International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 7 Issue 7 Ver III || July 2018 || PP 24-39

Joint resource allocation and edge computing forreal-time wireless video transmission*

QuanxinZhao
Department of Technology Center, Sichuan Netop Telecom Co.,Mianyang Sichuan 621000, China

Abstract: *Video with increasing resolution has been one offundamental network applications, such popular and emerging application attracts a variety of attentions from both industry and academia. Due to the constrained computational capability, limited power supply and dynamic transmission bandwidth, there exists a gap for general users in widely existed wireless networks to obtain the same user experience as those in wirednetworks. This paper analyzes video decoding for general low-end mobile devices in wireless networks and divides this problem into: a novel video decoding architecture design, and resource allocation based on the novel architecture. First, a novel architecture of real-time video decoding for computation offloading is introduced for mobile devices. Second, based on the proposed architecture, a joint computation offloading and multicast resource allocation optimization problem is introduced to maximize user satisfaction ratio and minimize energy consumption. Third, a feasibility condition of the optimization problem is derived in terms of the computational task offloading for real-time videos. Forth, a low-complexity sub-optimal scheme with proved computational complexity is designed by dividing the original NP-hard optimization problem into sub-problems to accomplish group-user allocation, group-subchannel allocation, and offloading ratio calculation.*

Keywords -*Computation offloading,energy saving,mobile edge decoding,resource allocation,user satisfaction ratio*

Date of Submission: 27-06-2018 Date of acceptance: 12-07-2018

I. Introduction

Video is a fundamental network application, which important application has dominated the network traffic [1]. Video resolution is significantly increasing with the consistent evolution of video technique, such as current ultra-high-definition (UHD) video [2-5] enabling much higher resolution than that of the conventional video and providing great user experience improvement. Such popular and emerging video is attracting a variety of attentions from both industry and academia [6-12]. Currently, network service providers are deploying plenty of UHD videos for fast obtaining profit from consumers.

Mobile devices have already become the necessity for users,bringing a promising and large market for network serviceproviders to deploy novel UHD videos. However, the deploymentspeed is not as fast as people desire because wireless users cannot obtain the same watch experienceas those in wired networks. This restriction primary liesin 1) mobile device with low computational capability fordecoding UHDvideo with unacceptable decodingtime, 2) battery with limited power supply for decodingUHD video traffic with high energy consumption,and 3) wireless networks with dynamic bandwidth for videotransmission under varying condition channels.The above restriction leaves wireless user watch experience guaranteeand energy saving under UHD videos a tough challenge and how tosolve this problem is an open issue.

Existing works on schemes for playing high-resolution videoson low-end mobile devices can be classified into: 1) coding/decoding protocols, such as scalable video coding (SVC)[35], 2) decoding architecture, e.g., mobile edgecomputing (MEC), remote graphical processing unit (GPU),cloud and fog *This paper was invited to IJESI 2018computing [17][18]. This paper focuses on real-time UHD video delivery scenario, which scenariobelongs to the second category. For the second category,previous works can be further classified into two kinds: 1)decoding on data source then transmitting to data receiver[19][21][23], 2) devices with video content deploy decodingtasks on remote devices [20][25][27]. There exists a restrictionif directly applying existing architecture, i.e., above twocategories, in our focused scenario. This reason of restrictionis list below: 1) the data volume after computing by remote device under thesecond categorybecomesextremely high, making the second category cannot be appliedin our focused scenario, 2) for decoding video at data source(e.g., cloud) under the first category, its traffic transmissionrequires much high bandwidth for the core networks. Asa result, traditional

architecture, i.e., above two categories,cannot be applied in our focused scenario, requiring a novelarchitecture for video decoding to support the UHDvideo in wireless networks for low-end mobile devices.

MEC [15][16] is a promising and emerging technique tooffload computational tasks from massive low-end mobileterminals to high-performance computing servers located inwireless access networks. To decode real-time UHDvideos under the MEC architecture, users can transmitall the video content to nearby servers for decoding,thereafter, users can receive and play the decoded video. Underthe MEC architecture, there is a significant difference between generalcomputational intensive tasks and graphically intensive ones.Compared with general computational intensive tasks, graphically intensive tasks, e.g., videodecoding, is much different, whichdifference introduces a restriction if directly applying existingMEC architecture into UHD video decoding.This restriction is that, 1) if a user sends all the video contentto a nearby computing server for decoding, both the uplink(from user to server) and downlink (from server to user) willbe occupied. 2) After decoding by the remote server, thedata volume transmitted in the downlink becomes extremelyhigh, compared with general computational intensive taskswith small traffic volume after computing. It is infeasible forthe twice transmissions with high bandwidth requirements forvideo decoding in current wireless networks. This motivates anew architecture for decoding UHD videos inwireless networks.

In the current field of video decoding architecturedesign, the primary goal is to save energy. For example,mobile devices utilize high-end remote devices for complexcomputations [20] or high-performance GPU servers [22][24]to decode video [18][26] and realize visualizing 3D videostreaming sessions [23] for energy saving. Although existingconsolidate works have solved the problem of video playingat low-end mobile devices, their works cannot address thefollowing situations: 1) energy saving from users' perspectivecannot fully reflect the system performance for evaluating thecase of multiple video resolutions,which motives a new user watch experience metric. 2) Fewworks focus on resource allocation under multicast condition,where there are many users with the same content request.

To address the challenges of video decoding and multicasttransmitting for low-end mobile devices in wireless networks, it finds our works into: 1) video decoding architecture design,2) resource allocation based on the designed architecture. Specificallyspeaking, first, a novel architecture is designed for real-timevideo decoding. Second, a joint computation offloadingand multicast resource allocation is introduced to maximizeuser satisfaction ratio and minimize energy consumption. Themain contribution of this paper to tackle this tough challengeis below:

1) A novel architecture of real-time video decoding forcomputation offloading is introduced for low-end mobile deviceswith limited battery supply, constrained computationcapability, and dynamic traffic transmission bandwidth.

2) Our optimization problem is formulated with the goalof energy saving and user watch experience improvement, whichproblem jointly considers computational task offloading andmulticast resource allocation.

3) A feasibility condition of the optimization problem isproposed in terms of the computational task offloading forreal-time videos under the proposed architecture.

4) A low-complexity sub-optimal scheme with proved computationalcomplexity is introduced by dividing the originalNP-hard optimization problem into sub-problems to accomplishgroup-user allocation, group-subchannel allocation, andoffloading ratio determination.

5) Simulation results show that a) taking clock frequencyof devices and channel gain into consideration on the processof user allocation, our scheme can achieve better performancethan that in existing works, b) joint computational complexitywith resource allocation will achieve higher performance thanthat in existing works.

The remainder of this paper is organized below. SectionII describes the related work. Section III shows the proposedmobile edge decoding architecture. Section IV defines the edgedecoding ratio. Section V presents the system model. SectionVI gives problem formulation. After introducing schemesin Section VII with the corresponding simulation results inSection VIII. Finally, section IX concludes the paper.

II. Related Work

This section first introduces the motivation of computationaltask offloading from the aspect of traffic type. Then, we givea NOVEL architecture from the aspect of traffic transmissionpath. Next, offloading schemes with the corresponding resource allocation are introduced.

A. Motivation

Low-end mobile deviceswith limited computation resource and power supplycan offload complex computation tasks to remote high-performancenodes, e.g., high-end mobile device [19][20],server [21-23], and cloud [25], for increasing computationcapability, decreasing execution time, saving energy, meetingrequirements for emerging applications, and improving userexperience. Existing works are shown as below.

Some works focus on computational intensive applications,such as voice recognition [25]. For example, in Ref. [20],low-end mobile devices utilized the high-performance GPUof a remote device in an ad hoc network to perform thecomplex computations for the benefit of energy consumptionand execution time. In Ref. [22], users with low-performanceGPU shared a high-performance GPU server to improve computationcapability and reduce execution time. In Ref. [25],mobile devices with low-power supply offloaded their highcomputing CPU load onto GPUs in the cloud, for emergingtime-insensitive high-computational applications.

Others focus on graphically intensive applications, suchas video coding, rendering and decoding, graphics intensiveapplications, CAD/CAM computing, remote desktop sharing.In Ref. [23], a comprehensive client-server 3D renderingframework enabled limited resource devices with collaborativevisualization to interact with graphics intensive OpenGL-basedapplications. In Ref. [19], they proposed a mobile-to-mobileremote computing protocol for smartphones to realizeremote desktop sharing by providing a remote view for real-time collaboration. In Ref. [21], efficient remote work withgraphically intensive applications (e.g., CAD/CAM and GPUcomputing) utilized GPU virtualization in remote computingto realize productive remote access to the office workplacescomposed with word processors, spreadsheets. However, thereis lack of metric to evaluate user watch experience undermultiple different resolution videos.

B. Traffic Type in MEC

MEC technique [17][18] brings a variety of benefits, suchas increasing computation capability, decreasing task executiontime, saving energy, meeting requirements for emergingapplications, and improving user experience. With MEC technique,massive low-end mobile devices with limited computationalresources and low-power supplycan offload their complex computation tasks tonearby high-performance nodes, e.g., high-end mobile devices[19][20], local servers [21-23], high-performance GPUservers [22][24], and even cloudlets [25].

Computational intensive applications are key targets thatthe MEC technique focuses on. These applications includevoice recognition [25], augmented reality, visualizing 3D videostreaming sessions [23], video decoding [18][26], etc. Existingworks, most related to this paper, on these applications areas follows. For example, low-end mobile devices in [20]utilized the high-performance GPU of a remote device in anad hoc network to perform the complex computations for thebenefits of energy consumption and execution time. In Ref. [22],users with low-performance GPUs shared a high-performanceGPU server to improve computation capability and reduceexecution time. Mobile devices with low-power supply [25]offloaded their CPU load onto remote GPUs, for emergingtime-insensitive high-computational applications.

Graphically intensive applications are a kind of computationalintensive applications, which difference is that theformer application is visible by eyes. Existing works, mostrelated to this paper, on these applications are below. In Ref. [23], acomprehensive client-server 3D rendering framework enabledlimited resource devices with collaborative visualization tointeract with graphics intensive OpenGL-based applications.The authors in Ref. [19] proposed a mobile-to-mobile remotecomputing protocol for smartphones to realize remote desktopsharing by providing a remote view for real-time collaboration.Efficient remote work with graphically intensive applications[21] utilized GPU virtualization in remote computing to realizeproductive remote access to the office workplaces composedwith word processors and spreadsheets.

Unfortunately, in graphically intensive applications consideredin this paper, most of exiting works on high-definitionvideo decoding for low-end terminals primarily focus on theoptimization of energy saving. Since energy saving cannotfully reflect users' satisfaction on visible watch experience,some metrics are required to evaluate user experience, especiallyunder the video with multiple different resolutions in thispaper. On the other hand, few works have considered multicastcondition, where there are many users in the same group withthe same video content requests. Above limitations motive usto design a joint experience improving and energy saving asa new optimization objective under multicast condition.

C. Traffic Journey in MEC

We classify existing traffic in MEC into two categories fromthe aspect of data transmission path, named, return-journeyand single-journey.

On one hand, some existing works belong to the return-journeycategory, which means the computation task is firsttransmitted from a data source node to a remote node fortask computing. The computed result is then delivered to thedata source [20][25][27], which process is shown as in Fig.1. There are some existing works in this category. For instance, theauthors in [20] realized a distributed computing technique forlow-end mobile devices to use remote high-end device's GPUfor offloading the task of computational intensive applicationsto save energy consumption. In Ref. [27], nodes with installedGPUs acted as acceleration servers for serving otherusers withGPU virtualization technique. Insufficient computing power ofmobile and wearable devices [25] offloaded computation tasksto GPUs nearby.

The return-journey scheme has its limitation in wirelessnetworks for decoding UHD videos under existingMEC architecture, because high bandwidth is requiredfor transmitting the undecodedUHD video andextremely high bandwidth is required for transmitting the decodedcontent, which is different from conventional computingtasks whose decoded data volume is much smaller than thatof the undecoded data.

On the other hand, some works belong to the single-journeycategory, which shows the task is computed on the data sourceand then transmitted to the data receiver [19][21][23]. Therelated works are list below. In Ref. [23], the server handledvisualization sessions for 3D video streaming to computevideo streams and transmitted for clients with different screenresolutions and bandwidth. Authors in [19] applied remotedesktop sharing for users with a remote desktop view. In [21],the authors studied GPU virtualization for remote computingfor the applications of virtual and remote workplace forremote work with graphically intensive applications, such asCAD/CAM and GPU computing.

As the computing tasks for graphically intensive applicationshave been decoded on the source data side and thentransmitted to the data receiver, single-journey category alsohas limitation under our considered scenario because decoding on the data source, i.e., a cloud videoserver, for the real-time UHD video will takehuge bandwidth requirements for core networks. In summary,the traditional MEC architecture cannot be directly appliedin the considered scenario of this paper, requiring a novelarchitecture.

D. Schemes

We divide existing offloading schemes in terms of the traffictransportation path into two categories: return-journey andsingle-journey as defined in Sec. II-C.

Some existing works belong to the return-journey category,where return-journey means the task is first transmitted fromthe data source to a remote node, on which the calculationtask is applied with the result transmittedback to the data source [20][25][27]. For instance, in Ref. [20], the authors realized adistributed computing technique for low-end mobile devicesto use remote high-end device’s GPU for offloading thetask of computational intensive applications to save energyconsumption. In Ref. [27], nodes with installed GPUs actedas acceleration servers for serving other users with GPUvirtualization technique. In Ref. [25], insufficient computingpower of mobile and wearable devices offloaded computationsto GPUs on the cloud. Return-journey scheme has limitationin wireless networks under UHD videos becausethe volume of data rate after decoding becomes much higherthan that of conventional computing tasks, which tasks consumegreat bandwidth for transmitting high bandwidth UHDvideo and extremely high bandwidth decoding videoin wireless networks.

Other works belong to the single-journey category, whereone-way shows the task is computed on the data source andthen transmitted to the data receiver [19][21][23]. In Ref.[23], the server handled visualization sessions for 3D videostreaming to compute video streams and transmit it for clientswith different screen resolutions and bandwidth. In Ref. [19],they applied remote desktop sharing for users with a remotedesktop view. In Ref. [21], the authors studied GPU virtualizationfor remote computing for the applications of virtual andremote workplace for remote work with graphically intensiveapplications, such as CAD/CAM and GPU computing, officeworkplaces composed with word processors, spreadsheets. Thecomputing tasks for graphically intensive applications havebeen decoded on the source data side and then transmitted tothe data receiver. Single-journey also has limitation becausedecoding cloud located video content, source data, will takehuge bandwidth requirements for core networks.

E. Resource Allocation

Multicast wireless transmission is a feasible technique fortransmitting the same content to a group of users. Mostexisting works utilize the channel gain as the metric to allocateusers to multicast groups to achieve better bitrate.

There are widely research works. For example, in Ref.[36], they studied a multicast group division scheme based onlink quality differences among multicast users in orthogonalfrequency-division multiple access (OFDMA)-based wirelessnetworks. In Ref. [37], they studied a high spectral efficiencymulticast transmission strategy and proposed a multicast subgroupformation scheme, where group users are divided intoseveral subgroups according to their channel state information(CSI). In Ref. [35], they proposed a subgrouping technique byexploiting multiuser diversity and frequency selectivity for thedelivery of real-time scalable multicast video flows, such asInternet Protocol television (IPTV) over Long-Term Evolution(LTE) networks. In Ref. [32], they designed a multicastsubgrouping strategy by aggregating subsets of users withsimilar channel quality levels for multilayer video services tooptimize user satisfaction ratio, throughput and fairness.

Most existing works utilize the channel gain as the metric toallocate users to multicast groups to achieve better bitrate forrequesting higher resolution video and better user watch experience,which metric cannot be directly used in our considered scenario,because real-time UHD video transmission for low-enddevices. This is because users with limited power supplynot only focus on channel gain for achieving high data ratefor receiving UHD video but require energy savingunder limited energy supply for consistently long video playingtime. This makes energy constraint, i.e., clock frequency,of users a key factor to be considered to combine withchannel gain under user allocation process. Moreover, there exitsthe variation of the volume of data traffic, i.e., much highervolume after decoding compared to small volume data ofexisting works, which will cost more bandwidth and give themulticast resource allocation a challenge. Besides, by combining multicast resource allocation with computationoffloading, our work is opposite to existing joint optimization problemfocusing on unicast transmission with computation offloading,especially under the traffic volume explosion condition.

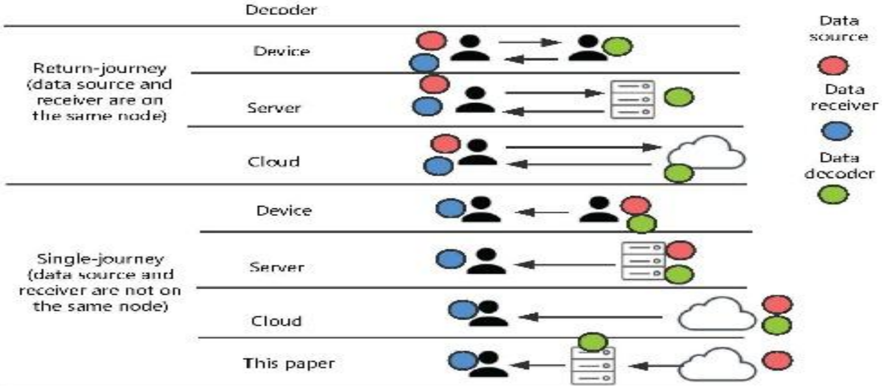
Decoder
Return-journey (data source and receiver are on the same node)
Device
Server
Cloud
Single-journey (data source and receiver are not on the same node)
Device
Server
Cloud
This paper
Data source
Data receiver
Data decoder

Fig. 1. architecture comparison

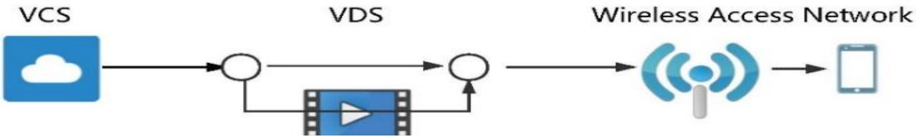

Fig. 2. system model

III. Mobile Edge Decoding Architecture

We propose a video decoding architecture, named mobileedge decoding (MED), for a general scenario where mobiledevices with limited power supply and constrained computationcapability can play real-time UHD videowith high user watch experience. MED architecture is shown as the’single-journey’ in Fig. 1, which consists of a data source,a decoding device, and a data receiver. Traffic transmissionunder the proposed MED architecture in Fig. 1 operates asbelow:

1) The video content located at the data source is transmittedfrom the cloud video server.

2) When the video content is delivered to the wireless accessnetwork, the decoding device in the computing server can helpdecode.

3) After remote decoding by the server, the traffic is thentransmitted to the user, i.e., the data receiver.

As video content under the MED architecture is decodedon its transmission path from the source to the receiver, thedifference between the MED architecture (’Single-journey’ inFig. 1) and existing MEC architecture (’Return-journey’ inFig. 1) is list below.

1) MED architecture has the ’single-journey’ traffic transmissionfrom the computing sever to the user in wirelessnetworks, which is different from existing traffic transmissionunder the MEC architecture with twice transmission.

2) After decoding by the computing server, the data volumebecomes much higher than those transmitted from the cloudvideo server. This is distinct from general computational intensivetraffic under the MEC architecture where the computeddata often has a small volume.

3) Remote task and local task are separated at remotedecoder and uncompleted remote task cannot be done by localcomputing if it cannot be completed within some thresholdconditions. Since local computing for uncompleted remotecomputing will lead to a large delay, computing timeand energy consumption, enough bandwidth should be usedto guarantee the transmission time for decoded video data.

4) Computing resource deployed at wireless networks islimited for serving only mobile users within certain distanceand deployed at small base station rooms, compared to conventionalcloud computing with nearly unlimited computingresources.

An example of the MED architecture is given as shownin Fig. 2, which is to be used as the analyzed system inthis paper. The example system consists of a Video ContentServer (VCS) located in the cloud network as the data source,a Video Decoding Server (VDS) located in the wireless accessnetwork as a decoding device, a base station, and some mobiledevices acting as data receivers. One video with 2 resolutions,labeled a and b, is transmitted independently from the VCS inthe cloud network. In the wireless access network, the videocontent can be decoded by both the VDS and the user. Eachresolution video can individually decide its data volume to bedecoded by the VDS.

The MED architecture is proposed for a general scenario,where mobile devices can play real-time UHDvideo in wireless networks with the design goal of maximizingsatisfaction ratio and minimizing energy consumption. In nextsection, based on the MED architecture, we will introduce acore optimization variable, named edge decoding ratio (EDR)to obtain the design goal.

IV. Edge Decoding Ratio

There is one video with G resolutions in the VCS as shownin Fig. 2, represented by G independent resolution groups.Each group can decide the volume of video decoded by theVDS, denoted as edge decodingratio (EDR)γ_g, where $0 \leq \gamma_g \leq 1$. The EDR indicates thatthere are $\gamma_g$percentage of decoding tasks sent to the VDS and$1 - \gamma_g$percentage sent to the receiver. In general, $\gamma_g$can beused to represent, how much volume of video traffic (bps, Mb) or how much time of video traffic decoded by remote devices.

The real-time infinite video sequence in VCS is dividedinto multiple segments for mathematical analysis based on thewidely used division scheme in [28], which is reasonable becausewe will analyze the system at the time scale of resourceallocation (ms) rather than playing (min). The sequence for the video with$G$ resolutions is represented as:

$$T = \{T_1, T_2, \cdots T_g, \cdots, T_G\} \qquad (1)$$

And the sequence for each resolution group is representedas:

$$T_g = \{\tau_{g,1}, \tau_{g,2}, \cdots, \tau_{g,1}, \cdots\}, \forall g \qquad (2)$$

Each segment, task$\tau_{g,i}$, can choose its own EDR$\gamma_{g,i}$in the range from 0 to 1 for decoding video on different devicesincluding the VDS and the receiver, i.e., $\gamma_{g,i}$ percentage forremote decoding and $1 - \gamma_{g,i}$percentage local decoding,compared with existing binary task division assumptions in[28][31], in which a task can be computed either on the VDSor the receiver.

To obtain the feasible condition of the mobile edge decoding(MED) system, demand bound function $dbf(\tau_{g,i}, v)$is appliedto represent the maximum energy consumption for task $\tau_{g,i}$ that must be satisfied within consumed energy v, whichfeasible condition is extended from existing time-oriented one[28] to energy-oriented one. The consumed energy interval ofthe MED system is assumed as $(V, V + v]$, where the energyconsumption of $\tau_{g,i}$ is smaller than v under current energylevel V. Within this interval, mobile edge decoding task mustbe finished, which introduces the performance metric, namedfeasible condition, of the MED system. The feasible conditionof the MED system is obtained from two theorems below.

Theorem 1: For task $\tau_{g,i}$with $\gamma_{g,i}$percentage of remotedecoding on the VDS and $1 - \gamma_{g,i}$percentage of local decodingon the receiver, the demand bound function $dbf(\tau_{g,i}, x)$ isupper bounded by:

$$dbf(\tau_{g,i}, x) \leq \frac{V(\gamma_{g,i})}{e_k} * x \qquad (3)$$

where system determined value $V(\gamma_{g,i})$ is the upper boundenergy consumption with respect to $\gamma_{g,i}$. e_k is energy consumptionfor decoding video on receiver k.

Proof. This comes from the definition of the demand boundfunction in [28] but extending from existing time-oriented oneto energy-oriented one.

After obtaining the demand bound function for the sequence$\tau_{g,i}$ in each resolution group in **Equation (3)**, then, wecalculate the feasible condition for the infinite real-time videosequence T in **Equation (4)**.

Theorem 2: For task $\tau_{g,i}$ with a given$\gamma_{g,i}$the feasiblecondition for all the infinite real-time video sequence can beguaranteed with energy consumption constraint by:

$$\sum_{T_g \in T} \sum_{\tau_{g,i} \in T_g} \left(\int_0^1 \frac{V(\gamma_{g,i})}{e_k} d(\gamma_{g,i})\right) \leq 1 \qquad (4)$$

Proof. see appendix A.

V. System Model

We denote resolution group set, mobile device set andsubchannel set as $\mathcal{g} = \{g, g = 1,2, \cdots, G\}, \varkappa = \{k, k = 1,2, \cdots, K\}, \aleph = \{n, n = 1,2, \cdots, N\}$,respectively.Groups, users and subchannels are distinct. All the N subchannels follow the i.i.d Rayleigh fading. Let $\alpha_{g,k} \in \{0,1\}$represent the index of group-user allocation, where $\alpha_{g,k} = 1$represents that user k is in resolution group g. Otherwise,$\alpha_{g,k} = 0$. Further, $\beta_{g,n} \in \{0,1\}$is applied as

the index of group-subchannel allocation, where $\beta_{g,n} = 1$shows allocating subchannel $n$to resolution group g.Otherwise, $\beta_{g,n} = 0$. In this section, we focus on one segment task and substitute $\gamma_{g,i}$by $\gamma_g$to simplify analysis.

Each user can select only one group to play one resolution video:

$$\sum_g \alpha_{g,k} \leq 1, \forall k \tag{5}$$

Each subchannel can be allocated no more than one group since groups are distinct:

$$\sum_g \beta_{g,n} \leq 1, \forall n \tag{6}$$

Offloading ratio for each group satisfying:

$$0 \leq \gamma_g \leq 1, \forall g \tag{7}$$

The data rate for the video with resolution $g$which is not decoded by VDS can be expressed as:

$$C_g = (1-\gamma_g)R_g, \forall g \tag{8}$$

where$R_g$is the data rate for the video with resolution $g$transmitted from the VCS.

The data rate of the decoded video with resolution $g$by VDS is:

$$M_g = J(\gamma_g, R_g), \forall g \tag{9}$$

where the function J(·,·) represents the output data rate after decoding by VDS.

Decoding capacity of the VDS in terms of the data rate is limited by:

$$\sum_g \gamma_g R_g \leq M_{max} \tag{10}$$

where M_{max} represents the upper bound capacity of the VDS.

The throughput for group$g$ is:

$$O_g = \sum_n \beta_{g,n} B_n log_2\left(1 + \frac{P_n H_{g,n}^2}{\sigma^2}\right), \forall g \tag{11}$$

where the channel gain satisfies $H_{g,n} = min_{k\in\{\alpha_{g,k}=1\}} H_{k,n}$. P_0 is radiation power. σ^2 is noise power. B_0 is subchannel bandwidth, where subchannels are assumed with the same channel quality among all the group for simplicy.

User satisfaction ratio is used to represent users choosing different resolution videos [29]:

$$s_k = \sum_g \alpha_{g,k} \frac{1-e^{-\theta\left(\frac{S_g}{S_{max}}\right)^{0.74}}}{1-e^{-\theta}}, \forall k \tag{12}$$

where $S_g$is the satisfaction value for choosing the video with resolution $g$and S_{max} is the highest value of satisfaction. Further, θ is a system parameter [29].

Energy consumption for decoding video and receiving traffic through network card [30] for each mobile device is calculated as:

$$e_k = \sum_g \alpha_{g,k} \left(\frac{I(F_k)C_g}{F_k} + \frac{P_{NIC}\,(M_g + C_g)}{O_g}\right), \forall k \tag{13}$$

where $F_k$is the clock frequency of devices.$I(F_k)$is decoding power function [30] in terms of F_k.$P_{NIC}$is receiving power of the network interface card. C_g/F_k and $(M_g + C_g)/\ O_g$ are time duration for video decoding and traffic reception delay [31], respectively.

VI. Problem Formulation

This section introduces an optimization problem of theMED system jointly considering both computational task offloadingand multicast resource allocation with the aim of maximizingsatisfaction ratio and minimizing energy consumptionfor users to play real-time UHD videos. Theoptimization problem is designed to find the optimal edge decoding ratio γ_g, group-user allocation index $\alpha_{g,k}$and group-subchannel allocation index $\beta_{g,n}$.

This optimization problem is shown below:

$$\boldsymbol{P0}: max_{\{\alpha_{g,k},\beta_{g,n},\gamma_g\}} = \sum_k s_k - q\sum_k e_k \tag{14}$$

subject to:

$$C1: \sum_g \alpha_{g,k} \leq 1, \forall k$$

$$C2: \sum_g \beta_{g,n} \leq 1, \forall n$$

$$C3: M_g + C_g \leq \ O_g, \forall g$$

$$C4: \sum_g \gamma_g R_g \leq M_{max}$$

$$C5: \alpha_{g,k} \in \{0,1\}, \forall \mathrm{g}, k$$

$$C6: \beta_{g,n} \in \{0,1\}, \forall \mathrm{g}, \mathrm{n}$$

$$C7: 0 \leq \gamma_g \leq 1, \forall g$$

where $q$is a system determined parameter.**Constraint C3** represents the allocated data rate for group $g$is greater than or equal to system requirement. **Constraint C4** shows the decoding capacity of the VDS.

The optimization problem **P0** is a non-convex non-linear programming problem [34], which is much difficult to obtain a global optimal solution. To solve this optimization problem, we will introduce a low-computational complexity solution in next section.

VII. Proposed Algorithms

This section proposes a low computation complexity sub-optimal solution for the joint optimization problem by dividing **P0** into two subproblems **P1** and **P2**.

The first subproblem **P1** focuses on offloading ratio $\gamma_g$calculation under the case where the indicators of group-user allocation $\alpha_{g,k}$and group-subchannel allocation $\beta_{g,n}$are given. Meanwhile, the second subproblem **P2** is to calculate $\alpha_{g,k}$and $\beta_{g,n}$when $\gamma_g$is obtained after solving by **P1**. These two subproblems are shown below.

$$\boldsymbol{P1}: min_{\{\gamma_g\}} q \sum_g \sum_k \alpha_{g,k} (\frac{I(F_k)C_g}{F_k} + \frac{P_{NIC}\,(M_g+C_g)}{O_g}) \qquad (15)$$

subject to: C3,C4.

$$\boldsymbol{P2}: max_{\{\alpha_{g,k},\beta_{g,n}\}} \sum_k \sum_g \alpha_{g,k} \frac{1-e^{-\theta\left(\frac{S_g}{S_{max}}\right)^{0.74}}}{1-e^{-\theta}} - q \sum_k \sum_g \alpha_{g,k} (\frac{I(F_k)C_g}{F_k} + \frac{P_{NIC}\,(M_g+C_g)}{O_g}) \qquad (16)$$

subject to: C1,C2,C3,C5,C6.

A. Subproblem P1

This subsection is to solve $\gamma_g$in **P1**. For each resolution group, we first calculate the optimal $\overline{\gamma_g}$of the objective function in **P1** by **Theorem 3**. Next, we modify the above obtained optimal$\overline{\gamma_g}$into$\check{\gamma}_g$by **Theorem 4** with the consideration of the **Constraint C3** in **P1**. Further taking the **Constraint C4** in **P1** into consideration,an iteration process is applied to decrease $\check{\gamma}_g$ with system determined step size $\Delta\gamma_g$ until **Constraint C4** is satisfied. Finally, we can obtain the results $\check{\gamma}_g$, which is optimal under the conditions given in **Theorem 5**. The detail process is shown in **Algorithm 1**.

Theorem 3: The optimal result $\gamma_g$of the objective function in **P1** can be obtained by:

$$\widehat{\gamma_g} = argmin_{\{\gamma_g\}} \left(q \sum_k \alpha_{g,k} \left(\frac{I(F_k)C_g}{F_k} + \frac{P_{NIC}\left(M_g+C_g\right)}{O_g}\right)\right), \forall \mathrm{g} \qquad (17)$$

under the case where $M_g$is either linear function or quadratic function.

Proof.see appendix B.

Theorem 4: The optimal result of **P1** can be obtained below when taking the **Constraint C3** in **P1** into consideration:

$$\check{\gamma}_g = \begin{cases} move\ \overline{\gamma_g}\ close\ to\ FR_g \cap [0,1], \overline{\gamma_g} \notin FR_g \cap [0,1] \\ \qquad\qquad \dot{\gamma}_g, \overline{\gamma_g} \in FR_g \cap [0,1] \end{cases} \qquad (18)$$

where $FR_g$is obtained as a feasible range against variable $\gamma_g$from **Constraint C3** in **P1**.

Proof: see appendix C.

Theorem 5: The optimal result of **P1** can be obtained no matter when $R_g$is equal or not, ifM_g is a linear function.

Proof.see appendix D.

Algorithm 1 Scheme for **P1**

```
Require:
    Index set of resolution group: $\mathscr{g}$,
    Index of group-user allocation:$\alpha_{g,k}$
    Index of group-subchannelallocation:$\beta_{g,n}$
    System determined step size: $\Delta\gamma_g$
    Temporary variable: $\delta_g$
Ensure:
for each resolution group$g$do
    Calculate$\overline{\gamma_g}$by Equation (15)
    Modify$\overline{\gamma_g}$ into $\check{\gamma}_g$ by Equation (18)
end for
while Constraint C4in P1 is not satisfied do
for each resolution group $g$do
    Calculate energy increase $\delta_g$ of the objective function ofP1 against the variable increase $\Delta\gamma_g$
  end for
```

```
Find resolution group $g^*$ with the minimal energy increase by $g^* = argmin_g \delta_g$
Renew $\breve{\gamma}_g$ by $\breve{\gamma}_g = \breve{\gamma}_g + \Delta\gamma_g$
  **if**$\breve{\gamma}_g$is beyond the range of $FR_g$**then**
    Delete $g$ from$\mathcal{G}$ by $\mathcal{G} = \mathcal{G} \backslash g$
**end if**
**end while**
Output: $\breve{\gamma}_g$
```

Algorithm 2 Scheme for **P2**

Require:

Index of group-user allocation:$\alpha_{g,k}$

Index of group-subchannelallocation:$\beta_{g,n}$

Objective function value of **P2** calculated from group-subchannel allocation scheme: $Obj1$

Objective function value of **P2** calculated from group-user allocation scheme: $Obj2$

System determined threshold value: Γ

Ensure:

```
**while**$|Obj1 - Obj2| > \Gamma$**do**
        Calculate$Obj1$by **Algorithm 3**
        Calculate$Obj2$by **Algorithm 4**
**end while**
```

B. Subproblem P2

After solving **P1** and obtaining γ_g, this subsection is to calculate $\alpha_{g,k}$and $\beta_{g,n}$. First, users are allocated to resolution groups randomly to get the initial value of $\alpha_{g,k}$. After that, we allocate subchannel for groups with the aim of maximizing the objective function value of **P2**. This iterate process including group-user allocation and group-subchannel allocation will not stop until the objective function value converges.

Group-subchannel allocation: When the initial user allocation is given, we iteratively search for each subchannel among all the groups to find one group achieving the maximum objectivefunction value of **P2** and then allocate the subchannel to the group. The group-subchannel allocation will not stop until all the subchannels are allocatedand the **Constraint C3** in **P2** has been satisfied. The detail process is list in **Algorithm 3**.

Group-user allocation: Based on the results obtained from the above group-subchannel allocation, one resolution group may have multiple subchannels. Thereafter, we allocate users to groups. This algorithm has two steps, where the first is to allocate each group subchannel one user and the second step is to allocate the remaining users for groups, i.e., allocating all the remaining users to all group channels, with the aim of maximizing the objective function value of **P2**. The two steps operate below in detail. 1) The algorithm operates for each group individually. In each group, round robin is applied for group subchannels, and one feasible user in set H with the criteria shown in **Theorem 6** is selected for each group subchannel, which user selection process will not stop until the **Constraint C3** is satisfied. When **Constraint C3** of **P2** is satisfied, we allocate all the selected users in this group to all the group subchannels. 2) After the first step, for each unallocated user, the algorithm searches for each group to find one with the maximum objective function value if the user can satisfy the channel gain requirement. The above detail process is list in **Algorithm 4**.

Theorem 6: A user $k$that can be allocated to subchannel $n$in resolution group$g$ should satisfy:

$$H_{k,n} \leq H_{k,n'}, n, n' \in \{\beta_{g,n} = 1\}, n' \neq n \qquad (19)$$

where $H_{k,n}$is the channel gain of user $k$on subchanneln.

Proof. See appendix E.

Algorithm 3 Group-subchannel Allocation

Require:

Index set of resolution group: $\mathcal{G}$

Index set of subchannel: ℵ

Index of group-user allocation:$\alpha_{g,k}$

Index of group-subchannelallocation:$\beta_{g,n}$

Temporary set: $Y_{g,n}$

Ensure:

```
**for** each subchannel $n$**do**
**for** each resolution group $g$**do**
```

```
        Calculate the objective function value $Y_n$ of **P2**
    **end for**
    Find the maximum objective function value by $(g^*, n) \leftarrow argmax_{g,n} Y_{g,n}$
    Allocate $n$ to $g^*$ by $\beta_{g^*,n} = 1$
    Delete $n$ from $\aleph$ by $\aleph = \aleph \backslash n$
    Calculate $O_{g^*}$by **Equation (11)** with $g^*, n, \beta_{g^*,n} = 1$
    **if**$M_{g^*} + C_{g^*} \leq O_{g^*}$**then**
        Delete $g^*$by$\mathscr{g} = \mathscr{g} \backslash g^*$
    **end if**
**endfor**
Output: $\beta_{g^*,n}$
```

Algorithm 4 Group-user Allocation

Require:

```
    Index set of resolution group: $\mathscr{g}$
    Index set of user: $\varkappa$
    Index of group-user allocation:$\alpha_{g,k}$
    Index of group-subchannelallocation:$\beta_{g,n}$
    Index set of user channel gain $\mathcal{H}$: by sorting the value of channel gain $H_{n,k}$ in increasing order
    Pointer for subchannel $n$ on matrix $\mathcal{H}$:$Z_n = 0$
    Temporary set: $Y_{g,k}$
```

Ensure:

```
**for** each resolution group $g$**do**
    **whileConstraint C3** in **P2** is not satisfied **do**
        Apply round robin searching for each subchannel $n \in \{\beta_{g,n} = 1\}$allocated to resolution group $g$
        **while** $Z_n \neq k$**do**
            **if**$Z_n$-th user in $\mathcal{H}$can satisfy the criteria in **Theorem 6 then**
                Allocate $Z_n$-th user to subchannel $n$
                **break**
            **else**
                $Z_n = Z_n + 1$
            **end if**
        **end while**
    **end while**
    Allocate above selected users to group $g$ and obtain $\alpha_{g,k}$
    Allocate the selected users to all the group subchannels
    Delete the selected users from user set $\varkappa$
**end for**
**for** each unallocated user$k$ **do**
    **for** each resolution group $g$**do**
        Calculate objective function value $Y_{g,k}$in **P2** with $g$ and $k$
    **end for**
    Find the maximum objective function value by $(g^*, k) = argmax_{g,k} Y_{g,k}$ if user $k$ satisfies the channel gain of all the group subchannels
    Allocate user$\mathscr{k}$ to group $\mathscr{g}^*$by $\alpha_{\mathscr{g}^*,\mathscr{k}} = 1$
    Delete users from user set by $\varkappa$= $\varkappa \backslash \mathscr{k}$
**end for**
Output: $\alpha_{\mathscr{g},\mathscr{k}}$
```

VIII. Simulation Results

This section conducts extensive performance evaluationbetween the proposed algorithm and existing works [31-33][38] from the aspects including the objective function valueof **P0**, energy consumption, user satisfaction ratio, etc. Theperformance is evaluated from two aspects: offloading ratio determinationand multicast resource allocation including group-subchanneland group-user allocation. The primary simulationparameters are shown in **Table I**. All the schemes comparedin this section include:

TableI Experimental setup

Parameter	**Value**
$\mathcal{B}_0$	200-1000KHz
α	{1,10,1000}
γ	[0,1]
$\mathcal{g}$	{1,2,3,4,5,6,7,8}
θ	{0.1,1,10}

PS: The proposed scheme for joint offloading and resourceallocation to maximize the objective function value of **P0**.

CP1: Only considering offloading without resource allocation[31][38]. For the offloading decision, they consider binarytask division assumptions, i.e., offloading and not offloadingconditions.

CP2: Only considering resource allocation without offloading[32]. For group-user allocation, they assume that a subgroupcollects users that have similar channel qualities. Forgroup-subchannel allocation, they assume that resources areallocated to subgroup iteratively.

CP3: Only considering resource allocation without offloading[33]. For group-subchannel allocation, base layer hashigher priority for channel allocation, besides, base layerrequires subchannels with the largest equivalent channel gains.For group-user allocation, a normalized equivalent channelgain threshold is set to determine which users should beallowed to access the enhancement layers, and the users withhigh signal-to-noise ratio (SNRs) are selected as the enhancementlayer users. Note that, base layer and enhancementlayers can be mapping to our groups.

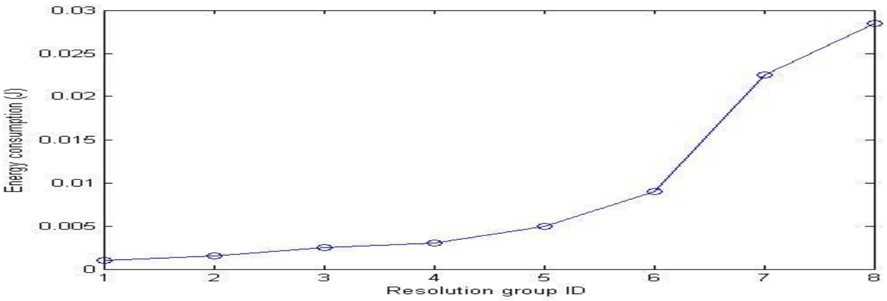

Fig. 3. user energy consumption under different resolution groups

Fig. 3 shows energy consumption for decoding video andreceiving traffic through the network card for each mobiledevice k on each resolution group g. The higher resolutiongroup ID with higher resolution video content, there will behigher energy consumption.

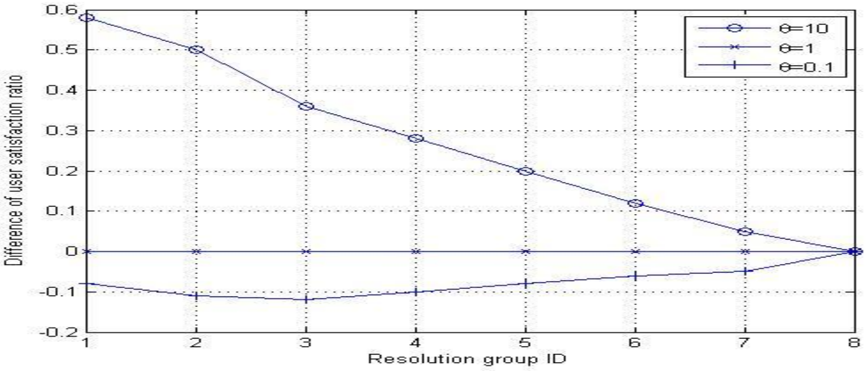

Fig. 4. user satisfaction ratio under different resolution groups

Fig. 4 shows user satisfaction ratio for each user k oneach resolution group g under different parameters θ in **Equation (12)**. Regarding the value of user satisfaction ratio under θ=1as the base-line, there exist a large gap between any adjacentresolution groups' satisfaction ratio, and a large gap betweenthe highest and lowest resolution groups under the case of$\theta = 10$, compared to that of $\theta = 0.1$.

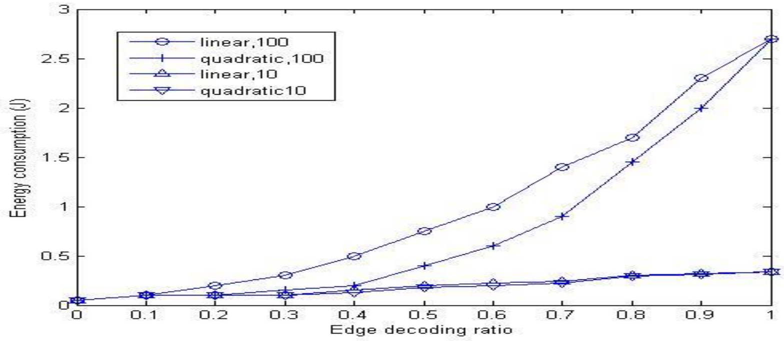

Fig. 5. user energy consumption under different offloading ratio

Fig. 5 shows user energy consumption for decoding videoand receiving traffic through the network card for each mobiledevice k versus different value of offloading ratio under thecases M_g, the data rate of the decoded video with resolution$g$ by the VDS in terms of γ_g, is linearfunction and quadratic function. User energy consumptionincreases with the value of offloading ratio $\gamma_g$increasing,which is because M_g increases with $\gamma_G$and e_k increases withM_g. Besides, different parameters in decoding equation, M_g,can achieve different value of energy consumption under thesame average value of offloading ratio.

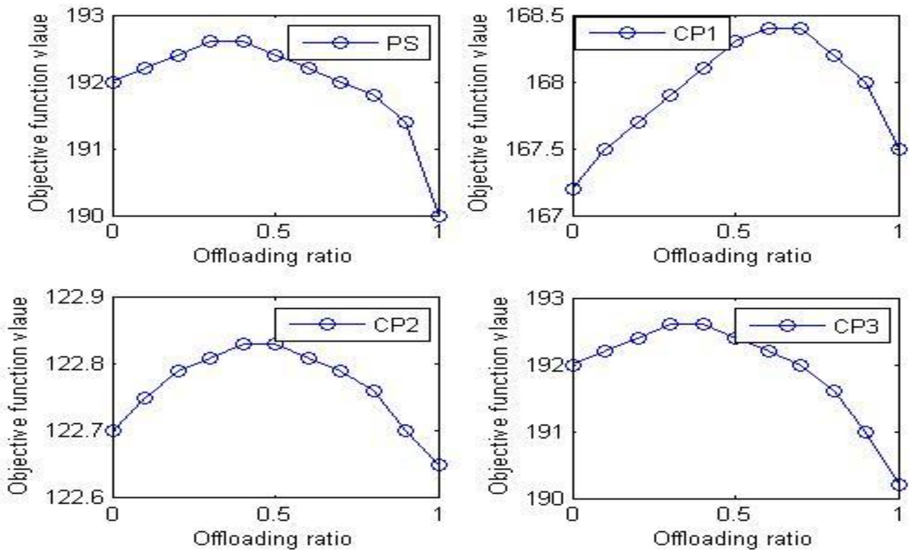

Fig. 6. objective function value versus offloading ratio for different schemes ($B_n = 3MHz$)

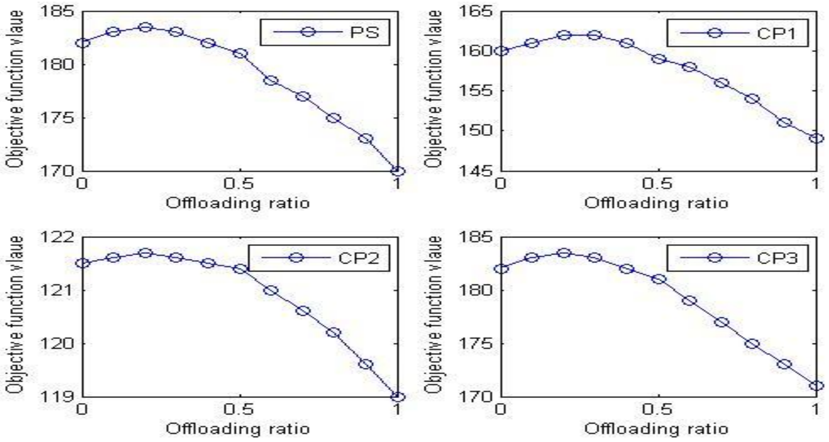

Fig. 7. objective function value versus offloading ratio for different schemes ($B_n = 400KHz$)

Figs. 6 and 7 show objective function values versusoffloading ratio for different schemes. Our proposed schemehas the highest value compared to those of other schemes.Specifically speaking, the objective function values of ourscheme can achieve 15% higher than CP1, 57% higher thanCP2, close performance to CP3, under $B_n = 3MHz$; 14%higher than CP1, 49% higher than CP2, close performance toCP3, under $B_n = 400KHz$; when binary variable are usedfor all the schemes in Figs. 6 and 7.

Moreover, the objective function values of our scheme canachieve 15% higher than CP1, 57% higher than CP2, closeperformance to CP3, under$B_n = 3MHz$; 14% higher thanCP1, 50% higher than CP2, close performance to CP3, under$B_n = 400KHz$; when relaxing binary variable forall the schemes. By relaxing the offloadingratio from integer to real, it can help increase the performanceamong all existing schemes. Moreover, the schemes can obtainhigher value when releasing the constraint that the offloadingratio $\gamma_g$can only be chosen as binary conditions.

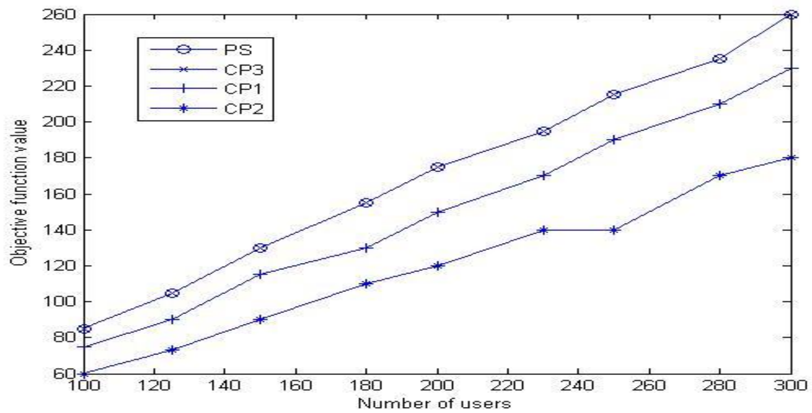

Fig. 8. objective function value versus the number of users for different schemes

Fig. 8 shows objective function value versus the numberof users for different schemes. Bandwidth of each subchannelis *150kHz*, and total bandwidth is *7.5MHz* under 50 subchannels.Our scheme can achieve higher performance thanthat of other schemes, which is due to channel gain andclock frequency of users are considered when applying userallocation.

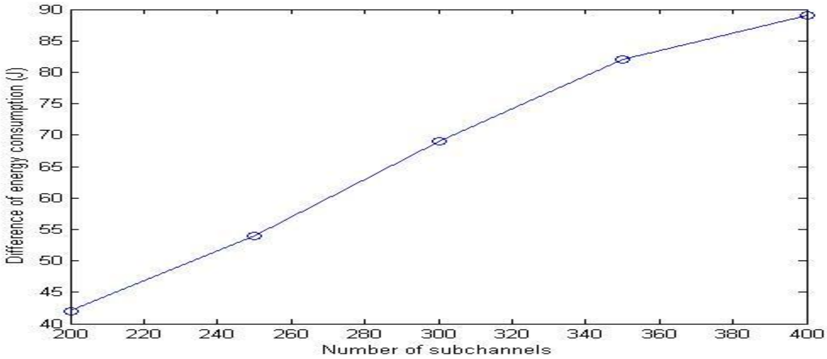

Fig. 9. difference of energy consumption versus the number of subchannels

Fig. 9 shows energy consumption versus the number ofsubchannels for different schemes. As the number of subchannelsincreasing, the bandwidth of each subchannel decreasingwhen total bandwidth is fixed, resulting in smaller group datarate and higher user energy consumption. As a result, thereis increasing objective function value when user satisfactionratio is constant. Our scheme can achieve the highest objectivefunction value than those of other schemes.

IX. Conclusion

This paper analyzes general video decoding and multicasttransmitting for massive low-end mobile devices in widely existedwireless networks. This problem is divided into: a novelvideo decoding architecture design, and resource allocationbased on the novel architecture. First, a novel architectureof real-time video decoding for computation offloading isintroduced, considering limited battery supply, constrainedcomputation capability, and dynamic traffic transmission bandwidth.Second, on the basis of the proposed architecture, ajoint

computation offloading and multicast resource allocationoptimization problem is proposed to maximize user satisfactionratio and minimize energy consumption. Third, a feasibilitycondition of the optimization problem is introduced interms of the computational task offloading for real-time videowith multiple resolutions. Forth, a low-complexity sub-optimalscheme with proved computational complexity is proposedby dividing the original NP-hard optimization problem intosub-problems to accomplish group-user allocation, group-subchannel allocation and offloading ratio calculation. Thispaper proposed a theoretical basis which can be potentiallyused as a guidance to the design and implementation of futuremobile edge decoding applications.

APPENDIX A
Proof of Theorem 2

Let us assume the task set is not schedulable, i.e., the task may excess the energy threshold. It is assumed that the first task misses the energy threshold has energy consumption value of$x$. Let $x0$ be the energy value before x, i.e., $x0 < x$. Therefore, the necessary condition to have energy threshold misses is that the demand energy consumption from $x0$ to x with absolute threshold less than or equal to x is larger that $x - x0$ [28]. There we have:

$$x - x0 < \sum_{T_g \in T} \sum_{\tau_{g,i} \in T_g} dbf \quad (\tau_{g,i}, x - x0) \leq_1 \left(\sum_{T_g \in T} \sum_{\tau_{g,i} \in T_g} \left(\int_0^1 \frac{V(\gamma_{g,i})}{e_k} d(\gamma_{g,i}) \right) \right) * (x - x0) \quad (20)$$

where $\leq_1$ coms form **Theorem 1**.

The equation $\int_0^1 \frac{V(\gamma_{g,i})}{e_k} d(\gamma_{g,i})$ gives the upper boundary in terms of energy consumption for task$\tau_{g,i}$ under all values of $\gamma_{g,i}$. There we have the energy threshold miss condition by:

$$1 < \sum_{T_g \in T} \sum_{\tau_{g,i} \in T_g} \left(\int_0^1 \frac{V(\gamma_{g,i})}{e_k} d(\gamma_{g,i}) \right) \quad (21)$$

AppendixB
Proof of Theorem 3

The summation energy of users in the objective function of **P1** equals to the summation energy of groups by the transformation below:

$$q \sum_k \sum_g \alpha_{g,k} \left(\frac{I(F_k) C_g}{F_k} + \frac{P_{NIC} \ (M_g + C_g)}{O_g}\right) = q \sum_g \sum_k \alpha_{g,k} \left(\frac{I(F_k) C_g}{F_k} + \frac{P_{NIC} \ (M_g + C_g)}{O_g}\right) (22)$$

Further, finding the minimal energy of the summation of groups is equivalent to calculate the minimal energy for each group individually, because users are distinct and group-user allocation index is given. Thereafter, we focus on calculating the minimal energy consumption for each group.

The energy equation in **Equation (17)** for each group has only one minimal value when $M_g$is either a linear function or a quadratic function in terms of γ_g. This is because the minimal value for a linear function can be obtained at the endpoints, i.e., $\gamma_g = 0$ or $\gamma_g = 1$. Moreover, the minimal value for a quadratic function may be obtained at the endpoints or the point which achieves the derivative of the function equals to zero, i.e., $\gamma_g = arg \quad_{\gamma_g} (\frac{\partial M_g}{\partial \gamma_g} = 0)$. Therefore, all the minimal value for groups can be calculated by above results.

Appendix C
Proof of Theorem 4

The **Constraint C3** gives each group an effective rangeFR_g, which should satisfy $FR_g \cap [0,1] \neq \Phi$. The range $FR_g \cap [0,1]$may not include the value $\widehat{\gamma_g}$obtained from **Theorem 3.** Therefore, **Constriant C1** drives us to move $\widehat{\gamma_g}$into the range of $FR_g \cap [0,1]$, which is then analyzed under different cases whereM_g is a linear function or a quadratic function.

Considering the case where$M_g$is a linear function, if $\widehat{\gamma_g} = 0$, we increase $\widehat{\gamma_g}$obtained from **Theorem 3** until reaching the left-hand side of the range $FR_g \cap [0,1]$. If $\widehat{\gamma_g} = 1$, we decrease $\widehat{\gamma_g}$obtained from **Theorem 3** until reaching the right-hand side of the range $FR_g \cap [0,1]$.

Taking the case where $M_g$is aquadratic function into consideration, if $\widehat{\gamma_g}$equals to either 0 or 1, follow the process above. When$0 < \widehat{\gamma_g} < 1$, if $\widehat{\gamma_g}$is outside the feasible range of $FR_g \cap [0,1]$, move $\widehat{\gamma_g}$to the range. Otherwise, if $\widehat{\gamma_g}$is inside the feasible range of $FR_g \cap [0,1]$, we can move $\widehat{\gamma_g}$in the range with bi-direction (left-hand side and right-hand side).

Appendix D
Proof of Theorem 5

If**Constraint C4**cannot be satisfied, we need to decrease obtained $\breve{\gamma}_g$from **Theorem 4** within the feasible range of $FR_g \cap [0,1]$. Then, we use the system determined step size $\Delta\gamma_g$to find a group which will obtain the minimal energy increase. Then we decrease $\breve{\gamma}_g$with $\Delta\gamma_g$and then renew **Constraint C2**.

For a linear objective function in**P1**, the slope of the energy function againstγ_g for each group$g$ is a constant, whenR_g has equal values among$g$. Meanwhile, the slope of the energy function multiplied by given $R_g$is also a constant even when$R_g$has unequal values among$g$. Therefore, we can increase a unit energy of the objective function value each time and find the group who can fastest satisfy **Constraint C4**. Note that, the chosen group will always be chosen since it frequently reaches the maximal decrease for **Constraint C4** until the variable $\gamma_g$cannot be changed any more. Under this case, we use the remaining groups for the remaining iteration process.

Appendix E
Proof of Theorem 6

The user allocated to current subchannel in group $g$will not degrade other subchannels in group when this user is allocated to those group subchannels. The above situation can be guaranteed if we use user criteria: the channel gain for this user at the current channel is less than the case when this user on other channels in group g, i.e.,$h_{c|current,k} < h_{c|others,k}$.

References

[1]. White paper: Cisco VNI Forecast and Methodology, 2015-2020.

[2]. S.Wang, D. Zhou, J. Zhou, T. Yoshimura, and S. Goto, VLSI implementation of HEVC motion compensation with distance biased direct cache mapping for 8K UHDTV applications, IEEE Transactions on Circuits and Systems for Video Technology, 27(2), 2017, 380-393.

[3]. Y.Kusakabe, Y. Ikeda, N. Shirai, K. Masaoka, T. Uamashita, Y. Nishida, T. Ikeda, and M. Sugaware, Extended image dynamic range system for UHDTV broadcasting, SMPTE Motion Imaging Journal, 125(4), 2016, 1-8.

[4]. S.Hara, A. Hanada, I. Masuhara, T. Yamashita, and K. Mitani, Celebrating the launch of 8K/4K UHDTV satellite broadcasting and progress on full-featured 8K UHDTV in Japan, SMPTE Motion Imaging Journal,127(2), 2018, 1-8.

[5]. S.Jeon, S. Kim, S. Hahm, Z. Yim, and Y. W. Suh, Laboratory measurement to provide threshold of visibility for terrestrial 4K-UHDTV broadcasting based on HEVC over DVB-T2, Journal of Broadcast Engineering, 21(4), 2016, 506-514.

[6]. S.Petrangeli, J. van der Hooft, T. Wauters, R. Huysegems, P. R. Alface, T. Bostoen, and F. D. Turck, Live streaming of 4K ultra-high definition video over the Internet,Proc.7thACM International Conference on Multimedia Systems (MMSys),Klagenfurt am Worthersee, Austria, 2016, 1-4.

[7]. H. Yamashita, H. Aoki, K. Tanioka, T. Mori, and T. Chiba, Ultra-high definition (8K UHD) endoscope: our first clinical success, Springer-plus, 5(1), 2016, 1-5.

[8]. H. T. Chang, H. W. Peng, and C. H. Tsai, CUDA-accelerated rendering of fireworks in nearly ultra high definition videos, Proc. 2ndIEEE International Conference don Multimedia Big Data (BigMM),Taipei, Taiwan, 2016, 251-254.

[9]. F. Xie, M. T. Pourazad, P. Nasiopoulos, and J. Slevinsky, Determining bitrate requirement for UHD video content delivery, Proc. IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, USA, 2016, 241-242.

[10]. Y. Wu, G. Min, and L. T. Yang,Performance analysis of hybrid wireless networks underbursty and correlated traffic, IEEE Transactions on Vehicular Technology, 62(1), 2013, 449-454.

[11]. G. Min, Y. Wu, and A. Y. Al-Dubai, Performance modelling and analysis of cognitive mesh networks, IEEE Transactions on Communications, 60(6), 2012, 1474-1478.

[12]. Y. Wu, G. Min, and A. Y. Al-Hubai, A new analytical model for multi-hop cognitive radio networks, IEEE Transactions on Wireless Communications, 11(5), 2012, 1643-1648.

[13]. Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, Mobile-edge computing: partial computation offloading using dynamic voltage scaling, IEEE Transactions on Communications, 64(10), 2016, 4268-4282.

[14]. K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, G. Zhang, S. Maharjan, and Y. Zhang, Energy-efficient offloading for mobile edge computing in 5G heterogenous networks, IEEE Access, 4(x),2016, 5896-5907.

[15]. P. Mach and Z. Becvar, Cloud-aware power control for real-time application offloading in mobile edge computing, Transactions on Emerging Telecommunications Technologies, 27(5), 2016, 648-661.

[16]. S. Sardellitii, G. Scutari, and S. Barbarossa, Joint optimization of radio and computational resources for multicell mobile-edge computing, IEEE Transactions on signal and information processing over networks, 1(2), 2015, 89-103.

[17]. A. Al-Shuwaili and O. Simeone, Energy-efficient resource allocation for mobile edge computing-based augmented reality applications, IEEE Wireless Communications Letters, 6(3), 2017, 398-401.

[18]. P. Markthub, A. Nomura, and S. Matsuoka, Reducing remote GPU execution's overhead with mrCUDA, Proc. GPU Technology Conference,2016, 1-1.

[19]. U. P. Moravapalle and R. Sivakumar, Peek: a mobile-to-mobile remote computing protocol for smartphones and tablets, Proc. International Conference on Computing, Networking and Communications (ICNC), Hawaii, USA, 2016, 1-6.

[20]. J. Lee, K. Choi, Y. Kim, H. Han, and S. Kang, Exploiting remote GPGPU in mobile devices, Cluster Computing, 19(3), 2016, 1571-1583.

[21]. V. A. Smirnov, E. V. Korolev, and O. I. Poddaeva, Cloud environments with GPU virtualization: problems and solutions, Proc. International Conference on Data Mining, Electronics and Information Technology (DMEIT), Pattaya, Thailand,2015, 147-154.

[22]. F. Silla, J. Prades, S. Iserte, and C. Reano, Remote GPU virtualization: is it useful?,Proc. 2nd IEEE International Workshop on High-Performance Interconnection Networks in the Exascale and Big-Data Era (HiPINEB),Barcelona, Spain,2016, 41-48.

[23]. F. Lamberti and A. Sanna, A streaming-based solution for remote visualization of 3D graphics on mobile device, IEEE Transactions on visualization and computer graphics, 13(2), 2007, 247-260.

[24]. C. Reano, F. Silla, A. J. Pena, G. Shainer, S. Schultz, A. Castello, E. S. Quintana-Orti, and J. Duato, POSTER: boosting the performance of remote GPU virtualization using infiniband connect-IB and PCIe 3.0.,2014, 266-267.

[25]. Y. Iida, Y. Fujii, T. Azumi, N. Nishio, S. Kato, GPUrps: exploring transparent access to remote GPUs, ACM Transactions on Embedded Computing Systems, 16(1), 2016, 1-25.

[26]. C. Reano and F. Silla, Tuning remote GPU virtualization for infiniband networks, Journal of Supercomputing, 72(12), 2016, 4520-4545.

[27]. C. Reano, F. Silla, A. Castello, A. J. Pena, R. Mayo, E. S. Quintana-Orti, and J. F. Duato, Improving the user experience of the rCUDA remote GPU virtualization framework, Concurrency and Computation: Practice and Experience, 27(14), 2015, 3746-3770.

[28]. W. Liu, J. J. Chen, A. Toma, T. W. Kuo, and Q. Deng, Computation offloading by using timing unrealizable components in real-time systems, Proc. 51st IACM/EDAC/IEEE Design Automation Conference (DAC), San Francisco, CA,2014, 1-6.

[29]. W. Ji, P. Frossard, B. W. Chen, and Y. Chen, Profit optimization for wireless video broadcasting systems based on polymatroidal analysis, IEEE Transactions on Multimedia, 17(12), 2015, 2310-2327.

[30]. A. Toma, S. Pagani, J. J. Chen, W. Karl, and J. Henkel, An energy-efficient middleware for computation offloading in real-time embedded systems, Proc. 22ndIEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), Daegu, South Korea, 2016, 228-237.

[31]. J. Cheng, Y. Shi, B. Bai, and W. Chen, Computation offloading in cloud-RAN based mobile cloud computing system, Proc. IEEE International Conference on Communications (ICC),Kuala Lumpur, Malaysia, 2016, 1-6.

[32]. S. Pizzi, M. Condoluci, G. Araniti, A. Molinaro, A. Iera, and G. M. Muntean, A unified approach for efficient delivery for unicast and multicast wireless video services, IEEE Transactions on Wireless Communications, 15(12), 2016, 8063-8076.

[33]. L. Chen,Layered multicast resource allocation with limited feedback scheme in single frequency networks, Wireless Personal Communications, 87(4), 2016, 1131-1146.

[34]. Y. Liu, X. Li, H. Ji, K. Wang, and H. Zhang, Joint aps selectin and resource allocation for self-healing in ultra dense network, Proc. IEEE International Conference on Computer, Information and Telecommunication Systems (CITS),Kunming, China,2016, 1-5.

[35]. M. Condoluci, G. Araniti, A. Molinaro, and A. Iera, Multicast resource allocation enhanced by channel state feedbacks for multiple scalable video coding streams in LTE networks, IEEE Transactions on Vehicular Technology, 65(5), 2016, 2907-2921.

[36]. C. K. Tan, T. C. Chuah, and S. W. Tan, Adaptive multicast scheme for OFDMA based multicast wireless systems, Electronics Letters, 47(9), 2011, 570-572.

[37]. T. Liu, H. Xia, and C. Feng, AQoS-based multi-rate multicast scheme over heterogeneous cellular network, Proc. 13th International Symposium on Wireless Communication Systems (ISWCS), Poznan, Poland,2016, 292-296.

[38]. Q. Zhao, T. You, X. Ma, Y.Mao, S. Leng, N. Yang, and Z. Zhao, Mobile edge decoding for saving energy and improving experience, Proc. 10th IEEE International Conference on Internet of Things (iThings), Exeter, UK, 2017, 475-482

QuanxinZhao "Joint resource allocation and edge computing forreal-time wireless video transmission* "International Journal of Engineering Science Invention (IJESI), vol. 07, no. 07, 2018, pp 24-39