Efficient Multi-View 3D Video Multicast with Depth-Image-Based Rendering in LTE-Advanced Networks with Carrier Aggregation

Yu-Chun Chen, De-Nian Yang, *Senior Member, IEEE,* Jitang Lee, and Wanjiun Liao, *Fellow, IEEE*

Abstract—With the recent emergence of naked-eye 3D mobile devices and various 3D-enabled laptops, service providers now afford the opportunity to provide mobile 3D video streaming in LTE-Advanced networks. Differing from traditional single-view 3D videos, multi-view 3D videos allow users to choose preferred view angles and thus are promising for new applications, such as free-viewpoint television (FTV). Nevertheless, enabling multi-view 3D video services may overwhelm the network resource when transmitting all views of every video. Fortunately, Depth-Image-Based Rendering (DIBR) allows each mobile client to synthesize the desired view from a nearby left view and right view, so that not all views of a video are necessarily transmitted. A new challenge with DIBR, however, is to carefully choose the transmitted views to limit the video distortion and minimize the bandwidth consumption. In this paper, therefore, we first formulate a new optimization problem, called View and MCS Selection (VMS) Problem, to minimize the bandwidth consumption for multi-view 3D video multicast in LTE networks. An algorithm, called View and MCS Aggregation (VMAG) is proposed to find the optimal solution to VMS. For Carrier Aggregation (CA) in LTE-Advanced networks, we formulate a new View, MCS and Carrier Selection (VMCS) Problem and prove that the problem is NP-Hard. We first design a dynamic programming algorithm, called the View Assignment with MCS and Carrier (VAMC) algorithm, to find the optimal solution for small instances. We then propose the View and MCS Aggregation with Carrier (VMAGC) algorithm based on VMAG to effectively find the near optimal solution to VMCS. The simulation results show that bandwidth consumption can be effectively reduced by over 30% in VMS and VMCS.

Index Terms—LTE-Advanced, multi-view 3D video, multicast, carrier aggregation, depth-image-based rendering

1 Introduction

DATA traffic, particularly video, has grown tremendously over mobile broadband networks in the past few years. Responding to this important need, Multimedia Broadcast Multicast Service (MBMS) was proposed by 3GPP Release 9 and 10 to satisfy the increasing requirements of multimedia applications for a group of users. This MBMS standard on LTE-A is also called evolved MBMS (eMBMS) [1]. However, most literature on LTE-l video multicast considers only single-view 2D videos.

With the recent emergence of naked-eye 3D mobile devices, such as HTC EVO 3D, LG Optimus 3D, and Sharp Lynx, together with numerous 3D-enabled laptops, service providers are now positioned with a new opportunity to enable 3D video streaming on mobile clients in cellular networks. In contrast to traditional stereo 3D video formats with a single-view, multi-view 3D videos provide users with more options on the viewing angles and have the potential for various applications, such as free-viewpoint television (FTV), movies, education, location-based advertisements, etc. While different users can enjoy their preferred views, the network load is expected to significantly increase with multi-view videos (typically with 16 views [2]).

- *Yu-Chun Chen, Jitang Lee, and Wanjiun Liao are with the Department of Electrical Engineering, National Taiwan University, Taiwan. E-mail: r01942052@ntu.edu.tw, r03942049@ntu.edu.tw, wjliao@ntu.edu.tw.*
- *De-Nian Yang is with the Institute of Information Science and Research Center of Information Technology Innovation, Academia Sinica, Taiwan. E-mail: dnyang@iis.sinica.edu.tw.*

One promising way to remedy the bandwidth issue is to exploit Depth-Image-Based Rendering (DIBR) in mobile clients. Because adjacent views usually share many similar parts, the desired view of a user can be synthesized from one nearby left view and one nearby right view transmitted in the network. Several schemes on bit allocation between the texture and depth map [3], [4], as well as the rate control with layered encoding for a multi-view 3D video [5], have been proposed to ensure that the quality of the synthesized view is very close to the original view (i.e., by minimizing the total distortion or maximizing the quality). With DIBR, it is no longer necessary to deliver every view of a multi-view video in the network, while the computation overhead incurred from DIBR is sufficiently small to enable use on current mobile devices [5], [6].

However, new research challenges have emerged for efficient multi-view 3D video multicast in LTE-A networks. One challenge is that the number of views between the left and right transmitted views needs to be constrained to maintain the quality of the synthesized view [7]. In other words, since each transmitted view is shared by multiple users, it is crucial to carefully select the transmitted views so that the desired view of each user can be synthesized with good quality. DIBR has a quality constraint [7], which specifies that the left view and right view are allowed to be at most R views away (i.e., $R-1$ views between them), to ensure that every view between the left view and right view can be successfully synthesized with good quality. To support more multi-view videos in LTE-A networks, it is also necessary to minimize the number of selected views for each video according to the requests of the views from

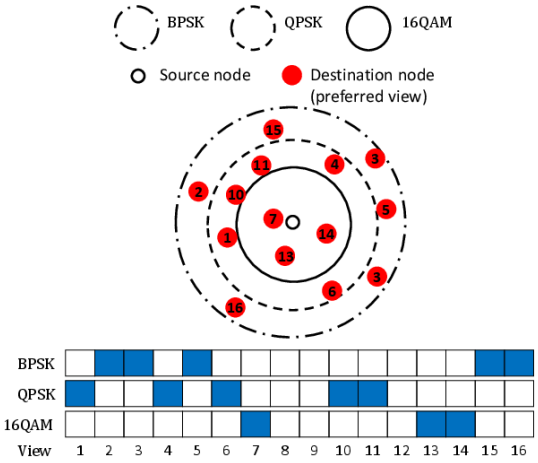


Fig. 1: An illustrative example with 16 views and 3 MCSs (BPSK, QPSK, and 16QAM).

users. Another research challenge is that in LTE-A networks, the number of resource blocks involved to deliver a view to each user is different due to adaptive modulation and coding (AMC). Ideally, the goal is to deliver a view with a higher rate to a user under good channel conditions so as to reduce the number of resource blocks, but the multicast view can no longer serve as the right or left view in the view synthesis for another user under poor channel conditions. It is therefore a challenge to efficiently multicast multi-view videos in LTE-A networks because both the views as well as the modulation and coding schemes (MCS) must necessarily be carefully examined at the same time.

Nevertheless, to the best of our knowledge, there is no related research on *multi-view 3D video* multicast over *wireless* network. For instance, the energy efficiency of video multicast in 4G wireless system with Scalable Video Coding(SVC) and Adaptive Modulation and Coding(AMC) is proposed [8]. Flexible transport of 3D videos [9] for digital video broadcasting(DVB) is studied to improve the video rates and resolutions. However, the above works focus on conventional single-view 3D videos, and the correlation of different views in a multi-view video has not been considered. Moreover, for IP or P2P multicast, an efficient approach [10] is proposed to deliver multi-view 3D videos in application layer multicast(ALM) trees for the clients that frequently switch the desired views. They also attempt to limit the interview delay variation for providing synchronized views for each client. Multi-View video streaming over a multi-tree peer-to-peer(P2P) network with the NUEPMuT protocol is designed to address the issue of the ungraceful departure of peers [11], while the interactivity of clients is also considered [12]. Nevertheless, the above works focus on the transmission over wired IP network, instead of the latest LTE network with carrier aggregation. Also, the correlation between nearby views has not been exploited.

Fig. 1 presents an illustrative example with 16 views and 3 MCSs (BPSK, QPSK, and 16QAM), where the preferred view is marked beside each user, and R is 3. The conventional way is to deliver each view separately with the existing LTE-A multicast, where the MCS of each view is determined from the farthest node requesting the view. As shown in Fig. 1, views 2, 3, 5, 15, and 16 are encoded by BPSK, views 1, 4, 6, 10, and 11 are encoded by QPSK, and views 7, 13, and 14 are encoded by 16QAM. All requested views are involved in the transmissions. In contrast, a more efficient way is to exploit DIBR to reduce the number of transmitted views. By multicasting only views 1, 4, 7, 13 and 16 by BPSK and view 10 by QPSK, views 2, 3, 5, and 15 in BPSK can be synthesized by DIBR. View 6, 11 in QPSK and view 14 in 16QAM can be synthesized as well. Note that view 1 is transmitted by BPSK, instead of QPSK, in order to simultaneously satisfy the users preferring view 1 and other users preferring view 2, where view 1 in this case is regarded as the left view for synthesizing view 2. Also, view 13 in BPSK, which originally was not requested by any user, is transmitted so that all users preferring views 13-16 can be jointly served. Compared with the conventional approach with existing LTE-A multicasting, it is important to effectively reduce the number of transmitted views by carefully examining the preferred view and MCS of each user.

Based on the above observations, we make a first attempt to study bandwidth-efficient view and MCS selection in multi-view video multicasting for LTE-A networks. We thus formulate a new optimization problem, called View and MCS Selection Problem (VMS) to minimize the resource block consumption for multicasting a multi-view video stream in LTE-A networks. We design an algorithm, called View and MCS Aggregation (VMAG), to find the optimal solution to VMS. For Carrier Aggregation (CA) in 3GPP LTE-A networks [13] (which aggregates multiple Component Carriers (CC) of smaller bandwidths and allow user equipment (UE) to employ multiple CCs to increase data rate), previous studies have focused on resource allocation in CCs to increase overall throughput [14], [15] or frequency-selective power allocation [16]. Some work has also selected CCs according to the carrier capability of each user and the corresponding circumstances [17], [18], and several studies have discussed packet scheduling by prioritizing the allocation of resources to users with good channel quality [19], [20]. Nevertheless, multi-view 3D multicast with CA has not yet been explored in the literature.

New research challenges arise for multi-view 3D multicast with CA in LTE-A networks. Firstly, different coverage is provided on different CCs due to various path losses and interferences. Therefore, each user has a different channel condition in different frequency bands, which means that the number of involved resource blocks to deliver a desired view to a user changes with different frequency bands. A second research challenge is that in a multi-carrier LTE-Advanced system, both LTE-Advanced and LTE users tend to co-exist in the network. LTE-A users can be assigned to all CCs, whereas LTE users support transmissions for only one CC [15]. Therefore, it is necessary to choose one CC for each LTE user, and it is also crucial to assign views to suitable CCs such that both LTE and LTE-A users are able to acquire the requested views accordingly.

To support CA in LTE-A networks, we extend VMS and formulate a new optimization problem, called the View, MCS and Carrier Selection (VMCS) Problem, to minimize the resource block consumption for multicasting a multi-view video stream. A delay constraint is also incorporated here for multi-view 3D videos. We prove that VMCS is NP-Hard. For small instances of VMCS, we present a dynamic programming algorithm, named the View Assignment with

MCS and Carrier (VAMC) algorithm, to find the optimal solution with exponential time. Based on VAMC, we then propose the algorithm View and MCS Aggregation with Carrier (VMAGC) to effectively find the near optimal solution to VMCS in polynomial time.

The rest of the paper is organized as follows. Section II describes the system model and formulates the VMS problem. In Section III, we present the VMAG algorithm and prove that it can find the optimal solution. In Section IV, we formulate the VMCS problem and prove that it is NP-hard, with algorithms VAMC and VMAGC then being explained in detail. Section V presents the simulation results, and we conclude this paper in Section VI.

2 System Model and VMS Problem

This paper considers a single-cell point-to-multipoint transmission in the eMBMS of LTE-A [1]. A 3D video in multi-view plus depth is encoded by simulcast encoding [21] or real-time dependent encoding [22]. Simulcast encoding neglects the correlations among nearby views and encodes each view separately. However, because adjacent views are inclined to share many similar parts, dependent encoding offers high compression efficiency by only encoding the parts different from adjacent views. The rationale behind spatial correlation exploited in dependent encoding is similar to the temporal correlation widely adopted in current video coding standards. Also, the idea of DIBR is similar by inferring and synthesizing the parts that are different from nearby views, while effective techniques are proposed to ensure the video quality [23], [24]. In this paper, we assume that each user prefers one view, while a user preferring multiple views can be modelled as multiple users with different views. The left view and right view received by a user may be transmitted in a different MCS, since LTE-A allows the user to receive the packets from a eNodeB with a different MCS [25].

More specifically, let M denote the set of modulation and coding schemes (MCS) in LTE. A higher MCS index supports a higher data rate and thus requires fewer resource blocks to transmit a view. Let $\tau_{m,v}$ denote the number of resource blocks to transmit view v in MCS m. Let D be the set of multicast users of a multi-view 3D video. Let V denote the set of views in a multi-view video, and the desired view of each user d in D is v_d. View v_d can either be transmitted directly or be synthesized from one nearby left transmitted view and one nearby right transmitted view. The quality constraint in DIBR ensures that there are at most $R-1$ views between them, and R can be set according to [7], where R can be different for each video.

In the following, we formulate VMS for efficient multi-view 3D video multicast in LTE-A networks. Given $M, V, D, R, \tau_{m,v}$, and v_d for each user d in D, the VMS problem is to select a set of transmitted views in V and assign an MCS for each selected view, so that each user d can receive v_d directly or synthesize v_d from a received left view and right view according to the quality constraint, where d can receive a view if the view is not transmitted with a higher MCS not decodable by d. The objective function of VMS is to minimize the number of resource blocks involved in the transmissions of all selected views of the multi-view 3D video. In Appendix A, we present an Integer Programming formulation for VMS.

TABLE 1: Notations for VMAG

Notation	Description
M	The set of Modulation and Coding Schemes (MCS).
V	The number of views in a multi-view 3D video.
R	The DIBR quality constraint.
D	The set of multicast users of a multi-view 3D video.
v_d	The desired view of each user $d \in D$.
$\tau_{m,v}$	The number of resource block to transmit view v in MCS m.
$\mu^m_{i,j}$	The minimum number of resource blocks to transmit some views in the candidate range from view i to view j with MCS 1, ..., m, to satisfy the user located in MCS 1, ..., m with preferred views from view i to view j.

3 Algorithm VMAG

In this section, we design algorithm VMAG to solve the VMS problem. Algorithm VMAG includes two phases: *Initialization* and *Aggregation*. The first phase initializes and identifies the service ranges (i.e., a continuous range of views to be served) for each MCS, while the second phase aggregates multiple service ranges in adjacent MCSs into one aggregated range and assigns the transmitted views. We also explore the possibility to aggregate adjacent ranges located in the same MCS. In Appendix B, we present the pseudo codes of VMAG. Later in this section, we will prove that VMAG can find the optimal solution to VMS. Table 1 represents the notations in VMAG.

3.1 Initialization Phase

Initialization Phase first identifies the service ranges of views to be covered by the transmitted views in each MCS according to the following two observations. The first is that for any preferred view v of each user in MCS m, when we serve the user by transmitting either v directly or nearby left and right views, every other user who also prefers v but who located from MCS $m+1$ to M can be served at the same time. Therefore, VMAG in this case incorporates view v from MCS m to M in the service range. In Fig. 1, since view 5 in BPSK is preferred by at least one user, view 5 is also located in the service ranges of BPSK, QPSK and 16QAM. The second observation is that for any two preferred views, say, view i and j in the same MCS, if there are at most $R-1$ views between them, all views from view i to j are in the same service range of the MCS. When we directly transmit view i and j, any views between them can be served by the two views thanks to DIBR. On the other hand, when view i or j is synthesized from other views, it is necessary to transmit a view between view i and j, and other views between i or j can thereby be served at the same time.

Example. Fig.1 presents an example with $R = 3$, given that view 7 and 10 are 3 views away in 16QAM, the views from view 7 to 10 are all in the same service range. Fig.2 shows the service ranges in each MCS, and there are 2, 3, and 1 service ranges in BPSK, QPSK, and 16QAM, respectively.

However, transmitting only the views in the service ranges does not guarantee finding the optimal solution to VMS. Fig.2 manifests that transmitting view 13 in BPSK is more efficient because it can be exploited to synthesize the views from view 11 to 15 in all MCSs, but view 13 is not in the service range of BPSK. Based on this observation, it is necessary to examine a wider range for each service range in order to find the optimal solution. In this paper, for each service range from view i to j, we define the *candidate range*

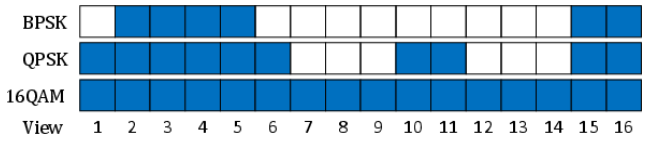


Fig. 2: An example of *Initialization Phase*.

of each service range as the one from $i-R+1$ to $j+R-1$. In Fig.2, for the service range from view 10 to 11 in QPSK, its candidate range is from view 8 to 13. For the service range from view 15 to 16 in BPSK and QPSK, its candidate range is from view 13 to 16. View 13 is overlapped by the two candidate ranges, and aggregating the two candidate ranges by transmitting view 13 can potentially reduce resource consumption because the view can serve the users with the required views in its left and right service ranges.

Note that if a service range in MCS m overlaps another one in MCS $m+1$ in at least a view, it is impossible for the candidate range of MCS m to start earlier or end later than the candidate range of MCS $m+1$, because the above two situations contradict the first observation in *Initialization Phase*. In other words, if a service range in MCS m overlaps another service range in MCS $m+1$, the candidate range in MCS m must be a subset of the candidate range in MCS $m+1$. Therefore, the candidate ranges always become wider for a larger MCS, as shown in Fig.2.

3.2 Aggregation Phase

In *Initialization Phase*, a view v transmitted at MCS m can serve v in MCS $m+1$ and even view $v+1$ in MCS $m+1$ with $R > 1$, when it appears in two candidate ranges. Therefore, this phase aggregates multiple candidate ranges both *vertically* (in adjacent MCS) and *horizontally* (in the same MCS), in order to identify the minimum number of resource blocks to satisfy all users.

More specifically, the goal of this phase is to derive $\mu^m_{i,j}$, which denotes the minimum number of resource blocks to transmit some views in the candidate range from view i to j with MCS $1, ..., m$, in order to satisfy the users located in MCS $1, ..., m$ with the preferred views located in the range from view i to j. In other words, the users located in MCS $m+1, .., M$ are not considered by $\mu^m_{i,j}$, and $\mu^m_{i,j}$ also does not examine the views outside the range from view i to j. Note that both view i and j are required to be transmitted when we derive $\mu^m_{i,j}$, because here it is not assumed that any view outside the range from view i to j is transmitted.

Algorithm VMAG systematically derives $\mu^{m_1}_{i_1,j_1}$ according to $\mu^{m_2}_{i_2,j_2}$ with the following three cases in Fig. 3, where $m_1 - 1 \le m_2 \le m_1$, $j_2 \le j_1$, and $i_2 \ge i_1$. The first case is *No Aggregation*, which presents the fundamental case to find the minimum number of resource blocks for a candidate range in MCS m, where the range does not overlap any candidate range in MCS $m-1$. *Vertical Aggregation* is the second case (Fig. 3b and 3c), and it examines the case in which the candidate range in MCS m overlaps at least one candidate range in MCS $m-1$, and those candidate ranges in MCS $m-1$ do not overlap each other. The final case is *Vertical and Horizontal Aggregation*(Fig. 3d), in which the candidate range in MCS m overlaps at least two candidate ranges in MCS $m-1$, and those candidate ranges in MCS $m-1$ also overlap each other. View 13 in Fig. 2 will be examined in this case.

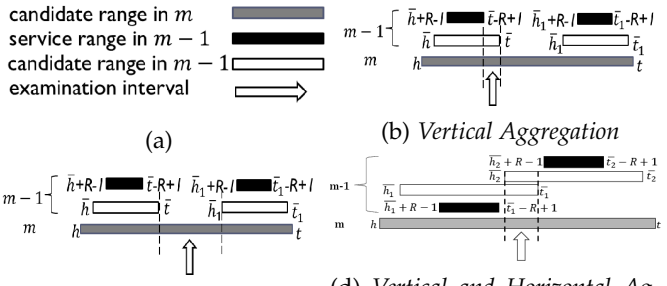


Fig. 3: *Aggregation Phase*

Algorithm VMAG derives $\mu^m_{i,j}$ from MCS 1 to MCS M, while in each MCS the smaller i and j will be investigated first. In the following, we explain the ideas behind the VMAG algorithm.

1) MCS 1

Since MCS 1 is the smallest MCS, *No Aggregation* is considered here. For each candidate range from view h to t in MCS 1, we derive $\mu^1_{h,h}, ..., \mu^1_{h,j}, ..., \mu^1_{h,t}$ step-by-step with dynamic programming as follows, where $\tau_{1,j}$ denotes the number of resource blocks to transmit view j in MCS 1, $h \le j \le t$.

$$\mu^1_{h,j} = \min_{\substack{(j-R)\le k<j \\ k\ge h}} (\tau_{1,j} + \mu^1_{h,k}). \quad (1)$$

Example. Fig.4 presents an example for VMAG with $R = 3$. Assume that BPSK, QPSK, and 16QAM respectively require 4, 3, and 2 resource blocks to transmit a view. *No Aggregation* is applied for all the candidate range in BPSK. Since $\mu^{BPSK}_{13,13} = 4$, we have $\mu^{BPSK}_{13,16} = 8$ by (1). *No Aggregation* is also applied for the candidate range from view 8 to 13 in QPSK, and $\mu^{QPSK}_{8,11}$, $\mu^{QPSK}_{9,11}$, $\mu^{QPSK}_{10,11}$, $\mu^{QPSK}_{9,12}$, $\mu^{QPSK}_{10,12}$, and $\mu^{QPSK}_{10,13}$ are all identical to 6, but $\mu^{QPSK}_{8,12}$, $\mu^{QPSK}_{9,13}$, and $\mu^{QPSK}_{8,13}$ are 9 because one additional view is required.

For $\mu^1_{h,j}$, view j needs to be transmitted because we do not assume that any view outside the range from view i to j has been transmitted. We exploit each $\mu^1_{h,k}$ with $k < j$ derived previously to find $\mu^1_{h,j}$ accordingly. Note that view k is also transmitted when we consider $\mu^1_{h,k}$, and therefore $j - R \le k$ must hold to ensure that every view from k to j can be synthesized successfully.

Recall that the service range of the above candidate range (from h to k) is from view $h+R-1$ to $t-R+1$. We also derive $\mu^1_{h,t-R+1}, ..., \mu^1_{h,t-1}$, instead of only $\mu^1_{h,t}$, for other MCS described later to consider the case when one of the views from $t-R+1$ to $t-1$ acts as the "bridge" (i.e., transmitted view) of the candidate range to connect to another candidate range located rightward. Similarly, in addition to view h, it is possible that one of the views from $h+1$ to $h+R-1$ acts as the bridge for the candidate range to connect to another candidate range leftward. Therefore, in addition to $\mu^1_{h,h}, ..., \mu^1_{h,t}$, it is necessary to derive $\mu^1_{\bar{h},\bar{h}}, ..., \mu^1_{\bar{h},\bar{t}}$ for every $\bar{h}$ with $h < \bar{h} \le h+R-1$ and every $\bar{t}$ with $t-R+1 \le \bar{t} < t$. Note that in the three phases *No Aggregation, Vertical Aggregation*, and *Vertical and Horizontal Aggregation*, instead of examining all views, only the boundary views in *service range* and *candidate range* are considered.

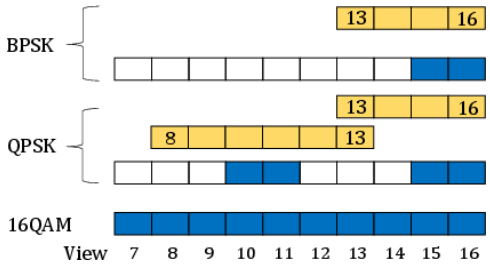


Fig. 4: An example for VMAG.

2) MCS m, $1 < m \leq M$

For each candidate range in MCS m, if it overlaps no candidate range in MCS $m-1$, (i.e., *No Aggregation*), we can handle it in the same way as the strategy mentioned in MCS 1. Recall that in *Initialization Phase*, we explain that if two service ranges overlap by at least one view, the candidate range in a smaller MCS must be a subset of the candidate range in a larger MCS, as shown in Fig. 3b and 3c. In the following, we explain *Vertical Aggregation* and *Vertical and Horizontal Aggregation* in detail.

Fig.3b illustrates *Vertical Aggregation*. For *Vertical Aggregation*, if the candidate range from h to t in MCS m overlaps the candidate range from $\bar{h}$ to $\bar{t}$ in MCS $m-1$, $h \leq \bar{h} \leq \bar{t} \leq t$, we exploit $\mu_{i,j}^{m-1}$ derived previously find $\mu_{i,j}^{m}$ with dynamic programming in Fig.3b, where $\bar{h} \leq i \leq \bar{h}+R-1$ and $\bar{t}-R+1 \leq j \leq \bar{t}$.

$$\mu_{h,j}^{m} = \min_{\bar{h} \leq i \leq \bar{h}+R-1} (\mu_{i,j}^{m-1} + \mu_{h,i}^{m} - \tau_{m,i}). \tag{2}$$

Example. Fig. 4, *Vertical Aggregation* is applied for the candidate views from 13 to 16 in QPSK. $\mu_{13,16}^{QPSK}$, $\mu_{14,16}^{QPSK}$ and $\mu_{15,16}^{QPSK}$ are all 8 due to *Vertical Aggregation* that transmits two views in BPSK to cover the need in both BPSK and QPSK. It is also applied for 16QAM to derive $\mu_{7,13}^{16QAM} = \mu_{7,10}^{16QAM} + \mu_{10,13}^{QPSK} - \tau_{16QAM,10} = 4+6-2 = 8$ according to (2). Note that view 10 in the equation is the "bridge" view i in (2), helping to connect two view ranges in QPSK and 16QAM.

More specifically, in order to derive $\mu_{h,j}^{m}$, we examine the range to view j from every possible view i in MCS $m-1$, which belongs to the candidate range in MCS m. The rationale behind *Vertical Aggregation* is to leverage the optimal allocation of the views to be transmitted in the range from i to j in MCS $m-1$ to support the users in MCS m. Note that every view i with $\bar{h} \leq i \leq \bar{h}+R-1$ needs to be examined to find the best "bridge" i between the candidate range in MCS $m-1$ and MCS m leading to the minimum number of resource blocks. Note that $\tau_{m,i}$ is subtracted here because we no longer need to transmit view i in MCS m since it has been supported by MCS $m-1$.

Moreover, in Fig.3c, for the interval after $\bar{t}$, we can derive $\mu_{h,j}^{m}$ in the way similar to (1), where $j > \bar{t}$, because the benefit of aggregating the candidate range of MCS m has been summarized in $\mu_{h,k}^{m}$ calculated before $\mu_{h,j}^{m}$.

$$\mu_{h,j}^{m} = \min_{j-R \leq k \leq j} (\tau_{m,j} + \mu_{h,k}^{m}). \tag{3}$$

$\mu_{h,k}^{m}$ can be derived from (2) when $\bar{t}-R+1 \leq k \leq \bar{t}$.

Note that (3) can apply not only for $j > \bar{t}$ but also for $\bar{t}-R+1 \leq j \leq \bar{t}$, where the latter one indicates that the most efficient aggregation has been achieved in k before j. Therefore, for $\bar{t}-R+1 \leq j \leq \bar{t}$, both cases described by (2) and (3) must be considered to derive $\mu_{h,j}^{m}$ in *Vertical Aggregation*. Thus, algorithm VMAG sets $\mu_{h,j}^{m}$ as the minimum one obtained from (2) and (3).

Fig.3d illustrates *Vertical and Horizontal Aggregation*. For *Vertical and Horizontal Aggregation*, the candidate range from h to t in MCS m overlaps at least two candidate ranges in MCS $m-1$, one is from $\bar{h}_1$ to $\bar{t}_1$, the other is from $\bar{h}_2$ to $\bar{t}_2$, and both of which also overlap each other such that $h \leq \bar{h}_1 \leq \bar{h}_2 \leq \bar{t}_1 \leq \bar{t}_2 \leq t$. In the overlapping area, we exploit the previously derived $\mu_{h,o}^{m}$ and μ_{o,j_2}^{m-1} to find μ_{h,j_2}^{m} with dynamic programming in Fig.3d, where $\bar{h}_2 \leq o \leq \bar{t}_1$ and $\bar{t}_2-R+1 \leq j_2 \leq \bar{t}_2$.

$$\mu_{h,j_2}^{m} = \min_{\bar{h}_2 \leq o \leq \bar{t}_1} (\mu_{h,o}^{m} + \mu_{o,j_2}^{m-1} - \tau_{\bar{m},o}). \tag{4}$$

Example. The two candidate ranges in QPSK overlap in view 13 in Fig. 4, VMAG exploit *Vertical and Horizontal Aggregation* and obtain $\mu_{7,16}^{16QAM} = \mu_{7,13}^{16QAM} + \mu_{13,16}^{QPSK} - \tau_{QPSK,13} = 8+8-3 = 13$. Note that $\mu_{7,13}^{16QAM}$, which is derived from *Vertical Aggregation*, can be exploited here to consider the first candidate range in QPSK, while $\mu_{13,16}^{QPSK}$ corresponds to the second candidate range that is served by transmitting view 13 and 16 in BPSK. Also note that $\tau_{\bar{m},13} = 3$ because $\bar{m} = QPSK$, and $\tau_{QPSK,13}$ represents view 13 originally transmitted in the first candidate range. However, since view 13 also needs to be transmitted by the second candidate range with BPSK, we no longer need to transmit the one in the higher MCS.

Note that $\mu_{h,o}^{m}$, which corresponds to the first candidate range in MCS $m-1$, is derived according to *Vertical Aggregation* explained above, while μ_{o,j_2}^{m-1} corresponds the second candidate range in MCS $m-1$. The role of view o is similar to view i in (2). View o acts as the best "bridge" to connect the candidate range in MCS m and the two candidate ranges in MCS $m-1$. However, the third term $\tau_{\bar{m},o}$ in (4) is different from that in (2). For $\mu_{h,o}^{m}$, it is possible that view o is transmitted in any MCS spanning from MCS 1 to MCS m because $\mu_{h,o}^{m}$ is a solution after considering multiple MCSs. This is also similar for view o in μ_{o,j_2}^{m-1}. Therefore, MCS $\bar{m}$ in $\tau_{\bar{m},o}$ is the higher MCS transmitted in $\mu_{h,o}^{m}$ and μ_{o,j_2}^{m-1}.

It is worth noting that employing only *Vertical and Horizontal Aggregation* may not generate the best solution, especially when the "horizontal bridge"(i.e., view o in (4)) requires a large number of resource blocks. In this case, the candidate range of MCS m will overlap the two candidate ranges of MCS $m-1$ individually in the way described in *Vertical Aggregation* with two "vertical bridges"(i.e., view k in (3)), and we do not need to horizontally aggregate the two candidate ranges in MCS $m-1$ due to the expensive bridge in MCS $m-1$. Therefore, both *Vertical Aggregation* and *Vertical and Horizontal Aggregation* must be examined in order to obtain the minimum $\mu_{h,j}^{m}$. Similarly, *Vertical Aggregation* and *Vertical and Horizontal Aggregation* are also applied when there are more than two candidate ranges in MCS $m-1$, because $\mu_{h,j}^{m}$ is derived in ascending order of j.

3) MCS $M+1$

To find the solution for MCS $1, ..., M$, we add a virtual MCS $M+1$, with $\tau_{(M+1),v} = 0$ for every view v. The rationale behind this is to exploit the above three strategies (*No Aggregation*, *Vertical Aggregation*, and *Vertical and Hor-*

izontal Aggregation) to find $\mu_{1,V}^{M+1}$. Since MCS $M+1$ does not consume resource blocks, $\mu_{1,V}^{M+1}$ corresponds to the total number of resource blocks for MCS $1, \cdots, M$ to support all users.

4) Selection of the Transmitted views and MCS

Since the above three strategies follow dynamic programming, VMAG selects the views to be transmitted and the corresponding MCS of each view to be transmitted by recursively applying the *arg min* operation in equations $(1)-(4)$. For example, if the optimal μ_{h,j_2}^m is derived according to (4) and $o^* = \arg\min_{\bar{h}_2 \le o \le \bar{t}_1} (\mu_{h,o}^m + \mu_{o,j_2}^{m-1} - \tau_{\bar{m},o})$, VMAG transmits views h and j_2 in MCS m and view o^* in the lower MCS in μ_{h,o^*}^m and μ_{o^*,j_2}^{m-1}. Algorithm VMAG then processes μ_{h,o^*}^m and μ_{o^*,j_2}^{m-1} recursively in the same way.

Example. In the example, according to (4), VMAG will transmit view $h=7$, $j_2=16$, and $o=13$ in 16QAM, QPSK, and QPSK respectively. However, view 13 and 16 are not transmitting in QPSK. The MCS to transmit view 13 and 16 is determined according to (2), and VMAG then assigns BPSK to view 13 and 16. Therefore, the optimal solution in this example is 13 resource blocks, with view 7 to be transmitted in 16QAM, view 10 in QPSK, and view 13, 16 both in BPSK.

3.3 Solution Optimality

In the following, we prove by induction that $\mu_{1,V}^{M+1}$ is the minimum number of resource blocks involved in the optimal solution.

Theorem 1. *VMAG finds the optimal solution to VMS.*

Proof. Initially, for MCS 1, $\tau_{1,h}$ appears to be the optimal solution for $\mu_{h,h}^1$. Assume that $\mu_{h,k}^1$ is the optimal solution for the range from h to k. Therefore, $\mu_{h,j}^1$ in (1), which identifies the k with the smallest $\mu_{h,k}^1$, is also the optimal solution.

For MCS m, if a candidate range does not overlap any other candidate range with a smaller MCS, the situation is identical to MCS 1 above, and the solution for the candidate range in MCS m is optimal. For *Vertical Aggregation* of a candidate range from h to $\bar{t}$ in MCS $m-1$, notice that we examine $\mu_{\bar{h},\bar{t}}^{m-1}$ but also $\mu_{i,j}^{m-1}$ for every $\bar{h} \le i \le \bar{h}+R-1$ and $\bar{t}-R+1 \le j \le \bar{t}$. In other words, every view from the boundary of the candidate range to the boundary of the service range is considered in i and j, and in (2), every possible i and j located in the candidate range in MCS $m-1$ is examined to find the best way for *Vertical Aggregation*. Afterwards, the result is exploited to find the optimal solution for the candidate range in MCS m in (3) in the same way as (1) in MCS 1. Since VMAG derives $\mu_{h,j}^m$ an ascending order of j by exploiting the existing optimal solution, VMAG can find the optimal solution incrementally in MCS m by induction.

For *Vertical and Horizontal Aggregation*, every view o overlapped by the two candidate ranges in MCS $m-1$ is examined to ensure that the best "bridge" o can be identified. More importantly, *Vertical and Horizontal Aggregation* is not always the best way for the two candidate ranges to overlap in MCS $m-1$. Algorithm VMAG also explores the case of the two candidate ranges being aggregated individually to the candidate range in MCS m. In other words, $\mu_{h,j}^m$ is computed in both ways, i.e., *Vertical and Horizontal Aggregation* and *Vertical Aggregation*, and the smaller result is assigned to $\mu_{h,j}^m$. Therefore, VMAG explores every possibility to leverage the existing results in MCS $m-1$.

Note that MCS M can have multiple candidate ranges, while it is possible to aggregate any two neighboring candidate ranges. To explore the above possibility, we add a new virtual MCS $M+1$ and set $\tau_{M+1,v}$. In other words, the optimal solution can transmit every view in MCS $M+1$, but those views are incapable of supporting any user in MCS $1, \cdots, M$. Therefore, *Vertical and Horizontal Aggregation* explores every possibility to aggregate and two neighboring candidate ranges in MCS M. In the above strategy cannot reduce the number of resource blocks, *Vertical Aggregation* can find the optimal solution of each candidate range in MCS M. Therefore, $\mu_{1,V}^{M+1}$ is the optimal solution to VMS. □

3.4 Complexity Analysis

In the worst case scenario, each MCS has $\lfloor V/R \rfloor$ service ranges. In MCS 1 of the *Aggregation Phase*, each candidate range includes at most V views, and each view is required to examine at most R cases in (1). Therefore, a maximum of $(\lfloor V/R \rfloor) \times V \times R = V^2$ iterations are involved to find the optimal solution in MCS 1. From MCS 2 to M of the *Aggregation Phase*, each candidate range has at most V views, and each view corresponds to at most $2R$ cases in the derivation of (2)-(4). Therefore, a maximum of $(\lfloor V/R \rfloor) \times V \times 2R = 2V^2$ iterations are involved to find the solution for each MCS thereafter. The virtual MCS $M+1$ contains only one candidate range, and each view in the range needs to examine at most $2R$ cases in (2)-(4). Therefore, it needs at most $V \times 2R = 2VR$ iterations to find the solution in the virtual MCS $M+1$. Moreover, due to $R \le V$, we can find $VR \le V^2$ in every case. Therefore, the complexity of the algorithm is $O(V^2 + (M-1) \times 2V^2 + 2VR) = O(MV^2 + VR) = O(MV^2)$.

4 Multi-Carrier Investigation

In this section, we explore multi-view 3D video multicasting for LTE-A networks with Carrier Aggregation (CA). CA in 3GPP enables a more flexible allocation of wireless resources by aggregating multiple Component Carriers (CC) of small bandwidths for UE to increase the data rate. Three types of CA are identified for different CCs [13], [26]. The first category is intra-band contiguous, which provides a wider contiguous bandwidth in the same band. The second type is intra-band non-contiguous, which represents the non-contiguous spectrum in the same band, whereas the middle carriers are loaded with other users. The last type is inter-band non-contiguous, which allows multiple carriers on different bands to be aggregated. Each CC contains different path loss and shadow fading for each user, and the channel conditions of carriers for a user in the inter-band non-contiguous type alter more dramatically than the channel conditions of carriers for a user in the intra-band contiguous type. All types of CA can be supported in this paper because the channel condition for a user derived from its location and carrier frequency can be specified differently in each CC. It should be mentioned that both LTE-A and LTE users may co-exist in the same network, where an LTE-A user can receive the information simultaneously from all CCs, but an LTE user can only select one CC at each time determined by eNodeB.

In the following, we formulate VMCS based on the VMS for multi-view 3D video multicasting in LTE-A networks with CA. Let $\mathcal{C}$ be the set of CCs, and C be the number of CCs in the multi-carrier system. The VMCS problem is to select a set of views to be transmitted in V and assign a CC and an MCS for each selected view, so that each user d can receive v_d directly or synthesize v_d by DIBR from a nearby left view and right view. Nevertheless, each LTE user d cannot synthesize v_d from a nearby left view and right view transmitting in different CCs. The objective function of VMCS is to minimize the number of resource blocks involved in the transmissions of all selected views for the multi-view 3D video. Moreover, a delay constraint T corresponding to Δ_c slots is included in VMCS, which means that the number of resource blocks transmitting in carrier c cannot exceed Δ_c. In Appendix C, we present an Integer Programming formulation for VMCS, and we prove that VMCS is NP-hard in Appendix D. In Appendix E, we first present algorithm VAMC to find the optimal solution to the small instances of VMCS, thus making VAMC a baseline approach for the comparison of different algorithms in Section V. In the following, we focus on the design of a practical heuristic algorithm to solve VMCS.

TABLE 2: Extra notations in VMAGC

Notation	Description
$\mathcal{C}$	The set of Component Carriers (CC).
C	The number of Component Carriers.
Δ_c	The delay constraint in carrier c.
$\Phi = \{1, ..., c\}$	The carrier list sorted in ascending order of the resource consumption.
$\Theta(c)$	The set of candidate transmitted views for carrier c.
$\Theta(k; c)$	The set of candidate transmitted views for carrier k by comparing the resource change between carrier k and c.
$\Theta(c, t)$	The t-th candidate transmitted view in $\Theta(c)$.
$\omega_a(k, c)$	The resource change of view a between carrier k and c.
$\omega_{i,j}(k, c)$	The resource change from view i to j between carrier k and c.
$M(c, a)$	The assigned MCS for sending view a in carrier c.
$\Pi(c)$	The views set to be transmitted in carrier c from VMAG.
$\Pi(c, l)$	The l-th view on the set.
$b(a, c)$	The view which is requested by LTE users and is nearest to view a in the range from view x to $a-1$, where view x is the closet left view of a in carrier c.
$f(a, c)$	The view which is requested by LTE users and is nearest to view a in the range from view $a+1$ to y, where view y is the closet right view of a in carrier c.
$Z_k(\lvert\Pi(c)\rvert, \Delta_k)$	The maximal resource change between carrier k and c.
$\bar{\Theta}(c, t)$	The view for $\Theta(c, t)$ to exchange.
$C(\Theta(c, \bar{t}))$	The serving carrier of $\Theta(c, \bar{t})$.
$\lambda(\Theta(c, t))$	The difference resource change for exchanging $\Theta(c, t)$ with $\Theta(c, \bar{t})$.
$\Gamma(\Theta(c, t))$	The carrier which has the maximal resource change between itself and carrier c.

4.1 Heuristic Algorithm for VMCS

Inspired by VMAG in Section III, we design a new algorithm, named View and MCS Aggregation with Carrier (VMAGC), to solve VMCS by selecting and aggregating the solution to VMAG in each carrier. Due to different channel conditions for each user in various carriers, VMCS can be split into multiple sub-problems, where each sub-problem corresponds to a carrier. VMAGC first assigns the desired views of each user to different carriers, and VMAG is then exploited to find the views to be transmitted with the corresponding MCSs in each carrier. More specifically, VMAGC processes a carrier at each iteration with three phases. The first phase is *Search Phase*, which estimates the resource consumption of each carrier according to VMAG and extracts the carrier to be processed first. The second phase is *Maximization Phase*, which assigns the views for the carrier selected from *Search Phase*. The last phase is *Examination Phase*, which finds the the corresponding MCSs of transmitted views. VMAGC stops when all carriers are examined. Table 2 summarizes the notations in VMAGC, and Appendix F presents the pseudo codes.

4.1.1 Search Phase

This phase first exploits VMAG to estimate the resource consumption of each carrier for all desired views and then generates the carrier list. The carrier list $\Phi = \{1, \cdots, c\}$ is sorted in ascending order of the resource consumption, and carrier c requires the most resources, which as a result implies that it can support the least number of views. Note that some LTE-A users may be able to receive packets from only a few carriers due to the channel quality. In VMAGC, for the users that can only receive packets from carrier c, *necessary transmitted views* desired by those users need to be included in the transmission of carrier c. The resource consumption of necessary transmitted views is first derived before entering the *Maximization Phase*. If the resource consumption of the necessary transmitted views in carrier c exceeds the resources that carrier c can support, VMAGC will terminate due to the violation of the delay constraint. Otherwise, the VMAGC enter *Maximization Phase*.

4.1.2 Maximization Phase

The goal of this phase is to find the set of *candidate transmitted views* for carrier c. Let $\Theta(c)$ denote the set of views desired by at least one user to be served in carrier c. To find $\Theta(c)$ with the minimal resource consumption, this phase tries to move some views in carrier c to another carrier k to reduce the resource consumption in c. At the beginning of the phase, $\Theta(c)$ is equal to the set of views to be transmitted acquired by VMAG. If some views consume the fewest resources in c than in other carriers, these views to be transmitted stay in $\Theta(c)$. For other candidate transmitted views in $\Theta(c)$, the *resource change* from carrier c to k is derived. More specifically, $\omega_a(k, c)$ represents the amount of resource consumption that can be reduced in carrier c if view a is moved from c to another carrier k, because this view can be delivered with higher modulation due to better channel quality. This phase aims to maximize the *resource change* in carrier c and transfer some views from carrier c to other carriers $\{1, 2, ..., c-1\}$, by investigating $\omega_a(1, c)$, $\omega_a(2, c)$,...,$\omega_a(c-1, c)$ for every view a. Since the channel condition of each user tends to vary in different carriers, three cases are examined to evaluate the resource change.

a) Solution Split

Fig. 5a presents an example of this case, where the red rectangles represent the views to be transmitted acquired by VMAG. The resource change of view 1 is $\omega_1(1, 2) = \tau_{i,1} - \tau_{j+1,1}$, and the resource change of view 7 is $\omega_7(1, 2) = \tau_{i,7} - \tau_{j+1,7}$. Because the serving range of view 2 and 5 in carrier 1 will be served only by view 4 in carrier 2, the resource change of view 2 is equal to view 5, and $\omega_2(1, 2) = \omega_5(1, 2) = \frac{[\tau_{i,4} - (\tau_{j,2} + \tau_{j,5})]}{2}$, where the resource consumption of view 4 is split to views 2 and 5.

b) Solution Merge

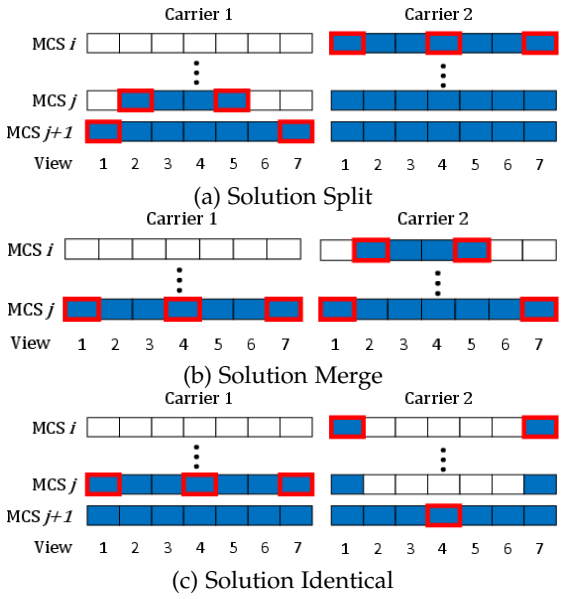


Fig. 5: Three cases of $\omega_a(k,c)$

Fig. 5b presents an example of this case. The resource change of view 1 is $\omega_1(1,2) = \tau_{j,1} - \tau_{j,1}$, and the resource change of view 7 is $\omega_7(1,2) = \tau_{j,7} - \tau_{j,7}$. Because the serving range of view 4 in carrier 1 will be jointly served by view 2 and 5 in carrier 2, the resource change of view 4 is $\omega_4(1,2) = (\tau_{i,2} + \tau_{i,5}) - \tau_{j,4}$, merged from the consumption of views 2 and 5.

c) Solution Identical

Fig. 5c presents an example of this case. The resource change of view 1 is $\omega_1(1,2) = \tau_{i,1} - \tau_{j,1}$, the resource change of view 4 is $\omega_4(1,2) = \tau_{j+1,4} - \tau_{j,4}$, and the resource change of view 7 is $\omega_7(1,2) = \tau_{i,7} - \tau_{j,7}$. Differing from the previous two cases, VMAG chooses the same view set to send in carrier 1 and 2, and we thereby are able to compare the views to be transmitted in carrier 1 and 2 one-by-one without merging or splitting.

The following example further explores the requirement of the LTE users, who can only select one carrier.

In VMAGC, $\Theta(k;c)$ is defined as the set of candidate transmitted views in carrier k by comparing the resource change between carrier k and c. Therefore, $\{\Theta(1;c), \Theta(2;c), \cdots, \Theta(c-1;c)\}$ can be derived when the resource change between carriers $\{1, \cdots, c-1\}$ and c is calculated. Afterward, $\Theta(c)$ is derived when $\{\Theta(1;c), \Theta(2;c), \cdots, \Theta(c-1;c)\}$ are all calculated. In VMAG, the minimal resource consumption of each carrier without the delay constraint can be derived as $\mu_{1,V}^{M+1}$, and we further extend the solution as $\mu_{1,V}(k)$, which is defined as the minimal resource consumption for serving all views from view 1 to V in carrier k without the delay constraint. Moreover, $\omega_{i,j}(k,c)$ is defined as the resource change of view range (i,j) from carrier c to carrier k, and $\tau_{M(c,i),i}$ represents the number of resource blocks for transmitting view i in carrier c with assigned MCS $M(c,i)$. We then have the following observation from the resource change.

Proposition 1. *Given a set of carriers $\{1, ..., c-1, c\}$, maximizing $\{\omega_{1,b}(1,c), ..., \omega_{1,b}(c-1,c)\}$ will determine the views to be transmitted with minimal resource consumption in carrier c if all carriers follow the delay constraint.*

Proof. We prove this proposition by contradiction. Suppose a view range $(1,a)$ is served in carrier c, and another view range (a,b) is served in carrier k in carrier set $\{1, ..., c-1\}$. However, the view range $(1,a)$ consumes fewer resources in carrier k than in carrier c, and view range (a,b) consumes fewer resources in carrier c than in carrier k. If view range $(1,a)$ is served in carrier k and view range (a,b) is served in carrier c, less resource consumption in range $(1,b)$ can be achieved because $\omega_{a,b}(k,c) < 0 < \omega_{1,a}(k,c)$. Therefore, $\mu_{1,a}(k) + \mu_{a,b}(c) < \mu_{1,a}(c) + \mu_{a,b}(k)$ holds, $\mu_{a,b}(c) - \mu_{a,b}(k) < \mu_{1,a}(c) - \mu_{1,a}(k)$ and $\omega_{a,b}(k,c) < \omega_{1,a}(k,c)$ also hold. For all carriers following the delay constraint, if $\{\omega_{1,b}(1,c), ..., \omega_{1,b}(c-1,c)\}$ is not maximized, there must exist some ranges to be exchanged with other ranges in carrier c that can achieve smaller resource consumption. □

After maximizing the resource change $\{\omega_{1,b}(1,c), ..., \omega_{1,b}(c-1,c)\}$, the set of candidate transmitted views for carrier c can be acquired by transferring some views from c to other carriers. Nevertheless, it is necessary to compare the resource change between every carrier in $\{1, ..., c-1\}$ and c because $\omega_a(1,c) \neq \omega_a(2,c) \neq \cdots \neq \omega_a(c-1,c)$, $1 \leq a \leq V$. Therefore, the resource change can be maximized by independently comparing each carrier in Φ with c.

Now, we define $\Pi(c)$ as a list of views that can be transmitted in carrier c acquired from VMAG, and $\Pi(c,l)$ is the l-th view on the list. Moreover, if view a is selected to be transmitted and there are some LTE users synthesizing their preferred views by view a, the resource change for view a needs to consider the extra transmitted views created by the LTE users, where *Extra transmitted views* are defined as the view which is required to be transmitted but not included in the solution in VMAG.

Proposition 2. *Given a view list $\Pi(c)$, there exists an LTE user requesting view a, $\Pi(c,l-1) < a < \Pi(c,l)$. After maximizing the resource change, if $\Pi(c,l-1)$ is transmitted in any carrier k but $\Pi(c,l)$ is not transmitted in carrier k, extra transmitted views are required to be transmitted in carrier k to serve view a for the LTE user.*

Proof. The LTE user desiring view a needs to receive the transmitted views $\Pi(c,l-1)$ and $\Pi(c,l)$ for synthesizing view a. However, the LTE user cannot receive the two views from different carriers. Therefore, view a must be synthesized from other views transmitted in carrier k, or view a needs to be transmitted in carrier k directly. In other words, *extra transmitted views*, which are transmitted between views $\Pi(c,l-1)$ and $\Pi(c,l)$, are necessary for the LTE user to receive view a. □

It is worth noting that extra transmitted views will reduce the resource change because it incurs more resource consumption to transmit a view in carrier k. Let view $b(a,c)$ denote the view requested by LTE users and is nearest to view a in the range from view x to view $a-1$, where view x is the closest left view of a in carrier c derived by VMAG. On the other hand, $f(a,c)$ is defined as the view requested by LTE users and is nearest to view a in the range from view $a+1$ to y, where view y is the closest right view of view a in carrier c from VMAG. Here, $\tau_{M(c,b(a,c)),b(a,c)}$ denotes

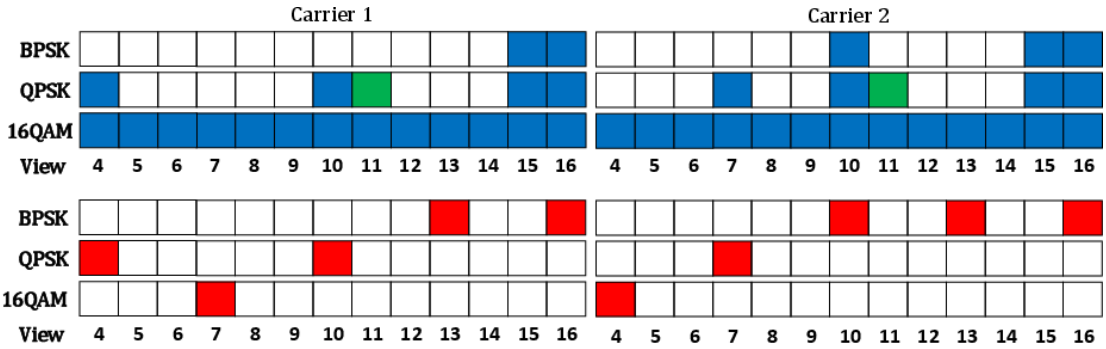


Fig. 6: An example for VMAGC.

the resource consumption for view $b(a,c)$ in carrier c when $b(a,c)$ cannot be served in carrier c. In addition, let

$$\begin{aligned}\omega_a(k,c)' =&\omega_a(k,c) - \tau_{M(c,b(a,c)),b(a,c)} \\ &- \tau_{M(c,f(a,c)),f(a,c)}.\end{aligned} \tag{5}$$

Example. Fig. 6 presents an illustrative example for VMAGC with $R=3$, where the green blocks represent the requests from LTE users, and the red blocks are the candidate transmitted views from VMAG. In this example, BPSK, QPSK, and 16QAM require 4, 3, and 2 resource blocks to transmit a view, respectively. The resource change of view 4 between carrier 1 and 2 is $\omega_4(1,2) = \tau_{16QAM,4} - \tau_{QPSK,4} = 2-3=-1$. The resource change of view 7 between carrier 1 and 2 is $\omega_7(1,2) = \tau_{QPSK,7} - \tau_{16QAM,7} = 3-2=1$. and the resource change of view 10 between carrier 1 and 2 is $\omega_{10}(1,2) = \tau_{BPSK,10} - \tau_{QPSK,10} = 4-3=1$. Also, the resource change of view 13 and 16 are $\omega_{13}(1,2)=0$ and $\omega_{16}(1,2)=0$, respectively. After considering the extra transmitted views in carrier 1, $\omega_{10}(1,2)' = \omega_{10}(1,2) - \tau_{BPSK,11} = 1-3=-2$, where $f(a,c)=f(10,2)=$ view 11. Also, $\omega_{13}(1,2)' = \omega_{13}(1,2) - \tau_{BPSK,11} = 0-3=-3$, where $b(a,c)=b(13,2)=$ view 11.

In (5), $\omega_a(k,c)$ can be updated based on *Proposition 2*. Here, $\omega_a(k,c)$ decreases because views $b(a,c)$ and $f(a,c)$ may not be served in carrier k, and views $b(a,c)$ and $f(a,c)$ have an opportunity to become extra transmitted views.

Note that the goal is to maximize the total resource change between carrier k and c. However, it is not possible to move all the views to carrier k due to the delay constraint in carrier k. The problem, which finds the views to be moved from carrier c to carrier k with the corresponding *resource change* and delay constraint in carrier k, is similar to the $0/1$ knapsack problem. Therefore, the dynamic programming algorithm [27] for $0/1$ knapsack problem can be employed here to find the maximal total resource change between carrier k and carrier c. However, $\omega_{\Pi(k,l)}(k,c)'$ may be negative, and thus the resource change for each view in $\Pi(c)$ needs to be normalized and modified as follows.

$$\begin{aligned}\omega_{\Pi(k,l)}(k,c)'' =&\omega_{\Pi(k,l)}(k,c)' \\ &- \min\{0, (min_{p\in\Pi(k)}\omega_p(k,c)') - 1\}.\end{aligned} \tag{6}$$

Eq. (6) guarantees that the new resource change for each view in $\Pi(k)$ is positive, since the value in the $0/1$ knapsack problem needs positive. In the following, we present the dynamic programming algorithm to maximize the resource change. Let $Z_c(|\Pi(k)|, \Delta_k)$ denote the maximal resource change between carrier k and c for the first N views, and $N=|\Pi(k)|$ here.

$$\begin{aligned}Z_c(N,\Delta_k) =& \max\{Z_c(N-1,\Delta_k), \\ & Z_c(N-1, \Delta_k - \tau_{M(k,\Pi(k,N)),\Pi(k,N))}) \\ & + \omega_{\Pi(k,N)}(k,c)''\}.\end{aligned} \tag{7}$$

$Z_c(N,\Delta_k)$ can be derived from (7). If the N-th view in $\Pi(k)$ is chosen, $\omega_{\Pi(k,N)}(k,c)''$ represents the gain, and $\tau_{M(k,\Pi(k,N)),\Pi(k,N)}$ is the resource consumption. The best carrier k can collaborate with the remaining views $\{\Pi(k,1),\Pi(k,2),...,\Pi(k,N-1)\}$, and delay constraint $\Delta_k - \tau_{M(k,\Pi(k,N)),\Pi(k,N)}$ is $Z_c(N-1,\Delta_k - \tau_{M(k,\Pi(k,N)),\Pi(k,N)})$. On the other hand, if the N-th view in $\Pi(k)$ is not chosen, the best carrier k can collaborate with the remaining views $\{\Pi(k,1),\Pi(k,2),...,\Pi(k,N-1)\}$, and the delay constraint Δ_k is $Z_c(N-1,\Delta_k)$.

Moreover, if views $b(\Pi(k,N),k)$ and $f(\Pi(k,N),k)$ requested by LTE users can be served in the same carrier after the N-th view in $\Pi(k)$ is chosen, the resource change according to (5) and (6) is necessary to be updated as follows

$$\begin{aligned}Z_c(N,\Delta_k) =& \max\{Z_c(N-1,\Delta_k), Z_c(N-1,\Delta_k \\ & - \tau_{M(k,\Pi(k,N)),\Pi(k,N))} + \omega_{\Pi(k,N)}(k,c)'' \\ & + \tau_{M(c,b(\Pi(k,N),k)),b(\Pi(k,N),k)} \times 2 \\ & + \tau_{M(c,f(\Pi(k,N),k)),f(\Pi(k,N),k)} \times 2\}.\end{aligned} \tag{8}$$

Eq. (7) can be defined again as (8), which increases the gain of $\Pi(k,N)$ because there is no extra transmitted views induced from $b(\Pi(k,N),k)$ and $f(\Pi(k,N),k)$ in carrier k. Through (7) and (8), the maximal resource change between carrier k and c can be derived accordingly. Since the above strategies follow dynamic programming, the algorithm selects the views to be transmitted and the corresponding MCSs in carrier k by recursively applying (7) and (8), and these views are padded to $\Theta(k;c)$. After the view selection for carrier k in the *Maximization Phase*, the demands on other carriers need to be updated because some demands have been satisfied in carrier k.

The next step is to calculate the resource change between carriers $k+1$ and c. *Maximization Phase* is recursively repeated until carriers $\{1,...,c-1\}$ have been independently compared to c, and the maximal resource change between carriers $\{1,...,c-1\}$ and carrier c is acquired, with $\{\Theta(1;c),\Theta(2;c),\cdots,\Theta(c-1;c)\}$ being derived accordingly. The remaining requested views that are not served in carriers $\{1,...,c-1\}$ can be served in carrier c, and the views

to be transmitted with the corresponding MCSs can be determined by VMAG to cover these remaining requested views. Note that the views remains in carrier c are regarded as the set of candidate transmitted views in $\Theta(c)$.

Example. Following the example in Fig. 6, after applying (6), we have $\omega_4(1,2)'' = -1 - \min\{0, -3-1\} = 1-(-4) = 3$, $\omega_7(1,2)'' = 1-(-4) = 5$, $\omega_{10}(1,2)'' = (-2)-(-4) = 2$, $\omega_{13}(1,2)'' = (-3)-(-4) = 1$, and $\omega_{16}(1,2)'' = 0-(-4) = 4$. Afterward, (7) and (8) show that the maximal resource change between carrier 1 and carrier 2 is $\omega_7(1,2)'' + \omega_{16}(1,2)'' = 9$, by selecting views 7 and 16 for carrier 1 and views 4, 10, and 13 for carrier 2.

4.1.3 Examination Phase

Nevertheless, after *Maximization Phase*, the resource consumption of $\Theta(c)$ may still exceed the delay constraint of carrier c. Therefore, the goal of this phase is to revise the views to be transmitted and the corresponding MCSs in carrier c to satisfy the delay constraint. This phase includes two approaches to adjust $\Theta(c)$: 1) exchanging some views in $\Theta(c)$ with other views in $\{\Theta(1;c), \Theta(2;c), ..., \Theta(c-1;c)\}$, and 2) moving some views in $\Theta(c)$ from carrier c to other carriers in $\{1, ..., c-1\}$ that still have resources.

a) Exchanging

First, $\Theta(c,t)$ is defined as the t-th candidate transmitted view in $\Theta(c)$, and $|\Theta(c)|$ is the number of candidate transmitted views in $\Theta(c)$. Here, our strategy is to exchange $\Theta(c,t)$ in carrier c with view $\Theta(c,\bar{t})$ that is also served in another carrier. $\Theta(c,\bar{t})$ can be derived from $\{\Theta(1;c), \Theta(2;c), ..., \Theta(c-1;c)\}$, and let $C(\Theta(c,\bar{t}))$ denote the serving carrier of $\Theta(c,\bar{t})$. The view exchange involves the follow four steps:

1. Note that the VMAGC exchanges a view because the resource consumption of $\Theta(c)$ exceeds the delay constraint of carrier c. The exchanging view needs to incur less resource consumption in carrier c. Therefore, the resource consumption of transmitting $\Theta(c,\bar{t})$ in carrier c is necessary to be smaller than the resource consumption of $\Theta(c,t)$ in carrier c.
2. The resource consumption of carrier $C(\Theta(c,\bar{t}))$ satisfies the delay constraint after $\Theta(c,t)$ is exchanged with $\Theta(c,\bar{t})$. That is, after the two views are exchanged, the delay constraint of carrier $C(\Theta(c,\bar{t}))$ should be still satisfied.
3. The difference of resource change, $\lambda(\Theta(c,t))$, for exchanging $\Theta(c,t)$ with $\Theta(c,\bar{t})$, is larger than the difference of resource change for exchanging $\Theta(c,t)$ with other views, where $\lambda(\Theta(c,t)) = \omega_{\Theta(c,t)}(C(\Theta(c,\bar{t})),c)' - \omega_{\Theta(c,\bar{t})}(C(\Theta(c,\bar{t}))),c)'$. That is, this step finds the $\Theta(c,\bar{t})$ for $\Theta(c,t)$ with the largest reduction of the resource consumption.
4. After every view in $\Theta(c)$ finds the corresponding view $\Theta(c,\bar{t})$ for exchange, the candidate exchanged view t^* with the maximal difference of resource change among $\Theta(c)$ and the corresponding $\Theta(c,\bar{t^*})$ are extracted.

The views to be transmitted in carrier c are adjusted through the above procedure until the delay constraint of carrier c is satisfied. However, if the resource consumption still exceeds the resources of carrier c after exchanging all candidate transmitted views with other views in $\{\Theta(1;c), \Theta(2;c), \cdots, \Theta(c-1;c)\}$, it implies that some candidate transmitted views in $\Theta(c)$ needs be moved to a carrier in $\{1, ..., c-1\}$ that still have resources.

b) Moving

The moving procedure consists of two steps. First, each candidate transmitted view $\Theta(c,t)$ finds its new carrier $\Gamma(\Theta(c,t))$, which has sufficient resources to transmit $\Theta(c,t)$. Moreover, $\Theta(c,t)$ has the maximal resource change between carrier $\Gamma(\Theta(c,t))$ and carrier c to reduce reduce the most resource consumption after we move view $\Theta(c,t)$ to carrier $\Gamma(\Theta(c,t))$. Second, after each view in $\Theta(c)$ finds its new carrier, the candidate moved view $\Theta(c,t^*)$, which has the maximal resource change between $\Gamma(\Theta(c,t^*))$ and c among $\Theta(c)$, is moved from carrier c to carrier $\Gamma(\Theta(c,t^*))$. Therefore, $\Theta(c)' = \Theta(c)/\Theta(c,t^*)$. The rationale behind the two steps follows*Proposition 2* and the delay constraint. If the resource consumption still exceeds the resources remaining in carrier c, the new set of candidate transmitted views, $\Theta(c)'$, is processed again by Exchanging.

The views to be transmitted and the corresponding MCSs in carrier c, $\Theta(c)$, are determined when the delay constraint of carrier c is satisfied based on *Proposition* 2. Then, the list of carriers is updated, $\Phi' = \Phi/c$, and *Search Phase*, *Maximization Phase*, and the *Examination Phase* are processed again for deriving the views to be transmitted of the next carrier. VMAGC stops after the views to be transmitted of all carriers are derived, and the minimal resource consumption in VMCS can be acquired from the summation of the resource consumption for the views to be transmitted in each carrier.

Example. Following the example in Fig. 6, here we assume that both carrier 1 and 2 have a delay constraint with 9 resource blocks. After *Maximization Phase*, the resource consumption in carrier 2 is $\tau_{16QAM,4} + \tau_{BPSK,10} + \tau_{BPSK,13} = 2+4+4 = 10$, which still exceeds its delay constraint. Therefore, it is necessary to exchange some views or move some views from one carrier to others. However, the VMAGC finds that after considering each candidate transmit view in carrier 2, the resource consumption in carrier 2 still exceed 8. Therefore VMAGC then tries to move some views to carrier 1. In $\Theta(2)$, only view 4 can be moved to carrier 1, because once view 10 is moved, view 13 will be moved due to the requirement of the LTE user, and vice versa. After view 4 is moved to carrier 1, the resource consumption in carrier 1 is $\tau_{QPSK,4} + \tau_{16QAM,7} + \tau_{BPSK,16} = 3+2+4 = 9$, and the resource consumption in carrier 2 is $\tau_{BPSK,10} + \tau_{BPSK,13} = 4+4 = 8$. Both carriers do not violate their delay constraint. Afterward, the solution returned by VMAGC in this example requires $9+8 = 17$ resource blocks.

4.2 Complexity Analysis

Algorithm VMAG with time complexity $O(MV^2)$ is the building block of VMAGC. There are at most C iterations involving the *Search Phase* and *Examination Phase* and C^2 iterations including the *Maximization Phase*. For each iteration of the Search Phase and Maximization Phase, C carriers are required to run VMAG in the worst case, and the overall time complexity is $O(C^3MV^2)$. In each iteration of the Maximization Phase, there are at most $\lceil (V-1)/R \rceil + 1$ views to be transmitted in each carrier, and each transmitted view needs to evaluate its resource change with the most $2MR$ iterations. For each iteration of the Maximization Phase, the time complexity of the dynamic programming is

TABLE 3: SIMULATION SETTING

Parameter	Value
#eNodeB	1
Carrier Frequency	2.0 GHz
Channel Bandwidth of CC	10 MHz
Number of CC	5
Slot Duration	0.5 ms
Resource blocks	50
OFDM data symbols	7
eNodeB Tx power	43 dBm
Subcarriers	12
Modulation and Coding Schemes	15 different defined in [28]
Distance Attenuation	*Macro cell propagation model-urban area [29]
Shadow Fading	Log-normal, $\sigma = 8$dB
Thermal Noise	-100dBm

*Path Loss(dB) $= 58.83 + 37.6log_{10}X + 21log_{10}f$, where X is the eNodeB-UE separation in kilo-meters and f is the carrier frequency in MHz.

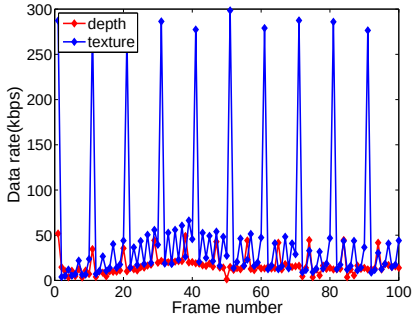


Fig. 7: Data trace for camera 10 of *Book_Arrival*

$(\lceil (V-1)/R \rceil + 1) \times \Delta_{max}$ in the worst case, where $\Delta_{max} = \max_{m \in C} \Delta_m$. There are at most $\lceil (V-1)/R \rceil + 1$ views in the candidate transmitted views for each iteration of the Examination Phase, and at most $(V-1)/R + 1$ views are examined from each candidate transmitted views, and each view should examine its resource change with at most $2MR$ iterations. Moreover, there are at most $(\lceil (V-1)/R \rceil + 1)^3$ iterations for Exchanging and Moving in each carrier. Therefore, the complexity of the algorithm is dominated by the Examination Phase, and the complexity is $O(CM(V^6/R^4))$. In Appendix I, we measured the average computation time for VMAGC.

5 Simulation Results

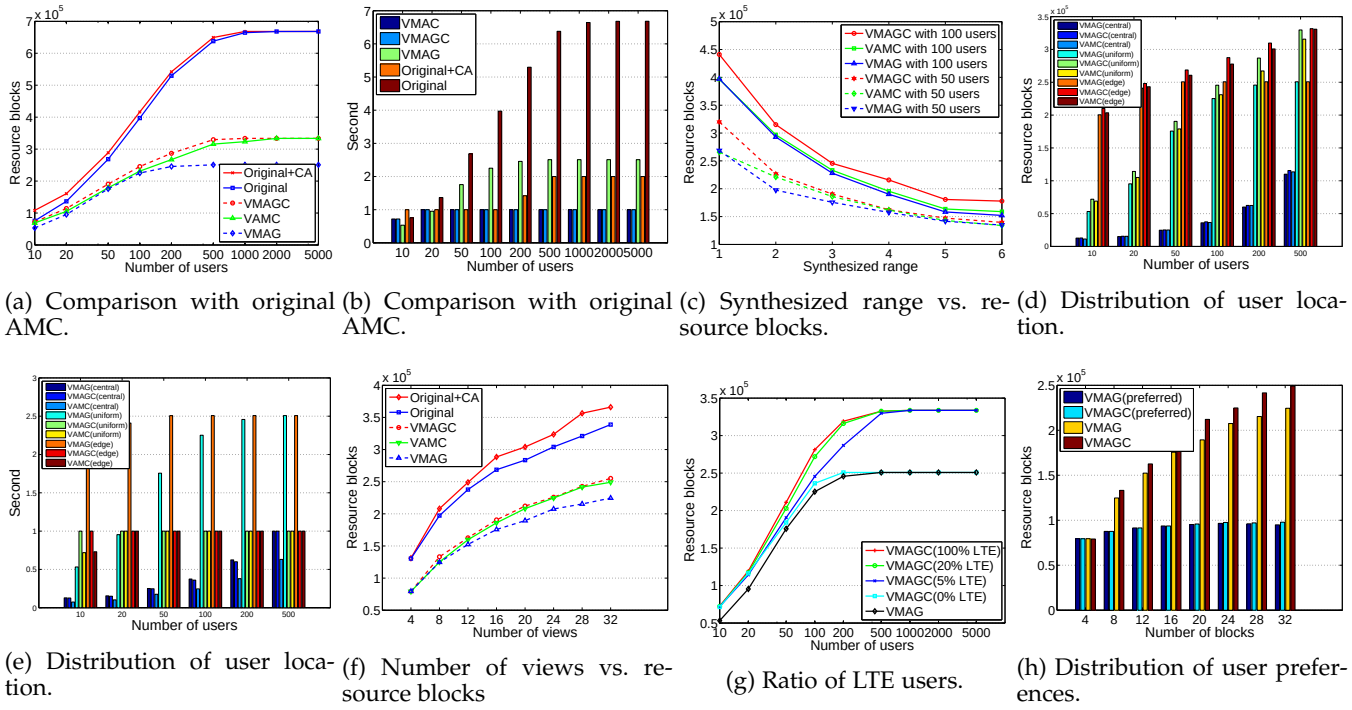
In this section, we evaluate the performance of VMAG and VMAGC in a series of scenarios. The simulation follows [30], [31], [32], [33], [34] to implement an LTE network with C/C++. To the best of our knowledge, there has till now been no related research on bandwidth minimization for multi-view 3D video multicast in LTE-A networks. Thus, we first compare VMAG with the original LTE-A multicast scheme without CA, in which all desired views are multicast to the users, while the lowest MCSs of the users are selected to ensure that all those users can be successfully served. Then, we compare VAMC and VMAGC with the original LTE-A multicast scheme with CA, and VMAGC is also compared with VMAG to investigate the advantages of CA. The delay constraint of VMCS is set as one second, and there are 5% of the LTE users in the multi-carrier system by default.

TABLE 3 summarizes the simulation setting for an LTE-A network with 10 MHz channel bandwidth in each CC, normal CP, and the number of CCs being 5. The type of CA is intra-band contiguous, which means that a wider contiguous bandwidth in the same band is employed for CA. The single carrier system with VMAG and the original LTE-A multicast scheme without CA utilizes a carrier in the lowest band of 5 CCs. Moreover, there is no delay constraint in the single carrier system. The interference, such as distance attenuation, shadow fading, thermal noise follows the existing setting [29]. The MCS and data rate in the above environment follow the literature as set in [28] and [35], respectively, and the confidence interval is 95%. We also adopt the setting of a real multi-view 3D dataset Book_Arrival [2] with 16 views, where the texture video quantization parameter is 30, and depth map quantization parameter is 36, and the PSNR of the synthesized views in DIBR is around $37 - 38$ dB, which is close to the quality of the original LTE-A multicast scheme [36]. The FFmpeg H.264/AVC encoder is employed to encode the view separately. Fig. 7 shows the video data trace of camera 10 in each frame. The default parameters are $V = 16, R = 3, |D| = 50$ selected uniformly at random from the cell, and there are 5% LTE users in D.

5.1 Scenario 1: Comparison with original AMC

Fig. 8a compares the proposed VMAGC and VAMC with the delay constraint, the proposed VMAG without the delay constraint, and the original LTE-A multicast with and without CA (Original+CA and Original). VAMC and VMAG find the optimal solutions with and without the delay constraint in problems VMCS and VMS, respectively. The results manifest that the resource consumption of heuristic algorithm VMAGC is very close to that of VAMC when the delay constraint is incorporated. Note that VMAG incurs the minimum resource consumption since video views are not required to be duplicated in different carriers(for traditional LTE users) to reduce the transmissions delay. The number of required resource blocks in LTE-A increases in all schemes with more users. Nevertheless, about 40% improvement can be achieved by the proposed algorithms due to the efficient aggregation of views with DIBR. Most importantly, the improvement becomes more significant when more users appear in the cell. The difference in the resource consumption between the original LTE-A multicast with CA and the original LTE-A multicast without CA diminishes as the number of users increases because more users with the lowest MCS appear. In this case, the two algorithms thereby tend to generate similar solutions. As the number of users exceeds 200, the resource block consumption saturates in all schemes. For the original LTE-A multicast, almost all views are necessary to be transmitted. By contrast, the resource consumption of VMAGC and VMAG can be effectively reduced thanks to DIBR.

Fig. 8b compares the transmission time of a one-second video in each scheme with different numbers of users. Although exploiting CA may slightly increase the resource consumption, the transmission time of VMAGC always follows the delay constraint. However, VMAG and the original LTE-A multicast without CA tend to violate the delay constraint as the number of users grows since the delay constraint is not considered in the above schemes. The original LTE-A multicast with CA is not able to ensure



(a) Comparison with original AMC.
(b) Comparison with original AMC.
(c) Synthesized range vs. resource blocks.
(d) Distribution of user location.
(e) Distribution of user location.
(f) Number of views vs. resource blocks
(g) Ratio of LTE users.
(h) Distribution of user preferences.

Fig. 8: Comparison of different algorithms for multi-view 3D videos.

the delay constraint when the number of users exceeds 100. Although the resource consumption of the original LTE-A multicast with and without CA is very similar in Fig. 8a, the difference of transmission time in Fig. 8b becomes more significant because the original LTE-A multicast with CA can employ the resources from all of the 5 CCs, but the original LTE-A multicast without CA can only utilize one CC in one second. Therefore, the results reveal that CA can effectively distribute the traffic in multiple carriers and thus limit the delay.

5.2 Scenario 2: Synthesized range

Fig. 8c evaluates VMAG and VMAGC with different R. As expected, the resource block consumption is effectively reduced as R increases. Nevertheless, it is not necessary to set a large R because the improvement becomes marginal as R exceeds 3. The reason is that more LTE users can be served in the same carrier as R increases, and the number of transmitted views also decreases. This result thus indicates that a small R (i.e., limited quality degradation) is sufficient to effectively reduce the bandwidth consumption in LTE-A networks.

5.3 Scenario 3: Distribution of user locations

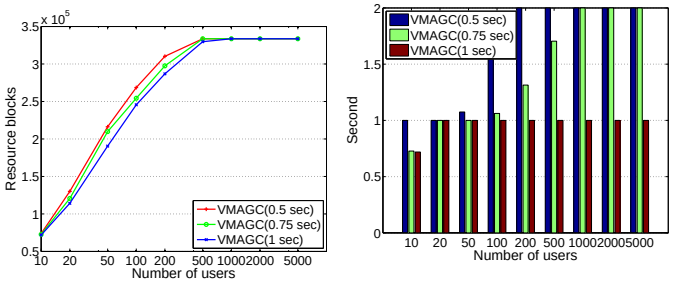
Fig. 8d depicts the impacts of different distributions on the locations for users. Central distribution represents that users have a higher probability (about 95%) to be located in the central part of the cell, while the users in edge distribution have a higher probability (about 95%) to be closer to the boundary of the cell [19]. The locations of the users are assigned uniformly at random in uniform distribution. The result indicates that central distribution incurs much fewer resource blocks in both VMAGC and VMAG due to higher data rates, and both schemes in central distribution follow the delay constraint, as shown in Fig. 8e. Moreover, for central distribution, the performance of VMAGC is almost the same as the performance of VMAG in both Figs. 8d and 8e. On the other hand, as the number of users increases, Fig. 8d reveals that edge distribution and uniform distribution become more similar because both schemes still need to address the requirements of the users closer to the boundary of the cell. Nevertheless, only VMAGC follows the delay constraint, as shown in Fig. 8e, and VMAGC outperforms VMAG in both edge distribution and uniform distribution because VMAGC always follows the delay constraint.

5.4 Scenario 4: Number of views

Fig. 8f graphically explores the impact of different view numbers in a video. The bandwidth consumption in all schemes increases as the video includes more views, because the desired view of each user is chosen uniformly at random, and more views are necessary to be transmitted. The difference on the resource consumption between VMAGC and VMAG also increases since the number of desired views from LTE users grows, and in this case, many more views need to be transmitted. The result manifests that VMAG and VMAGC consistently outperform the original LTE-A multicast scheme for varied numbers of views due to DIBR.

5.5 Scenario 5: Ratio of LTE users

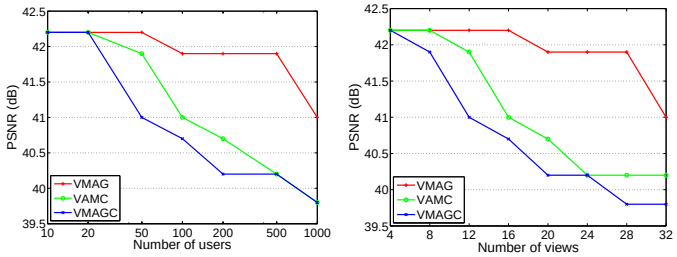
Fig. 8g examines the impacts on different proportions of LTE users in the LTE-A networks. The resource consumption of VMAGC is close to VMAG when there is no LTE user in the LTE-A networks since no extra views needs to be transmitted for LTE users. The resource consumption, on the other hand, increases with increases in LTE users, though the resource consumption only slightly increases when the ratio



(a) Resource blocks

(b) Transmission time for 1 second video data

Fig. 9: Impact of different delay constraints.



(a) Impact of number of users on PSNR.

(b) Impact of number of views on PSNR.

Fig. 10: Video quality under different simulation settings.

of LTE users grows from 20% to 100%, because many views have already been delivered in multiple carriers to LTE users. These findings show that the existence of LTE users hugely impacts on the resource consumption of VMAGC.

5.6 Scenario 6: Distribution of user preferences

Fig. 8h examines the influence on the distributions of the preferred views with the preferred distribution [37], $f(k;s;N) = (1/k^2)/\sum_{n=1}^{N}(1/n^s)$, where k is the preference rank of a view, s is the value of the exponent characterizing the distribution, and N is the number of views. We set $s = 2$ and $N = 16$ in the simulation. In the preferred distribution, the rank of major views is smaller than that for other views, indicating that major views have a higher probability of being requested, and that each user chooses a desired view uniformly at random in the Uniform Distribution. Preferred distribution confirms the idea that many applications with a few major views (i.e., the front sides of objects) are more preferred by users. These results indicate that the transmitted views can be more efficiently aggregated as the user requirements are more skewed in both VMAG and VMAGC.

5.7 Scenario 7: Delay constraint

Fig. 9a shows the impact of different delay constraints. The consumption of resource blocks increases when the delay constraint decreases, because the maximum number of resource blocks that can be allocated to each carrier is reduced. In this situation, more users are inclined to be assigned suboptimal carriers and thereby require more resource blocks. Fig. 9b presents the transmission time for one-second video data in different delay constraints, and a smaller delay constraint is harder to be ensured when the number of users and the number of views become greater. On the other hand, we run the simulation with 10MHz channel bandwidth in each component carrier. It is envisaged that if the simulation is performed with 20MHz channel bandwidth, a smaller delay constraint can be achieved more easily.

5.8 Video quality assessment

We have evaluated the video quality according to the multi-view 3D video techniques [7], [38]. Fig. 10a indicates the impact of number of users on PSNR with 16 views , when the number of resource blocks consumption limit in 1 sec video data 2.5×10^5. VMAG outperforms other two algorithms because it does not consider the delay constraint. As the user number grows, the network becomes more congested, and the video quality of the three proposed algorithm thereby decreases. Nevertheless, the results of VAMC and VMAGC are very close. Also, Fig. 10b shows the impact of number of views on PSNR with 50 users. When a video includes more views, the desired views from users become more diverse, and each view in this case thereby shares a smaller network bandwidth.

6 Conclusion

In light of the emergence of naked-eye mobile devices, this paper investigated bandwidth-efficient multi-view 3D video multicast in LTE-A networks. By exploiting the DIBR, an efficient virtual view rendering algorithm, we first formulated VMS to minimize resource consumption for a multi-view 3D video, and algorithm VMAG was proposed to find the optimal solution of the VMS problem. Moreover, we further extended VMS to support CA and formulated VMCS for multi-view 3D video multicasting with a delay constraint, and VMCS was proven to be NP-hard. We designed algorithm VAMC to find the optimal solutions to small instances of VMCS, and we devised algorithm VMAGC to find near-optimal solutions in polynomial time. The simulation results show that the bandwidth consumption could be effectively reduced by over 30%, and the improvement was more prominent with an increasing number of users in LTE-A networks in both VMAG and VMAGC.

References

[1] D. Lecompte and F. Gabin, "Evolved multimedia broadcast/multicast service (eMBMS) in LTE-advanced: overview and Rel-11 enhancements," *Communications Magazine, IEEE*, vol. 50, no. 11, pp. 68–74, Nov. 2012.

[2] I. Feldmann, M. Mueller, F. Zilly, R. Tanger, K. Mueller, A. Smolic, P. Kauff, and T. Wiegand, "HHI test material for 3-D video," in *Proc. 84th Meet. ISO/IEC JTC1/SC29/WG11, document M15413*, Apr. 2008.

[3] Y. Liu, Q. Huang, S. Ma, D. Zhao, W. Gao, S. Ci, and Hui Tang, "A novel rate control technique for multiview video plus depth based 3d video coding," *Broadcasting, IEEE Transactions on*, vol. 57, no. 2, pp. 562–571, Jun. 2011.

[4] F. Shao, G. Jiang, M. Yu, K. Chen, and Yo-Sung Ho, "Asymmetric Coding of Multi-View Video Plus Depth Based 3-D Video for View Rendering," *Multimedia, IEEE Transactions on*, vol. 14, no. 1, pp. 157–167, Feb. 2012.

[5] Ahmed Hamza and Mohamed Hefeeda, "Energy-efficient Multicasting of Multiview 3D Videos to Mobile Devices," *Multimedia Computing, Communications, and Applications, ACM Transactions on*, vol. 8, no. 3s, pp. 45:1–45:25, Oct. 2012.

[6] Y. Aksoy, O. Sener, A. Alatan, and K. Ugur, "Interactive 2D-3D image conversion for mobile devices," in *Image Processing(ICIP), 2012 19th IEEE International Conference on*, Sep. 2012, pp. 2729–2732.

[7] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, "View Generation with 3D Warping Using Depth Information for FTV," *Signal Processing: Image Communication*, vol. 24, no. 1-2, pp. 65–72, Jan. 2009.

[8] Ya-Ju Yu, Pi-Cheng Hsiu, and Ai-Chun Pang, "Energy-Efficient Video Multicast in 4G Wireless Systems," *Mobile Computing, IEEE Transactions on*, vol. 11, no. 10, pp. 1508–1522, Oct. 2012.

[9] C.G. Gurler, B. Gorkemli, G. Saygili, and A.M. Tekalp, "Flexible Transport of 3-D Video Over Networks," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 694–707, April 2011.

[10] Jongryool Kim, Kiho Choi, Hyunyong Lee, and JongWon Kim, "Multi-View 3D Video Transport using Application Layer Multicast with View Switching Delay Constraints," in *3DTV Conference, 2007*, May 2007, pp. 1–4.

[11] E. Kurutepe and T. Sikora, "Feasibility of Multi-View Video Streaming Over P2P Networks," in *3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2008*, May 2008, pp. 157–160.

[12] Li Zuo, Jian Guang Lou, Hua Cai, and Jiang Li, "Multicast of real-time multi-view video," in *Multimedia and Expo, IEEE International Conference on*, July 2006, pp. 1225–1228.

[13] Zukang Shen, A. Papasakellariou, J. Montojo, D. Gerstenberger, and Fangli Xu, "Overview of 3GPP LTE-Advanced carrier aggregation for 4G wireless communications," *Communications Magazine, IEEE*, vol. 50, no. 2, pp. 122–130, Feb. 2012.

[14] A. Li, K. Takeda, N. Miki, Y. Yan, and Hidetoshi Kayama, "Search Space Design for Cross-Carrier Scheduling in Carrier Aggregation of LTE-Advanced System," in *Communications(ICC), 2011 IEEE International Conference on*, Jun. 2011, pp. 1–5.

[15] Y. Wang, K.I. Pedersen, T.B. Sorensen, and P.E. Mogensen, "Carrier load balancing and packet scheduling for multi-carrier systems," *Wireless Communications, IEEE Transactions on*, vol. 9, no. 5, pp. 1780–1789, May 2010.

[16] H.-W. Lee and S. Chong, "Downlink Resource Allocation in Multi-Carrier Systems: Frequency-Selective vs. Equal Power Allocation," in *a World of Wireless, Mobile and Multimedia Networks(WoWMoM), 2007 IEEE International Symposium on*, Jun. 2007, pp. 1–8.

[17] L. Liu, M. Li, J. Zhou, X. She, L. Chen, Y. Sagae, and M. Iwamura,, "Component Carrier Management for Carrier Aggregation in LTE-Advanced System," in *Vehicular Technology Conference(VTC Spring), 2011 IEEE 73rd*, May 2011, pp. 1–6.

[18] F. Wu, Y. Mao, S. Leng, and Xiaoyan Huang, "A Carrier Aggregation Based Resource Allocation Scheme for Pervasive Wireless Networks," in *Dependable, Autonomic and Secure Computing(DASC), 2011 IEEE Ninth International Conference on*, Dec 2011, pp. 196–201.

[19] L. Chen, W. Chen, X. Zhang, and D. Yang, "Analysis and Simulation for Spectrum Aggregation in LTE-Advanced System," in *Vehicular Technology Conference Fall(VTC Fall), 2009 IEEE 70th*, Sept. 2009, pp. 1–6.

[20] K.I. Pedersen, F. Frederiksen, C. Rosa, H. Nguyen, L.G.U. Garcia, and Yuanye Wang, "Carrier aggregation for LTE-Advanced: functionality and performance aspects," *Communications Magazine, IEEE*, vol. 49, no. 6, pp. 89–95, Jun. 2011.

[21] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[22] A. Vetro, T. Wiegand, and G.J. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 626–642, Apr. 2011.

[23] G. Cheung, V. Velisavljevic, and A. Ortega, "On Dependent Bit Allocation for Multiview Image Coding with Depth-Image-Based Rendering," *Image Procseeing, IEEE Transactions on*, vol. 20, no. 11, pp. 3179–3194, Dec. 2011.

[24] P. Ndjiki-Nya and M. Koppel and D. Doshkov and H. Lakshman and P. Merkle and K. Muller and T. Wiegand, "Depth Image-Based Rendering with Advanced Texture Synthesis for 3D Video," *Multimedia, IEEE Transactions on*, vol. 13, no. 3, pp. 453–465, Jun. 2011.

[25] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRAN); Overall description; Stage 2," 3rd Generation Partnership Project (3GPP), TS 36.300, Sep. 2008. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/36300.htm

[26] N. Miki, M. Iwamura, Y. Kishiyama, U. Anil, and H. Ishii, "CA for Bandwidth Extension in LTE-Advanced," *NTT DOCOMO Technical Journal*, vol. 12, no. 2, Sep. 2010.

[27] S. Martello and P. Toth, *Knapsack Problems: Algorithms and Computer Implementations*. New York, NY, USA: John Wiley & Sons, Inc., 1990.

[28] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures," 3rd Generation Partnership Project (3GPP), TS 36.213, Sep. 2008. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/36213.htm

[29] ——, "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Frequency (RF) system scenarios," 3rd Generation Partnership Project (3GPP), TR 36.942, Sep. 2008. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/36942.htm

[30] Donghyeok Ho and Hwangjun Song, "Networking cost effective video streaming system over heterogeneous wireless networks," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*, Sept. 2013, pp. 3589–3593.

[31] Cheng-Chung Lin, K. Sandrasegaran, and S. Reeves, "Handover algorithm with joint processing in LTE-advanced," in *9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2012*, May 2012, pp. 1–4.

[32] Jing Han and Haiming Wang, "Uplink Performance Evaluation of Wireless Self-Backhauling Relay in LTE-Advanced," in *6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), 2010*, Sept. 2010, pp. 1–4.

[33] A.V. Zelenkov and A. Litvinenko, "OFDM PAPR reduction by pre-scrambling and clipping," in *13th Biennial Baltic Electronics Conference (BEC), 2012*, Oct. 2012, pp. 141–144.

[34] Sinh Chuong Nguyen, K. Sandrasegaran, and F.M.J. Madani, "Modeling and simulation of packet scheduling in the downlink LTE-advanced," in *17th Asia-Pacific Conference on Communications (APCC), 2011*, Oct. 2011, pp. 53–57.

[35] G. Araniti, V. Scordamaglia, M. Condoluci, A. Molinaro, and A. Iera, "Efficient frequency domain packet scheduler for point-to-multipoint transmissions in LTE networks," in *Communications(ICC), 2012 IEEE International Conference on*, Jun. 2012, pp. 4405–4409.

[36] G. Zhu, G. Jiang, M. Yu, F. Li, F. Shao, and Zongju Peng, "Joint video/depth bit allocation for 3D video coding based on distortion of synthesized view," in *Broadband Multimedia Systems and Broadcasting(BMSB), 2012 IEEE International Symposium on,*, Jun. 2012, pp. 1–6.

[37] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications," in *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings.*, vol. 1, Mar. 1999, pp. 126–134.

[38] C. Zitnick, S. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-Quality Video View Interpolation Using a Layered Representation," in *ACM SIGGRAPH*, vol. 23, no. 3, August 2004, p. 600608.

[39] I-Hong Hou and P.R. Kumar, "Scheduling Heterogeneous Real-Time Traffic over Fading Wireless Channels," in *INFOCOM, 2010 Proceedings IEEE*, Mar. 2010, pp. 1–9.

[40] D.O. Reudink, "Properties of mobile radio propagation above 400 MHz," *Vehicular Technology, IEEE Transactions on*, vol. 23, no. 4, pp. 143–159, Nov. 1974.

[41] M.N. Krishnan, E. Haghani, and A. Zakhor, "Packet Length Adaptation in WLANs with Hidden Nodes and Time-Varying Channels," in *Global Telecommunications Conference(GLOBECOM), 2011 IEEE*, Dec. 2011, pp. 1–6.

TABLE 4: Notations in Integer Programming for VMS

Notation	Description
M	The set of Modulation and Coding Schemes (MCS).
V	The number of views in a multi-view 3D video.
R	The DIBR quality constraint.
D	The set of multicast users of a multi-view 3D video.
v_d	The desired view of each user $d \in D$.
$\tau_{m,v}$	Number of resource block to transmit view v in MCS m.
$f_{m,v}$	$f_{m,v} = \{0,1\}$, view v is sent by MCS m or not.
$\epsilon_{d,v}$	$\epsilon_{d,v} = \{0,1\}$, the user d is decided to receive view v or not.
γ_d	The distance between receiving views of user d.
α	The weight of QoE factor.
L_{v_d}	The set of views on the range from $v_d - R + 1$ to $v_d - 1$.
$Q_{(v_d)v}$	The set of views which cannot be chosen when user d selects view v to synthesize.
$A_{d,m}$	$A_{d,m} = \{0,1\}$, user d can decode the packets from MCS m or not, respectively.

APPENDIX

APPENDIX A
INTERGER PROGRAMMING OF VMS PROBLEM

Table 4 summarizes the notation of the formulation.

The objective function (Eq. (2.1)) of VMS is to minimize the number of resource blocks to multicast all selected views in a multi-view 3D video. VMS is to decide binary decision variable $f_{m,v}$, i.e., to select a set of transmitted views in V and assign an MCS for each selected view, where the views to be received by each user d is represented by binary variables $\epsilon_{d,v}$. Eqs. (2.2) and (2.3) state that user d can receive v_d directly ($\epsilon_{d,v_d} = 1$) or synthesize v_d ($\epsilon_{d,v_d} = 0$) from a received left view and right view, whereas the view received must follow the quality constraint. The quality constraint is formulated in Eq. (2.4). If a left view v is selected by user d, every right view that is too far away to v (i.e., every view in $Q_{(v_d)v}$) cannot be selected, and Eq. (2.3) ensures that a closer right view thereby will be selected. In this work, we set an identical quality constraint for each user. Nevertheless, Eq. (2.4) can be extended to provide each user with a different quality constraint. Eq. (2.5) and (2.6) records the distance between receiving views of user d. Eq. (2.7) enforces that each view v can be assigned at most one MCS m. For each view v required by d, Eq. (2.8) ensures that user d can always receive the view with a decodable MCS.

$$\min \sum_{v=1}^{V} \sum_{m=1}^{M} f_{m,v} \times \tau_{m,v} + \alpha \sum_{d \in D} \gamma_d \tag{2.1}$$

$$\sum_{v \in L_{v_d}} \epsilon_{d,v} = 1 - \epsilon_{d,v_d} \quad , d \in D \tag{2.2}$$

$$\sum_{v=v_d+1}^{V} \epsilon_{d,v} = 1 - \epsilon_{d,v_d} \quad , d \in D \tag{2.3}$$

$$\epsilon_{d,v} + \sum_{j \in Q_{(v_d)v}} \epsilon_{d,j} \leq 1 \quad , d \in D,\ v \in L_{v_d} \tag{2.4}$$

$$(\epsilon_{d,i} + \epsilon_{d,j} - 1) \times (j - i) \leq \gamma_d \quad , j > i,\ d \in D \tag{2.5}$$

$$\gamma_d \geq 0 \quad , d \in D \tag{2.6}$$

$$\sum_{m=1}^{M} f_{m,v} \leq 1 \quad , v \in V \tag{2.7}$$

$$\sum_{m=1}^{M} A_{d,m} \times f_{m,v} \leq \epsilon_{d,v} \quad , v \in V, d \in D \tag{2.8}$$

$$f_{m,v}, \epsilon_{d,v} \in \{0,1\} \tag{2.9}$$

APPENDIX B
ALGORITHM VMAG

Algorithm 1 Algorithm VMAG

Require: candidate ranges in MCS 1 to M
Ensure: minimum number of resource block to serve all demands

1: **Initial Phase**;
2: **for** each MCS from $m = 1$ to $m = M + 1$ **do**
3: **while** MCS m has a candidate range c which has not been examined **do**
4: choose candidate range c;
5: let h = first view in c; t = last view in c; $j = h$;
6: **while** $j \neq t$ **do**
7: **if** MCS $m - 1$ has a candidate range c' ending at j **then**
8: **if** c' overlaps another candidate range in MCS $m - 1$ **then**
9: **Vertical and Horizontal aggregation**;
10: **else**
11: **Vertical aggregation**;
12: **end if**
13: **else**
14: **No aggregation**;
15: **end if**
16: $j = j + 1$;
17: **end while**
18: **end while**
19: **end for**

Line 1 is *Initialization Phase*. It initializes and identifies the *service ranges* (i.e., a continuous range of views to be served) for each MCS. From line 2 to 19, the **VMAG** algorithm examines each candidate range from the smallest MCS (e.g., BPSK) to the largest MCS. From line 3 to 18, the second phase aggregates multiple service ranges in adjacent MCSs and assigns the transmitted views accordingly. It also explores the possibility to aggregate adjacent ranges located in the same MCS.

APPENDIX C
INTERGER PROGRAMMING OF VCMS PROBLEM

Table 5 summarizes the notations of the formulation. VMCS is to select a set of transmitted views and assign the corresponding MCSs and carriers. Eq. (3.1) presents the objective function of VMCS to minimize the number of resource blocks for all selected views sent in different MCSs in an LTE-A network with carrier aggregation. Eqs. (3.2)-(3.6), and (3.8) are similar to Eqs (2.2)-(2.6) and (2.8), respectively. Eq. (3.7) states that each view v can be assigned only one MCS m and one carrier c. Eq. (3.9) states the delay constraint. It ensures that the total number of allocated resource blocks for each carrier c cannot exceed the delay bound Δ_c. Eq. (3.10) addresses the capability of each LTE user, which does not support CA and can be assigned to only one carrier. It ensures that if one left view is selected in carrier c, the corresponding right view also needs to be allocated to the same carrier.

TABLE 5: Notations in Integer Programming for VMCS

Notation	Description
M	The set of Modulation and Coding Schemes (MCS).
V	The number of views in a multi-view 3D video.
R	The DIBR quality constraint.
D	The set of multicast users of a multi-view 3D video.
$\mathcal{C}$	The set of Component Carriers (CC).
C	The number of Component Carriers.
v_d	The desired view of each user $d \in D$.
$\tau_{m,v}$	The number of resource block to transmit view v in MCS m.
$f_{m,v,c}$	$f_{m,v,c} = \{0,1\}$, view v is sent by MCS m in carrier c or not.
$\epsilon_{d,v,c}$	$\epsilon_{d,v,c} = \{0,1\}$, the user d is decided to receive view v in carrier c or not.
γ_d	The distance between receiving views of user d.
α	The weight of QoE factor.
L_{v_d}	The set of views on the range from $v_d - R + 1$ to $v_d - 1$.
$Q_{(v_d)v}$	The set of views which cannot be chosen when user d selects view v to synthesize.
$A_{d,m,c}$	$A_{d,m,c} = \{0,1\}$, user d can decode the packets from MCS m in carrier c or not.
x_d	$x_d = \{0,1\}$, user d is an LTE-A user or not.

$$\min \sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{v=1}^{V} f_{m,v,c} \times \tau_{m,v} + \alpha \sum_{d\in D} \gamma_d \tag{3.1}$$

$$\sum_{c=1}^{C}\sum_{v\in L_{v_d}} \epsilon_{d,v,c} = 1 - \sum_{c=1}^{C} \epsilon_{d,v_d,c} \quad , d \in D \tag{3.2}$$

$$\sum_{c=1}^{C}\sum_{v=v_d+1}^{V} \epsilon_{d,v,c} = 1 - \sum_{c=1}^{C} \epsilon_{d,v_d,c} \quad , d \in D \tag{3.3}$$

$$\sum_{c=1}^{C} \epsilon_{d,v,c} + \sum_{c=1}^{C}\sum_{j\in Q_{(v_d)v}} \epsilon_{d,j,c} \le 1 \quad , d \in D,\ v \in L_{v_d} \tag{3.4}$$

$$(\sum_{c=1}^{C} \epsilon_{d,i,c} + \sum_{c=1}^{C} \epsilon_{d,j,c} - 1) \times (j - i) \le \gamma_d \quad , j > i,\ d \in D \tag{3.5}$$

$$\gamma_d \ge 0 \quad , d \in D \tag{3.6}$$

$$\sum_{c=1}^{C}\sum_{m=1}^{M} f_{m,v,c} \le 1 \quad , v \in V \tag{3.7}$$

$$\sum_{c=1}^{C}\sum_{m=1}^{M} A_{d,m,c} \times f_{m,v,c} \le \epsilon_{d,v,c} \quad , v \in V, d \in D \tag{3.8}$$

$$\sum_{v=1}^{V}\sum_{m=1}^{M} f_{m,v,c} \times \tau_{m,v} \le \Delta_c \quad , d \in D \tag{3.9}$$

$$(1 - x_d) \times (\sum_{v\in L_{v_d}} \epsilon_{d,v,c} - \sum_{v=v_d+1}^{V} \epsilon_{d,v,c}) = 0 \quad , c \in C, d \in D \tag{3.10}$$

$$f_{m,v,c}, \epsilon_{d,v,c} \in \{0,1\} \tag{3.11}$$

APPENDIX D
VMCS IS NP-HARD

Theorem 2. *VMCS is NP-hard.*

Proof. We prove that VMCS is NP-Hard with the reduction from the 0/1 knapsack problem [27]. Specifically, given a set of I items, each item i with a weight w_i and a value a_i, the knapsack problem aims to maximize the total value by selecting a subset of items I' of I with the total weight not exceeding M. For each instance of the knapsack problem, we build an instance of VMCS with $R = 1$ and $C = 2$. In other words, DIBR is not supported, and the MCS of each view is determined from the user with worst channel condition requesting the view. That is, the number of resource blocks for transmitting view v in carrier c is fixed, $\tau'_{c,v} = \tau_{\bar{M}(c,v),v}$, where $\bar{M}(c,v)$ is the MCS to serve the user with the worst channel condition requesting view v in carrier c. In the VMCS instance, $V = I$, $\tau'_{c_1,i} = w_i$, $\tau'_{c_2,i} = w_i + a_i$, $\Delta_{c_1} = M$, and $\Delta_{c_2} = \sum_{i\in I}(w_i + a_i)$. Therefore, since the channel condition is better in c_1, the VMCS instance aims to select a subset V' of V for c_1 to minimize $\sum_{i\in V/V'} \tau'_{c_2,i} + \sum_{j\in V'} \tau'_{c_1,j}$, such that $\sum_{v\in V'} \tau'_{c_1,v} \le \Delta_{c_1}$, $\sum_{v\in V/V'} \tau'_{c_2,v} \le \Delta_{c_2}$.

In the following, we prove that the optimal solution to a knapsack instance and the optimal solution to VMCS are one-to-one correspondent. Given the optimal solution I' in the knapsack instance, we let $V' = I'$ accordingly. The total value of I' is $\sum_{j\in I'} a_j$, and the total resource consumption of the VMCS instance is $\sum_{i\in I}(w_i + a_i) - \sum_{j\in I'} a_j = \sum_{i\in I/I'}(w_i + a_i) + \sum_{j\in I'} w_j = \sum_{i\in I/I'} \tau'_{c_2,i} + \tau_{j\in I'}\tau'_{c_1,j}$. More specifically, the resource consumption in carrier c_1 is $\sum_{j\in I'} \tau'_{c_1,j} = \sum_{j\in I'} w_j$, which is smaller than M. The resource consumption in carrier c_2 is the summation of resource consumption for each view in carrier c_2. Since the total value of solution I', i.e., $\sum_{j\in I'} a_j$, is maximized, the resource consumption in VMCS, i.e., $\sum_{i\in I}(w_i + a_i) - \sum_{j\in I'} a_j$, is also minimized (i.e., $\sum_{i\in I}(w_i + a_i)$ is a constant). Conversely, given the optimal solution V' in the VMCS instance, we let $I' = V'$ accordingly. Since the resource consumption in VMCS, i.e., $\sum_{i\in V}(w_i + a_i) - \sum_{j\in V'} a_j$, is minimized, the total value $\sum_{j\in V'} a_j$ is maximized. □

APPENDIX E
ALGORITHM VAMC

In the following, we present algorithm VAMC to find the optimal solution to the small instances of VMCS. To find the optimal solution, a possible approach is to extend the VMAG in Section III. However, a candidate range from view h to t in MCS 1, which is a simple scenario for *No Aggregation*, needs to record all combinations of resource consumption for multiple carriers. More specifically, view h needs to be recorded for each carrier, i.e., $\mu^1_{h,h}$ in carrier 1, $\mu^1_{h,h}$ in carrier $2, \cdots$, $\mu^1_{h,h}$ in carrier C. Also, the solution $\mu^1_{h,h+1}$ in carrier 1 needs to employ the solutions from $\mu^1_{h,h}$ in carrier 1, $\mu^1_{h,h}$ in carrier $2, \cdots$, as well as $\mu^1_{h,h}$ in carrier C. Therefore, the number of solutions required to be recorded for the candidate range from view h to t in MCS 1 is C^{t-h}. Moreover, the order from MCS 1 to M examined in VMAG is not applicable in VMCS because each user has different channel conditions in different carriers.

Therefore, in the following, we design a new algorithm, named VAMC, for finding the optimal solution. VAMC includes two phases: *Initialization* and *View and Carrier Assignment*. The first phase identifies the *candidate assigned MCSs* for each view in different carriers, and the second phase aggregates the serving range of each view through connecting candidate assigned MCSs in adjacent views, which are in the synthesized range of DIBR, for different carriers.

Initialization Phase

This phase first identifies the candidate assigned MCS for view v in carrier c, which is represented as $\alpha_{v,c}$, according to the following two observations. Firstly, for requested

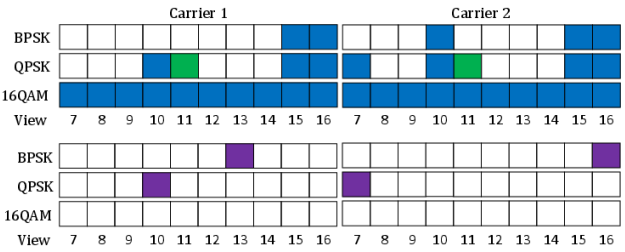


Fig. 11: An example for VAMC

view v, users located from MCS $m+1$ to M can be served when view v is transmitted in MCSs from 1 to m. Therefore, VAMC determines the candidate assigned MCSs for preferred view according to the user with worst channel condition in view v. That is, view v transmitted with its candidate assigned MCS is guaranteed to serve all users preferred view v. Secondly, for preferred view v in carrier c, $\alpha_{v,c}$ can be different from the candidate assigned MCSs for view v in other carriers because users tend to have different channel conditions in different carriers. Therefore, the user with the worst channel condition for view v varies in different carriers.

View and Carrier Assignment Phase

This phase aggregates multiple views by DIBR and determines the transmitted views with the corresponding MCSs and carriers. Specifically, algorithm VAMC examines each requested view sequentially. That is, view v will be examined only after all candidate assigned MCSs for view j with $j < v$, i.e., $\{\alpha_{j,1}, \alpha_{j,2}, ..., \alpha_{j,C}\}$, are investigated. The goal is to derive $\Pi(\delta_1, \delta_2, ..., \delta_C)$, which is defined as the set of candidate solutions with the corresponding *resource consumption set*, $\{\delta_1, \delta_2, ..., \delta_C\}$, where δ_c represents the resource block consumption in carrier c. In other words, $\{\delta_1, \delta_2, ..., \delta_C\}$ represents current resource block consumption in all carriers. Note that each resource consumption set is associated with several candidate solutions, since different candidate solutions may require the same resource consumption in each carrier.

Equipped with DIBR, when view v is examined for the candidate solutions in $\Pi(\delta_1, \delta_2, ..., \delta_C)$, every candidate solution with transmitted view j following $j \geq v - R$ will have an opportunity to be aggregated with view v. Moreover, the number of candidate assigned MCSs for view j in carrier c is at most M, and the number of candidate assigned MCSs for view j is at most $M \times C$. The number of candidate solutions in $\Pi(\delta_1, \delta_2, ..., \delta_C)$ is constrained to be $R \times M \times C$, and $\pi(f, \delta_1, \delta_2, ..., \delta_C)$ denotes the f-th candidate solution in $\Pi(\delta_1, \delta_2, ..., \delta_C)$.

Each candidate solution, such as $\pi(f, \delta_1, \delta_2, ..., \delta_C)$ in $\Pi(\delta_1, \delta_2, ..., \delta_C)$, records the *last transmitted view* v (i.e., the transmitted view with the largest index) in $\pi(f, \delta_1, \delta_2, ..., \delta_C)$, together with its corresponding carrier c and MCS m. MCS m is chosen from $\alpha_{v,c}$ from *Initialization Phase*. Let (v, m, c) denote the *last transmitted triple* for $\pi(f, \delta_1, \delta_2, ..., \delta_C)$, which also means that the service range of $\pi(f, \delta_1, \delta_2, ..., \delta_C)$ is from view 1 to v. In the following, $\Pi(\delta_1, \delta_2, ..., \delta_C)$ is the result of adding view v with the candidate assigned MCS m in carrier c to the f-th candidate solution in $\Pi(\delta'_1, \delta'_2, ..., \delta'_C)$, i.e., $\pi(f, \delta'_1, \delta'_2, ..., \delta'_C)$, where $m \in \alpha_{v,c}$.

Example. Following the example in Fig. 11, where where the green blocks represent the requests from LTE users. Here we assume that both carriers 1 and 2 have a delay constraint of 8 resource blocks. The resource consumption set is $\{0, 0\}$ at the beginning. First, we examine view 7. Since view 7 can be satisfied by all three MCSs in carrier 1 and by BPSK and QPSK in carrier 2, we have five candidate solutions with resource consumption set $\{4, 0\}$, $\{3, 0\}$, $\{2, 0\}$, $\{0, 4\}$, and $\{0, 3\}$, and the last transmitted triples are $(v, m, c) = (7, BPSK, 1), (7, QPSK, 1), (7, 16QAM, 1), (7, BPSK, 2)$, and $(7, QPSK, 2)$, respectively.

Afterward, view v can be added to $\pi(f, \delta'_1, \delta'_2, ..., \delta'_C)$ when the following conditions hold.

1) Delay Constraint.
   First, δ_c does not exceed the delay constraint in carrier c, i.e., $\delta_c \leq \Delta_c$. Next, because the difference between $\Pi(\delta_1, \delta_2, ..., \delta_C)$ and $\pi(f, \delta'_1, \delta'_2, ..., \delta'_C)$ is view v with candidate assigned MCS $m \in \alpha_{v,c}$, $\delta_c = \delta'_c + \tau_{m,v} \leq \Delta_c$.
2) Quality Constraint.
   After adding view v to the candidate solution, view v' in the last transmitted triple (v', m', c') in $\pi(f, \delta'_1, \delta'_2, ..., \delta'_C)$ can synthesize the views between v' and v, where $v' \geq v - R$. Therefore, other candidate solutions in $\Pi(\delta'_1, \delta'_2, ..., \delta'_C)$ can be discarded when the last transmitted view v' is smaller than $v - R$, and the number of candidate solutions in $\Pi(\delta'_1, \delta'_2, ..., \delta'_C)$ is at most $R \times M \times C$
3) Receive Constraint.
   The last transmitted triple (v', m', c') in $\pi(f, \delta'_1, \delta'_2, ..., \delta'_C)$ shows that the requests of users from view 1 to v' can all be satisfied. Therefore, when view v is examined, only the requests of users from view $v'+1$ to v need to be investigated. That is, the candidates (v, m, c) and (v', m', c') must cover all requests of users from view $v'+1$ to v. All requests of users from view $v'+1$ to v are examined to ensure the channel condition of these users can meet both MCS m' in carrier c' and MCS m in carrier c. Moreover, if there are any requests from LTE users between v' and v, then carrier c must be the same as c'. Since a LTE user can only receive the data from one carrier, any two transmitted views must belong to the same carrier if any request from an LTE user exists in the range of these two transmitted views.
4) Duplication Constraint.
   The last transmitted triple for each candidate solution in $\Pi(\delta_1, \delta_2, ..., \delta_C)$ is unique. That is, any two transmitted triples from candidate solutions in $\Pi(\delta_1, \delta_2, ..., \delta_C)$ cannot be same. View v with candidate assigned MCS m in carrier c can be added to $\pi(f, \delta'_1, \delta'_2, ..., \delta'_C)$ as a new candidate solution in $\Pi(\delta_1, \delta_2, ..., \delta_C)$, when (v, m, c) is not the same as any other last transmitted triples of the candidate solutions in $\Pi(\delta'_1, \delta'_2, ..., \delta'_C)$.

Example. Following the example in Fig. 11, when view 8 are examined after view 7, new candidate solutions are generated, such as $\Pi\{2, 4\}$ with original $\Pi\{2, 0\}$ and new $(v, m, c) = (8, BPSK, 2)$. Note that some new candidate solutions may not be generated if it has the same last transmitted triple as another candidate solution but it requires more resource to transmit the views in each carriers. For

example, $\Pi\{4,4\}$ with original $\Pi\{4,0\}$ and new $(v,m,c) = (8, BPSK, 2)$ will not be generated. The candidate solutions $\Pi\{2,4\}$ and $\Pi\{4,4\}$ mentioned above have the same last transmitted triple $(8, BPSK, 2)$. However, the candidate solution $\Pi\{2,4\}$ dominates $\Pi\{4,4\}$, i.e., it requires no more resource in each carrier. Therefore, not every new candidate solutions will be generated.

VAMC examines view 11 after the view 8, 9 and 10 are examined, and the candidate solutions with last transmitted view $v \leq 7$ will be discarded due to *Quality Constraint*. When the view 12 and 13 are examined, VAMC also carefully selects the right candidate solutions to satisfy the requested view 11 from LTE user. For example, the last transmitted triple$(10, QPSK, 1)$ cannot combine $(12, QPSK, 2)$, since the request view 11 from an LTE user cannot be synthesized from views in different carriers. The procedure ends when view 16 is examined. The solution returned by VMAG is $\Pi\{7,7\}$ with transmitted triples $(7, QPSK, 2), (10, QPSK, 1), (13, BPSK, 1)$ and $(16, BPSK, 2)$, as the purple blocks in fig. 11. By transmitting these views with their candidate assigned MCS in the corresponding carriers, all requests can be satisfied. The resource consumption is $7 + 7 = 14$ in this example.

View and Carrier Assignment Phase finishes when all candidate assigned MCSs for view V, $\{\alpha_{V,1}, \alpha_{V,2}, ..., \alpha_{V,C}\}$ are examined for each candidate solution in different resource consumption sets. The optimal solution of VMCS can be found by fully scanning each candidate solution in different resource consumption sets. Therefore, the candidate solutions with minimal resource consumption, $\sum_{i \in \mathcal{C}} \delta_i$, and view V in its last transmitted triple is returned as the solution of the VMCS problem for serving all requests of users from view 1 to V.

Solution Optimality

In the following, we prove by induction that one of candidate solutions with minimal resource consumption, $\sum_{i \in \mathcal{C}} \delta_i$, and view V in its last transmitted triple, is the optimal solution of the VMCS problem. Since *Initialization Phase* investigates all possible assigned MCSs for each preferred view in each carrier, the optimal solution must be combined by some preferred views with corresponding candidate assigned MCSs and carriers. *View and Carrier Assignment Phase* further explores the possible solutions in each resource consumption set that can be generated, beginning with resource consumption set $\{0, ..., 0\}$. The first preferred view 1 generates several candidate solutions of serving only view 1 with each candidate assigned MCS in different carriers. Other preferred views are examined sequentially afterward. That is, view v is considered when views from view 1 to $v-1$ are investigated. Therefore, the insertion of view v can extend the range of candidate solutions from view 1 to v, but not every view in this range will be transmitted thanks to DIBR. The final optimal solution can be acquired when view V is processed. Therefore, the candidate solution with minimal resource consumption and view V as the last transmitted view is regarded as the optimal solution.

Complexity Analysis

In the worst case, there are at most V preferred views, and each preferred view v has at most M candidate assigned MCSs in carrier c from the *Initialization Phase*. In the *View and Carrier Assignment Phase*, a maximum of $R \times M \times C$ candidate solutions are examined in each resource consumption set for view v due to *Quality Constraint*. For each preferred view, the maximum number of scanning iteration is $(\Delta_c)^C$ iterations. Moreover, each iteration involves at most $M \times C$ selections. Therefore, the complexity of VAMC is $O((R \times M \times C) \times (V \times M \times C) \times (\Delta_c)^C) = O(M^2C^2RVV(\Delta_c)^C)$. Note that C is usually a small constant (such as 5 [13]) in real LTE-A networks.

APPENDIX F
ALGORITHM VMAGC

Algorithm 2 Algorithm VMAGC

Require: candidate ranges in MCS 1 to M and in carriers 1 to C
Ensure: minimal number of resource block to serve all demands
1: carrier list $\Phi = \{1, ..., C\}$;
2: **while** $|\Phi| \neq 1$ **do**
3: **for** each carrier in Φ **do**
4: run algorithm **VMAG**;
5: **end for**
6: sort carrier list $\Phi = \{1, ..., c\}$ in ascending order of the resource consumption;
7: **if** any view v can only be served in carrier c **then**
8: $\Theta(c) = \Theta(c) + v$;
9: **end if**
10: **if** any view v consumes the fewest resource in carrier c **then**
11: $\Theta(c) = \Theta(c) + v$;
12: **end if**
13: $k = 1$;
14: **while** $k \neq c - 1$ **do**
15: maximize *resource change* between carrier k and c;
16: update demand for other carriers;
17: $k = k + 1$;
18: **end while**
19: $\Theta(c) = \Theta(c)+$ views not served by carriers $\{1, 2, ..., c-1\}$
20: **if** resource consumption in carrier c exceeds Δ_c **then**
21: **if** views can be exchanged **then**
22: **Exchanging**;
23: **if** resource consumption in carrier c still exceeds Δ_c **then**
24: **Moving**;
25: **end if**
26: **else**
27: **Moving**;
28: **end if**
29: **end if**
30: **if** resource consumption in carrier c still exceeds Δ_c **then**
31: **return Solution infeasible!**
32: **end if**
33: update demands for carriers $\{1, 2, ..., c-1\}$;
34: $\Phi = \Phi - \{c\}$;
35: **end while**

Line 3 to 12 in **VMAGC** are the pseudo code of *Search Phase*, which estimates the resource consumption of each carrier according to VMAG. Next, line 13 to 19 represent *Maximization Phase* to assign the views for the carrier with the most resource change. Line 20 to 29 are the *Examination Phase*, which examines whether the resource consumption in

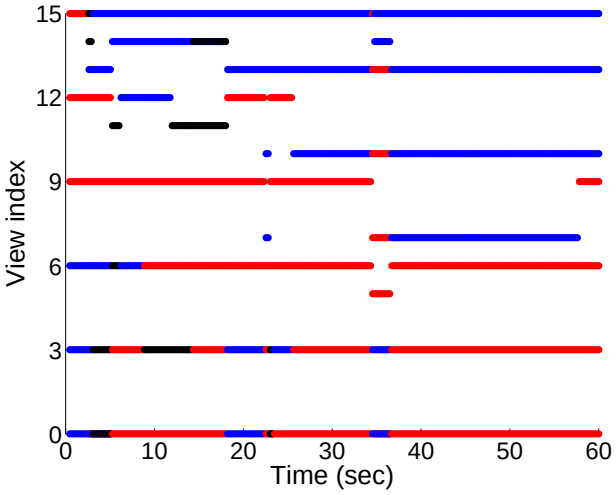


Fig. 12: The transitions of transmitted views in 60 seconds.

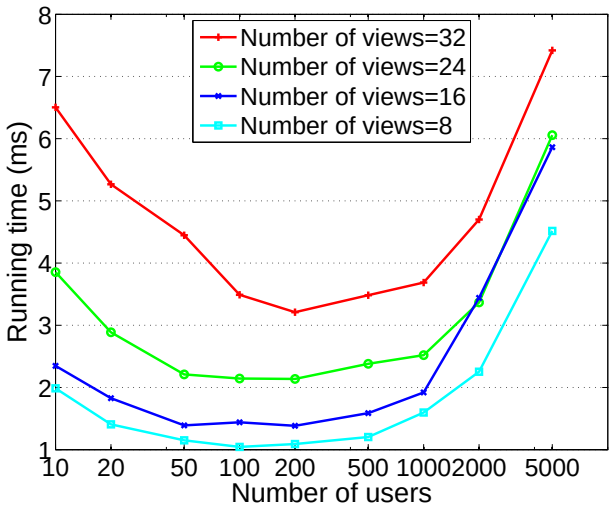


Fig. 13: The VMAGC computation time.

the carrier with the most resource consumption (i.e., carrier c) exceeds the delay constraint.

APPENDIX G
GENERAL CASE OF VMCS

In Section IV, we first explored the fundamental case with each selected view transmitted in only one CC, and in this part we now extend it to the general case in which a view can be transmitted in multiple CCs. In both algorithms VAMC and VMAGC, the final solution must necessarily be investigated when a view can be transmitted in multiple CCs. v^1 is defined as the first transmitted view in the final solution to the VMCS problem, and we have $\{v^1, v^2, ..., v^L\}$ as the set of transmitted views in the final solution to the VMCS problem, where L is the number of transmitted views. Moreover, $\{m^1, m^2, ..., m^L\}$ is the set of corresponding MCS for each transmitted view, and $\{c^1, c^2, ..., c^L\}$ is the set of the corresponding carriers for each transmitted view. First, for each transmitted view v^k, if v^k can cover the demands of users from v^k to v^{k+2} and consume fewer resource blocks than v^{k+1} when v^k is transmitted repeatedly in different carriers except for carrier c^k, then v^{k+1} can be updated as view v^k. In this case, v^{k+1} is transmitted originally to satisfy the demands of the LTE users between v^k and v^{k+2} because some LTE users between v^k and v^{k+2} cannot receive v^k transmitted in c^k, and we try to transmit v^k repeatedly in other carriers both to satisfy LTE users between v^k and v^{k+2} and to consume fewer resource blocks than v^{k+1}. Then, we also showed that if v^k can cover the demands of users from v^{k-2} to v^k and consume fewer resource blocks than v^{k-1} consumes, when v^k is transmitted repeatedly in different carriers except for carrier c^k, then v^{k-1} can be updated as view v^k. Each view in $\{v^1, v^2, ..., v^L\}$ is examined such that if some views can be transmitted repeatedly for less resource consumption than $\{v^1, v^2, ..., v^L\}$, $\{\bar{v}^1, \bar{v}^2, ..., \bar{v}^L\}$ is defined as the final set of transmitted views in the general case of VMCS.

APPENDIX H
SIMULATION IN TIME-VARYING CHANNEL CONDITION

The channel condition [39], [40], [41] for each user changes at random time between each group of pictures(GOP), where 1 GOP$=$ 240ms typically. Fig.12 presents the simulation result under a time-varying channel with 50 users. The red, blue, and black dots represent the views transmitted in carrier 1, 2, and 3, respectively. The result manifests that the view assignment is quite steady and robust even when the channel condition of a user frequently varies. The reason is that this paper targets on multicast for multi-view 3D videos, and thus a view is usually subscribed by multiple clients. When the channel condition of a client improves, the channel condition of another client may degrade. Moreover, with DIBR and carrier aggregation in LTE-A, our algorithm can assign a nearby views in different carrier to a user when the channel condition of the original carrier changes, so that the current view assignment remains valid for all users.

APPENDIX I
ALGORITHM VMAGC RUNNING TIME

The analysis is in Section 4.2 is the worst-case complexity. We have measured the average computation time for VMAGC in an HP server with 4 Intel(R) Xeon(R) CPU E7-4870 @ 2.40GHz and 384G RAM. Fig.13 presents the running time with different numbers of users and views. When the number of users are small, there are many feasible view and MCS combinations in each carrier, which means the VMAGC spends more time to acquire the minimal number of resource block consumption. As the number of users exceed 500, the algorithm takes more time to *Initialize* the candidate ranges for each user, resulting in a higher computation time. The result manifests that the average computation time is smaller than 10ms, indicating that it is feasible to deploy the algorithm in a real network.