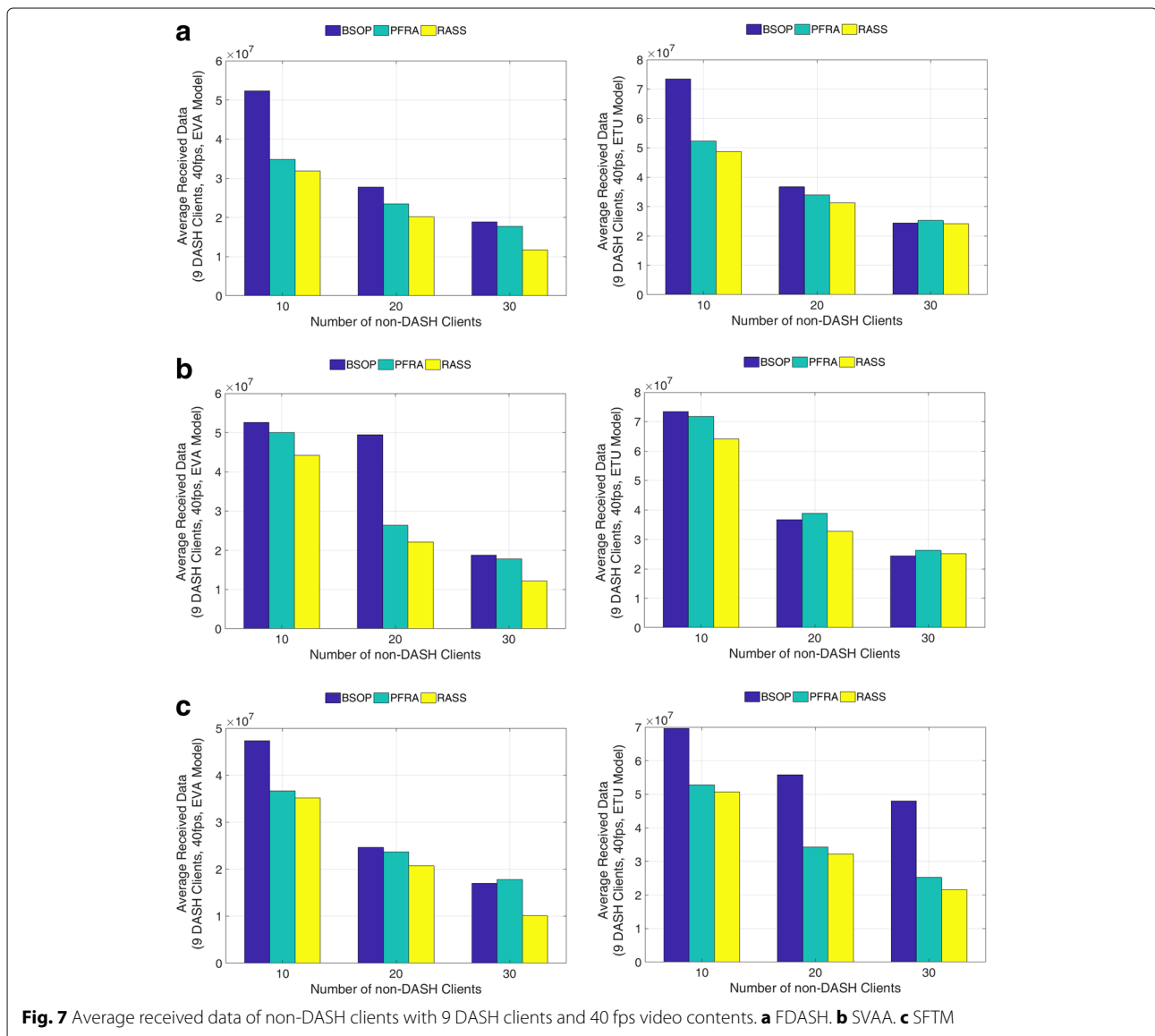
Fig. 8 Standard deviation of the received data with 9 DASH clients and 40 fps video contents. **a** FDASH. **b** SVAA. **c** SFTM

Endnotes

[1] Also, the α_1 and α_2 used in this optimization solution represent the proportion of the total PRBs that the DASH clients and non-DASH clients will use. Reviewer 1's comment 1.

[2] Moreover, among the three evaluation items (4)–(6), only re-buffering time ratio is considered by using $B_i[t]$. Reviewer 1's comment 5.

[3] Same as KCRDC, only re-buffering time ratio is considered in the optimization process by using $B_i[t]$. Reviewer 1's comment 5.

[4] Unlike the results of the Tables 2, 3, and 4, there are only results of 9 DASH clients in Figs. 3, 4, and 5 since lower number of DASH Clients does not impact the results. Reviewer 1's comment 2.

References
1. Cisco, Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper (2017)
2. ISO/IEC 23009-1:2014, Information technology – Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats
3. ISO/IEC 23009-2:2017, Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 2: Conformance and reference software
4. ISO/IEC 23009-3:2015, Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 3: Implementation guidelines
5. ISO/IEC 23009-4:2013, Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 4: Segment encryption and authentication
6. ISO/IEC 23009-5:2017, Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 5: Server and network assisted DASH (SAND)
7. ISO/IEC 23009-6:2017, Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 6: DASH with server push and WebSockets
8. R. Kwan, C. Leung, J. Zhang, Proportional fair multiuser scheduling in LTE. IEEE Signal Process. Lett. **16**(6), 461–464 (2009)
9. T. Girici, C. Zhu, J.R. Agre, A. Ephremides, Proportional fair scheduling algorithm in OFDMA-based wireless systems with QoS constraints. J. Commun. Netw. **12**(1), 30–42 (2010)
10. D. Zhou, N. Baldo, M. Miozzo, in *Proceedings of the 6th International ICST Conference on Simulation Tools and Techniques, SimuTools '13*. Implementation and Validation of LTE Downlink Schedulers for Ns-3 (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering (ICST), Brussels, 2013), pp. 211–218. http://dl.acm.org/citation.cfm?id=2512734.2512763
11. A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, H. Zhang, Developing a predictive model of quality of experience for internet video. ACM SIGCOMM Comput. Commun. Rev. **43**(4), 339–350 (2013)
12. C. Chen, L.K. Choi, G. de Veciana, C. Caramanis, R.W. Heath, A.C. Bovik, in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. A dynamic system model of time-varying subjective quality of video streams over HTTP (IEEE, 2013), pp. 3602–3606
13. K. Xiao, S. Mao, J.K. Tugnait, in *Global Communications Conference (GLOBECOM), 2016 IEEE*. QoE-Driven Resource Allocation for DASH over OFDMA Networks (IEEE, 2016), pp. 1–6
14. S. Kumar, A. Sarkar, A. Sur, A resource allocation framework for adaptive video streaming over LTE. J. Netw. Comput. Appl. **97**, 126–139 (2017)
15. K. Ivesic, L. Skorin-Kapov, M. Matijasevic, Cross-layer QoE-driven admission control and resource allocation for adaptive multimedia services in LTE. J. Netw. Comput. Appl. **46**, 336–351 (2014)
16. A. El Essaili, D. Schroeder, E. Steinbach, D. Staehle, M. Shehada, QoE-based traffic and resource management for adaptive HTTP video delivery in LTE. IEEE Trans. Circ. Syst. Video Technol. **25**(6), 988–1001 (2015)
17. J. Wu, C. Yuen, B. Cheng, Y. Yang, M. Wang, J. Chen, Bandwidth-efficient multipath transport protocol for quality-guaranteed real-time video over heterogeneous wireless networks. IEEE Trans. Commun. **64**(6), 2477–2493 (2016)
18. M. Zhao, X. Gong, J. Liang, W. Wang, X. Que, S. Cheng, QoE-driven cross-layer optimization for wireless dynamic adaptive streaming of scalable videos over HTTP. IEEE Trans. Circ. Syst. Video Technol. **25**(3), 451–465 (2015)
19. M.S. Bazaraa, H.D. Sherali, C.M. Shetty, *Nonlinear programming: theory and algorithms*. (Wiley, 2013)
20. F.H. Clarke, A new approach to Lagrange multipliers. Math. Oper. Res. **1**(2), 165–174 (1976)
21. The network simulator - ns-3. http://www.nsnam.org/. Accessed 25 May 2017
22. A.G. Al-Sakkaf, S. Khan, K. Abdullah, in *Communications (MICC), 2015 IEEE 12th Malaysia International Conference on*. QoS downlink schedulers in LTE towards 5G network (IEEE, 2015), pp. 225–229
23. G. Tian, Y. Liu, in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*. Towards agile and smooth video adaptation in dynamic HTTP streaming (ACM, 2012), pp. 109–120
24. C. Liu, I. Bouazizi, M.M. Hannuksela, M. Gabbouj, Rate adaptation for dynamic adaptive streaming over HTTP in content distribution network. Signal Process. Image Commun. **27**(4), 288–311 (2012)
25. D.J. Vergados, A. Michalas, A. Sgora, D.D. Vergados, P. Chatzimisios, FDASH: A Fuzzy-Based MPEG/DASH Adaptation Algorithm. IEEE Syst. J. **10**(2), 859–868 (2016)
26. T. Berners-Lee, R. Fielding, H. Frystyk, Hypertext transfer protocol–HTTP/1.0 (1996)
27. A. Ghosh, R. Ratasuk, B. Mondal, N. Mangalvedhe, T. Thomas, LTE-advanced: next-generation wireless broadband technology. IEEE Wirel. Commun. **17**(3) (2010)
28. D. Astély, E. Dahlman, A. Furuskär, Y. Jading, M. Lindström, S. Parkvall, LTE: the evolution of mobile broadband. IEEE Commun. Mag. **47**(4) (2009)
29. R. Love, R. Kuchibhotla, A. Ghosh, R. Ratasuk, W. Xiao, B. Classon, Y. Blankenship, in *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*. Downlink control channel design for 3GPP LTE (IEEE, 2008), pp. 813–818
30. 3GPP TS 36.213, E-UTRA Physical layer procedures (2008)
31. A. Saul, S. Khan, G. Auer, W. Kellerer, E. Steinbach, in *Communications, 2007. ICC'07. IEEE International Conference on*. Cross-layer optimization with model-based parameter exchange (IEEE, 2007), pp. 5701–5707
32. J. Jiang, V. Sekar, H. Milner, D. Shepherd, I. Stoica, H. Zhang, in *NSDI*. CFA: A Practical Prediction System for Video QoE Optimization., (2016), pp. 137–150
33. C. Chen, X. Zhu, G. de Veciana, A.C. Bovik, R.W. Heath, Rate adaptation and admission control for video transmission with subjective quality constraints. IEEE J. Sel. Top. Sig. Process. **9**(1), 22–36 (2015)
34. X. Yin, A. Jindal, V. Sekar, B. Sinopoli, A control-theoretic approach for dynamic adaptive video streaming over http. ACM SIGCOMM Comput. Commun. Rev. **45**(4), 325–338 (2015)
35. K.B. Johnsson, D.C. Cox, in *Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd*. QoS scheduling of mixed priority non-real-time traffic, vol. 4 (IEEE, 2001), pp. 2645–2649
36. D.D. Kouvatsos, *Traffic and Performance Engineering for Heterogeneous Networks*, vol. 1. (River Publishers, 2009)