

QoE-Driven Secure Video Transmission in Cloud-Edge Collaborative Networks

Tantan Zhao, Lijun He, Xinyu Huang, Fan Li

Abstract—Video transmission over the backhaul link in cloud-edge collaborative networks usually suffers security risks, which is ignored in most of the existing studies. The characteristics that video service can flexibly adjust the encoding rates and provide acceptable encoding qualities, make the security requirements more possible to be satisfied but tightly coupled with video encoding by introducing more restrictions on edge caching. In this paper, by considering the interaction between video encoding and edge caching, we investigate the quality of experience (QoE)-driven cross-layer optimization of secure video transmission over the wireless backhaul link in cloud-edge collaborative networks. First, we develop a secure transmission model based on video encoding and edge caching. By employing this model as the security constraint, then we formulate a QoE-driven joint optimization problem subject to limited available caching capacity. To solve the optimization problem, we propose two algorithms: a near-optimal iterative algorithm (EC-VE) and a greedy algorithm with low computational complexity (Greedy EC-VE). Simulation results show that our proposed EC-VE can greatly improve user QoE within security constraints, and the proposed Greedy EC-VE can obtain the tradeoff between QoE and computational complexity.

Index Terms—QoE, cross-layer optimization, wireless backhaul links security, edge caching, video encoding.

I. INTRODUCTION

With the rapid development of video processing and mobile communication technology, large numbers of ultra-high-definition (UHD) video-on-demand (VoD) facing massive links have emerged [1]. The number of video requests is growing dramatically, which leads to the heavy traffic load of networks and high latency of users. These factors could bring poor quality of experience (QoE) for users. Luckily, edge caching (EC) technology, as one of the key technologies of the cloud-edge collaboration networks in the fifth generation (5G) system, provides a promising method to handle these challenges [2], [3]. Since EC servers are closer to the users at the network edge and own powerful intelligent storage capabilities to precache hot videos, massive video contents

can be delivered directly to users from EC servers to reduce backhaul traffic burden and transmission latency [4]–[6]. In addition to transmission latency, video encoding quality is also a key factor affecting QoE. Video encoding quality is influenced by video encoding parameters: video encoding rate. Higher video coding rate means larger total size of the video packets and will bring higher received video quality. Transmission latency depends on both video encoding parameters and caching strategy. This is because most of the transmission time is spent on downloading the uncached video packets over the backhaul link from MBS. Generally speaking, high video encoding quality and low transmission latency imply high QoE. While in the practical scenarios with fixed available caching capacity, too high video encoding quality may result in too high transmission latency and vice versa. To maximize QoE, we need to find the tradeoff between video encoding quality and transmission latency by jointly optimizing video encoding parameters and caching strategy subject to limited available caching capacity. Therefore, how to balance video encoding characteristics and edge caching to improve user QoE is worth studying.

Recently, the copyright protection of video is becoming more and more important. Moreover, the emerging paid VIPs and confidential video services make the secure transmission of video contents become an issue that cannot be ignored in cloud-edge collaborative networks. To be specific, for the video transmission in cloud-edge collaborative networks, when video data is transmitted from the cloud server to the user in unicast mode, the eavesdropping user and the legitimate user receive information on the same frequency channel at the same time. Once channel condition of the eavesdropper is better than that of the legitimate user, the eavesdropper could successfully intercept the video packets, which causes data leakage and infringes on video copyright and benefits of users. Therefore, secure video transmission is important.

For the transmission security of wireless links from the EC servers to the users, the existing physical layer secure (PLS) mechanism, which has been studied intensively and extensively [7]–[10], is sufficient. However, the PLS mechanism is complicated because it could ensure the security of each video packet. If it is utilized to ensure the transmission security of wireless backhaul links in the distributed cache scenarios, where an EC server may be connected to multiple base stations, the system overhead could be unbearable. Therefore, it is an urgent issue to develop a secure mechanism with low complexity to ensure secure transmission of wireless backhaul links, rather than using the strict and complicated traditional PLS mechanism.

Copyright (c) 2021 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

Manuscript received June 07, 2021; revised August 26, 2021; accepted October 25, 2021. The review of this paper was coordinated by Prof. Kaigui Bian. This work was supported in part by the National Natural Science Foundation of China under Grant 62071369 and in part by the Key Research and Development Program of Shaanxi Province under Grant 2020KW-009. (Corresponding author: Fan Li.)

T. Zhao, L. He, X. Huang and F. Li are with the Ministry of Education Key Laboratory for Intelligent Networks and Network Security, School of Information and Communications Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: {zhao10111446, xinyu_huang}@stu.xjtu.edu.cn; {lijunhe, lifan}@mail.xjtu.edu.cn).

The flexible and cost-effective wireless backhaul has become an attractive alternative to wired backhaul for small cell networks. However, wireless backhaul is vulnerable to wiretapping due to the intrinsic broadcasting nature of wireless channels. Therefore, the issue of secure backhaul links has aroused wide interest of researchers in recent years. Considering unreliable wireless backhaul connections, the authors in [11] investigated the secrecy performance of cooperative single carrier HetNets. The authors in [12] studied the effect of unreliable wireless backhaul links on the secrecy performance of an energy harvesting relay network. Different transmitter selection schemes were proposed to enhance the secrecy. Wireless backhaul along with secrecy was considered for CR network in [13] and [14].

However, most of the existing studies related to secure backhaul links only analyzed the significant negative impact of the backhaul unreliability on secure transmission instead of developing methods to enhance the reliability of wireless backhaul links. Furthermore, the issue of the secure backhaul links in the cloud-edge collaborative networks is rarely considered. The authors in [42] considered the security risks of the backhaul links when studying the optimal cache placement in edge heterogeneous networks. Nevertheless, the issue aimed at general data transmission, ignoring the characteristics of the video applications, which resulted in the poor performance for video transmission applications. Therefore, secure video transmission through the backhaul links in the cloud-edge collaborative networks is still an open question.

To address the abovementioned problems in secure video transmission via backhaul links in cloud-edge collaborative networks, we propose a QoE-driven cross-layer optimization algorithm to improve video encoding quality, reduce transmission latency and guarantee security of backhaul links. The main contributions of this paper can be summarized as follows.

1) *Problem Formulation of Joint Optimization of Video Encoding and Edge Caching*: Guaranteeing secure video transmission is of equal importance in the cloud-edge collaborative network besides optimizing QoE. Since video service can flexibly provide different acceptable encoding qualities to adapt to the real scenario's requirements by adjusting the encoding parameters correspondingly, the extra security requirements can be satisfied with higher possibility but has to be tightly coupled with video encoding by introducing more restrictions on edge caching. Therefore, how to find the tradeoff among transmission latency, video encoding quality and security becomes the new challenge. To address this issue, we first develop a secure transmission model based on video encoding and edge caching. By employing this model as the security constraint, we comprehensively consider the interaction of video encoding and edge caching and then formulate a QoE-driven joint optimization problem. The optimization problem aims to optimize both video encoding and caching strategies simultaneously subject to the limited available caching capacity.

2) *Near-Optimal and Suboptimal Solutions*: The formulated joint optimization problem is a high-dimension non-linear mixed integer programming problem, which makes it more difficult to achieve analytical solution. On one hand, the non-

smooth objective function, the non-convex constraints and multiplication among the optimization variables make the original optimization problem intractable. On the other hand, the high dimensionality of optimization variables makes it difficult to obtain the optimal solution. Therefore, we propose a near-optimal algorithm to solve it. To be specific, efficacious transformations are performed on the original intractable problem to obtain its relaxing solution. Then, Boolean bound is developed to obtain the near-optimal solution by iteration until all Boolean variables are integers. Furthermore, considering that the near-optimal algorithm is time-consuming, we also propose a suboptimal algorithm based on the greedy method with low computational complexity to get the tradeoff between the user QoE and the computational complexity.

The rest of this paper is organized as follows. Section II reviews the related works. In Section III, we introduce the system model, which includes the framework of QoE-driven cross-layer system and describes the secure transmission model. In Section IV, we formulate the video encoding quality maximization and latency minimization problem under the secure constraints. In Section V we propose a near-optimal solution based on joint optimization of video encoding and edge caching. In Section VI, we propose a suboptimal solution based on greedy method. Section VII gives the experimental results, and discusses the performance gains of the proposed algorithms compared to the existing algorithms. Finally, Section VIII concludes this paper.

II. RELATED WORK

In this section, we will review and analyze the progress of existing related work from the following two aspects.

A. QoE-Driven Edge Video Caching

The edge servers are closer to users and have powerful storage capabilities. By caching multimedia contents during off-peak hours, delay and congestion can be reduced [15], thereby improving mobile user QoE [16], [17].

To exploit the storage capacity of edge servers, the authors in [18] proposed a distributed cache optimization algorithm through belief propagation of heterogeneous networks, which greatly reduced the download delay of users. From the perspective of the basic information theoretical limit of EC, the authors in [19] gave the best trade-off between latency and caching. Considering the particularity of video transmission, the researchers in [20] proposed a co-caching strategy for small-cell base stations (SBSs) and mobile devices, which greatly reduced the delay in video content delivery.

In addition to latency, the popularity of video contents also determines whether the cached videos meet the needs of users. The authors in [21], [22] proposed a context-aware caching scheme to predict the popularity and learn the video popularity of a specific context online, which is used to determine cache replacement strategy. However, with the increasing heterogeneity of user groups demanding specific video content, the issue of caching mobile video streaming has become a more complicated task. Therefore, there are also many papers that use machine learning to predict preferences of users [24]–[28].

The studies mentioned above are for traditional video streaming, to improve the QoE of dynamic adaptive streaming users. The authors in [29] studied the QoE-driven optimization of EC placement for dynamic adaptive video streaming, which considers the different rate-distortion characteristics of videos and the coordination among distributed EC servers. The authors in [30] proposed a refined caching update strategy based on video popularity, content importance, and user playback status, which ensured that the video segments to be played by users could be cached in time. To further improve user QoE, multiple modules in addition to EC are considered to be optimized comprehensively. The authors in [31] proposed to enhance the QoE-aware wireless edge caching with bandwidth provisioning in software-defined wireless networks. Specifically, a joint optimization mechanism of caching strategy, bandwidth configuration, and adaptive video streaming was designed to reduce delay and improve QoE. In addition, the researchers in [32] proposed a joint scheme of caching strategy, power allocation, user association, and adaptive video streaming to improve the system spectrum efficiency and user QoE. However, the scheme in [32] will cause a greater burden on backhaul traffic. Therefore, in [33], the authors studied the joint optimization scheme of mobile terminal QoE and backhaul traffic, which reduced power consumption and backhaul traffic, meanwhile improving QoE. Although the above studies aim to improve the QoE of mobile users, security issues in the cloud-edge collaborative networks were not considered.

B. Caching-Based Secure Transmission

With the increasing complexity of network environment, legal transmission in the networks is inevitably attacked by external malicious nodes, or the SBS originally used for caching becomes a malicious node and implements eavesdropping. Therefore, caching-based secure transmission scheme has aroused wide interest of researchers. The secure schemes proposed by these studies can be classified as two categories according to different methods used to achieve security, including encryption and coding based secure schemes and wireless security mechanism based secure schemes.

For the encryption and coding based secure scheme, in [34], the authors studied the secure device-to-device coding and caching scheme in the sense of information theory. The authors in [35] proposed a decentralized secure coding caching method in wireless ad-hoc networks. The researchers in [36] studied the attack models in EC systems and proposed lightweight authentication and secure collaborative caching schemes based on reinforcement learning techniques for the mobile edge caching. The authors in [37] proposed a secure caching solution for disaster backup in mobile social networks using fog computing. In [38], a secure caching scheme in heterogeneous networks for multi-homing users is designed. A trust mechanism and a Chinese remainder theorem-based privacy preservation protocol is proposed.

For the wireless security mechanism based secure scheme, in [39], a caching scheme to enhance PLS for a backhaul-limited cellular network was proposed. The authors focused on minimizing the transmit power subject to the secrecy rate,

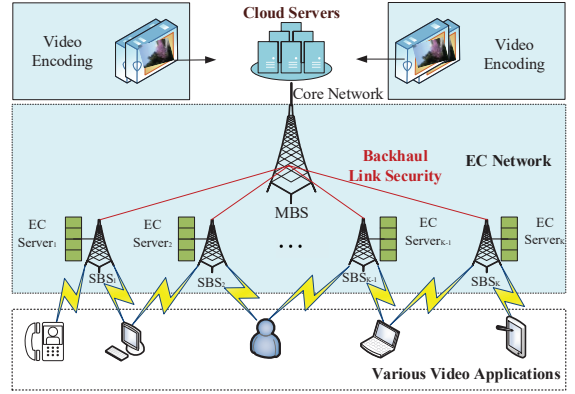


Fig. 1. VoD downlink transmission in cloud-edge collaborative networks.

where the backhaul links are considered as wired. In [40], the authors studied the secure transmission problem in a cache-assisted heterogeneous networks. To realize the secure and energy-efficient transmission, a joint cache placement and file delivery scheme was proposed. In [41], the authors investigated the secure transmission delay minimization problem in an edge cache-assisted millimeter-wave cloud radio access network. The authors designed beamforming to minimize the fronthaul transmission delay and access transmission delay.

Apart from the above studies, for the security risks of backhaul links, the authors in [42] studied the optimal cache placement problem under secrecy constraints of backhaul links in edge heterogeneous networks. Specifically, the authors proposed an edge caching strategy to minimize the average backhaul links rate and ensure the security of backhaul links. However, what has been studied in this study is a secure transmission scheme for general data. The unique characteristics of videos, which allow distortion within acceptable range, is ignored. Generally speaking, different encoding parameters correspond to different encoding qualities, which bring different QoE levels. Therefore, the proposed secure scheme is not specific to video transmission.

III. SYSTEM MODEL

A. Framework

In this paper, we consider a VoD downlink scenario in the cloud-edge collaborative networks, where multiple users request video files from the cloud servers via a macro-cell base station (MBS), which has access to the cloud servers through wired core network. SBSs are deployed in the coverage area of the MBS to serve user video requests and communicate with the MBS via wireless backhaul links. As shown in Fig. 1, the solid red lines denote the wireless backhaul links between the MBS and SBSs, following [43]–[45]. The number of EC servers is K , and let \mathcal{K} denote the set of K EC servers and the caching capacity of each EC server is $\Phi_k, \forall k \in \mathcal{K}$. Each EC server is connected to an SBS. A user requesting video files is initially served by SBSs. For video services without packet loss, n encoding packets are required to successfully decode video information. If the number of video packets sent by all EC servers within the connection range is m and less

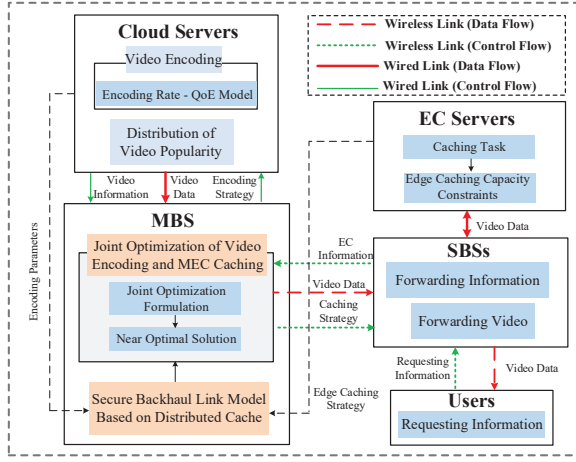


Fig. 2. QoE-driven joint optimization scheme of video encoding and edge caching based on secure backhaul links model.

than n , the decoding conditions are not met. Then, the MBS is contacted to send the remaining $n-m$ video packets to the user via backhaul links. Based on the above scenario, we propose a QoE-driven joint optimization scheme of video encoding and edge caching based on the secure transmission model for the backhaul links, as illustrated in Fig. 2.

Specifically, the whole block diagram of the joint optimization scheme contains the modules of MBS, cloud servers, SBSs, EC servers and users. First, the video encoding information, i.e., encoding rate, QoE model and the distribution of popularity, from the cloud servers, and the EC information, i.e., the caching capacity and the number of EC servers from the SBSs, are sent to the system analysis module MBS. Then, the optimal encoding and caching strategy based on the collected information and the secure transmission model of backhaul links are designed by the MBS and sent to the cloud servers and the SBSs, respectively. The secure transmission model of the backhaul links is based on video encoding and edge caching. This means that the encoding packet size, the number of encoding packets cached on the EC servers and the capacity of EC servers together determine whether the secure transmission can be achieved. Finally, the video streamings are encoded by the cloud servers according to the designed encoding strategy and cached on the EC servers according to the designed caching strategy after being transmitted from the cloud servers. Here we consider utilizing the adjustable encoding parameters and the distributed edge cache to avoid data leakage over the wireless backhaul links from the MBS to the SBSs and construct the secure transmission model of wireless backhaul links. Based on the secure transmission model, the QoE-driven joint optimization model of video encoding and edge caching is established.

B. Secure Transmission Model Based on Video Encoding and Edge Caching

For the high-definition (HD)/UHD VoD services, we assume that there are F original video streamings to be encoded and cached, and let \mathcal{F} denote the set of F video streamings. $p_j, \forall j \in \mathcal{F}$ indicates the requested probability of the j -th video file, which is ranked in descending order according to video

popularity. Without loss of generality, we can mark the most popular video file with index 1 and the least popular video file with index F . Specifically, the requested probability of a video file can be described as a Zipf distribution. Assuming that all video files are sorted according to their popularity, the probability of being requested for each video file can be obtained by the following formula:

$$p_j = \frac{1/j^{-\theta}}{\sum_{j=1}^F 1/j^{-\theta}}, 0 < \theta < 1, \forall j \in \mathcal{F} \quad (1)$$

where θ is the tilt factor of the distribution, which is used to control the distribution of video popularity, and the larger value means that the requested probability is more concentrated.

To be specific, the following assumptions are made when encoding and caching video streamings:

- 1) Each video file is encoded into n packets of different sizes, and let \mathcal{N} denote the set of n encoding packets. Let $s_{i,j}$ denote the size of the i -th video packet of the j -th video file. The n video encoding packets are cached distributedly on different EC servers, following [31] and [46].
- 2) $\tilde{m}_{k,i,j}$ denotes whether the i -th video packet of the j -th video file is cached on the k -th EC server ($\tilde{m}_{k,i,j} = 1$) or not ($\tilde{m}_{k,i,j} = 0$).

Based on the above assumptions, the cooperative caching strategy among multiple EC servers can be formulated as a matrix $\mathbf{M}_{K \times F}$ of size $K \times F$, where the rows indicate the EC servers, the columns indicate the video files, and the element $m_{k,j}$ represents the number of encoding packets of the j -th video file cached on the k -th EC server, obviously, $m_{k,j} = \sum_{i=1}^n \tilde{m}_{k,i,j}$. As a result, for each request, the number of encoding packets of the j -th video file that need to be transmitted through the backhaul links can be expressed as

$$R_j = n - \min \left(n, \sum_{k=1}^K m_{k,j} \right) \quad (2)$$

Meanwhile, assuming that the number of total requests during the whole delivery phase received by SBSs is Ψ , then the number of requests for the video file j received by SBSs is $\Psi_j = \Psi \cdot p_j, \forall j \in \mathcal{F}$. Therefore, for Ψ_j requests, the total number of encoding packets for the j -th video file transmitted through the backhaul links can be obtained as follows:

$$P_j = \Psi_j \cdot \left[n - \min \left(n, \sum_{k=1}^K m_{k,j} \right) \right] \quad (3)$$

For clarity, considering that $n \geq \sum_{k=1}^K m_{k,j}$ in our considered scenario, Eq. (3) can be further formulated as

$$P_j = \Psi_j \cdot \left(n - \sum_{k=1}^K m_{k,j} \right) \quad (4)$$

We know that for the video services without packet loss, secure transmission can be realized if and only if $P_j \leq n - 1$. Consequently, the secure transmission model of backhaul links based on the distributed edge cache can be expressed by

$$\Psi_j \cdot \left(n - \sum_{k=1}^K m_{k,j} \right) \leq n - 1 \quad (5)$$

After further simplification, we can get

$$\sum_{k=1}^K m_{kj} \geq n \cdot \left(1 - \frac{1}{\Psi_j}\right) + \frac{1}{\Psi_j} \quad (6)$$

Based on the above derivation, we know that the probability of secure transmission for the requested video is closely related to the encoding packets cached on EC servers and the number of video requests.

IV. PROBLEM FORMULATION

Based on the secure transmission model achieved for backhaul links, the QoE-driven joint optimization problem of video encoding parameters and edge caching strategy under the constraints of edge caching capacity and video encoding rate has been formulated. We define the optimization variables as $\tilde{\mathbf{M}}_{K \times n \times F} = \{\tilde{m}_{k,i,j}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ and $\mathbf{S}_{n \times F} = \{s_{i,j}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$, where $\tilde{m}_{k,i,j}$ and $s_{i,j}$ denote the caching strategy and the size of each video encoding packet, respectively. To improve user QoE, we construct the joint optimization problem expressed by Eq. (7) as follows:

$$\max_{\tilde{\mathbf{M}}, \mathbf{S}} Q(\tilde{\mathbf{M}}, \mathbf{S}) \quad (7a)$$

$$\text{s.t.} \sum_{j=1}^F \sum_{i=1}^n \tilde{m}_{k,i,j} s_{i,j} \leq \Phi_k, \forall k \in \mathcal{K} \quad (7b)$$

$$\sum_{k=1}^K \sum_{i=1}^n \tilde{m}_{k,i,j} \leq n, \forall j \in \mathcal{F} \quad (7c)$$

$$\sum_{k=1}^K \sum_{i=1}^n \tilde{m}_{k,i,j} \geq n \cdot \left(1 - \frac{1}{\Psi_j}\right) + \frac{1}{\Psi_j}, \forall j \in \mathcal{F} \quad (7d)$$

$$R_{\min,j} \leq \frac{\sum_{i=1}^n s_{i,j}}{T_d} \leq R_{\max,j}, \forall j \in \mathcal{F} \quad (7e)$$

$$\tilde{m}_{k,i,j} \in \{0, 1\}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F} \quad (7f)$$

where $Q(\cdot)$ is the objective function for different application demands, which can represent a single optimization objective or the combination of multiple optimization objectives, depending on the practical application demands. $R_{\min,j}, \forall j \in \mathcal{F}$ and $R_{\max,j}, \forall j \in \mathcal{F}$ denote the maximum and minimum encoding rates of the j -th video file, and the larger encoding rate means the higher QoE. T_d denotes the duration of each video file.

To be specific, in our considered VoD downlink scenario, the QoE is determined by video encoding quality and transmission latency. The optimization objective is to maximize the video encoding quality and minimize the transmission latency simultaneously. Then, the $Q(\cdot)$ can be defined as

$$Q(\tilde{\mathbf{M}}, \mathbf{S}) = \sum_{j=1}^F [f(s_{i,j}) - g(\tilde{m}_{k,i,j}, s_{i,j})], \quad (8)$$

where $f(s_{i,j})$ denotes the relationship between video encoding parameters and video encoding quality, which is determined by the chosen mean opinion score (MOS) model, and $g(\tilde{m}_{k,i,j}, s_{i,j})$ denotes the relationship among caching strategy, video encoding parameters and transmission latency.

In particular, according to the MOS model in [47], for the j -th video file, $f(s_{i,j})$ and $g(\tilde{m}_{k,i,j}, s_{i,j})$ can be expressed as

$$f(s_{i,j}) = C_{1j} \cdot \left(\frac{\sum_{i=1}^n s_{i,j}}{T_d}\right)^3 + C_{2j} \cdot \left(\frac{\sum_{i=1}^n s_{i,j}}{T_d}\right)^2 + C_{3j} \cdot \left(\frac{\sum_{i=1}^n s_{i,j}}{T_d}\right) + C_{4j} \quad (9)$$

where $C_{1j}, C_{2j}, C_{3j}, C_{4j}, \forall j \in \mathcal{F}$ are video encoding quality parameters which can be determined by the encoding rate range of the j -th video file for a given MOS model.

$$g(\tilde{m}_{k,i,j}, s_{i,j}) = V_j \cdot \left[\frac{p_j \cdot \left(\sum_{i=1}^n s_{i,j} - \sum_{i=1}^n \sum_{k=1}^K \tilde{m}_{k,i,j} s_{i,j}\right)}{R_{bk}} \right]^{\frac{2}{3}} \quad (10)$$

where $V_j, \forall j \in \mathcal{F}$ is the weighting coefficient of transmission latency for a given MOS model. R_{bk} is the transmission rate of the backhaul links.

Constraint (7b) indicates that the total size of video packets cached in each EC server should not be greater than the edge caching capacity $\Phi_k, \forall k \in \mathcal{K}$. Constraint (7c) indicates that the number of video encoding packets cached in all EC servers should not be larger than the total number of video encoding packets n . Constraint (7d) reveals the number of video encoding packets that need to be cached in EC servers to ensure the secure transmission of backhaul links. Constraint (7e) gives the maximum and the minimum limits of encoding rate for each video streaming. Constraint (7f) denotes caching strategy as the binary variables, which means whether the k -th EC server caches the i -th packet of the j -th video file or not.

By observing the formulated optimization problem in Eq. (7), we can know that caching strategy and video encoding parameters cooperatively affect the security of the transmission. Specifically, the security constraint in Eq. (7d) acts on caching strategy, meanwhile Eq. (7b) reveals that the video encoding parameters interact with caching strategy. Thus, security constraint in Eq. (7d) actually acts on caching strategy and video encoding parameters simultaneously. When the capacity of EC server is limited and security constraint is strict, video encoding parameters and caching strategy can be jointly optimized and cooperatively adjusted to ensure security. More video encoding packets could be cached on EC server in proper encoding rate to satisfy the secure constraint and the limited edge caching capacity.

V. NEAR OPTIMAL SOLUTION BASED ON JOINT OPTIMIZATION OF VIDEO ENCODING AND EDGE CACHING

The optimization problem formulated in section IV is an intractable problem. Because it is a high-dimension non-linear mixed 0-1 integer programming problem. First, the non-smooth objective function and the non-convex constraints make the original optimization problem intractable. Furthermore, the multiplication and coupling among the high-dimension optimization variables make the original optimization problem even more intractable. Finally, the high dimensionality of optimization variables make it difficult to obtain the optimal solution of the original optimization problem. To solve it, a near-optimal solution based on relaxation and

Boolean bound is proposed in this subsection. Specifically, the whole solving procedure can be divided into two steps, i.e., relaxation of the original high-dimension non-linear mixed 0-1 integer programming problem and the near-optimal solution based on the derived Boolean bound. In the following section of this paper, we will describe them in detail.

A. Relaxation of the Original High-Dimension Non-linear Mixed 0-1 Integer Problem

We first relax the original mixed 0-1 integer programming problem with integer constraints into the relaxing problem without integer constraints. Then, the relaxing solution of the relaxing problem can be obtained by utilizing GlobalSearch. We can easily get the relaxing problem, which can be expressed by the following:

$$\max_{\tilde{\mathbf{M}}, \mathbf{S}} \sum_{j=1}^F [f(s_{i,j}) - g(\tilde{m}_{k,i,j}, s_{i,j})] \quad (11a)$$

$$\text{s.t. (7b), (7c), (7d), (7e)} \quad (11b)$$

$$0 \leq \tilde{m}_{k,i,j} \leq 1, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F} \quad (11c)$$

Particularly, the relaxing processing is based on the fact that the optimal solution of the mixed integer programming, denoted as $\tilde{\mathbf{M}}_{opt}$ and \mathbf{S}_{opt} , will not be superior to the optimal solution of its relaxing problem, which is the upper bound of the original mixed integer programming. Here, the definition of the relaxing problem can be expressed as: the objective function and remaining constraints of the original mixed integer programming problem without considering the integer constraint (7f), which is replaced by continuous range constraint (11c).

The relaxing problem (11) above is a maximization problem, which is obtained by replacing the integer constraint (7f) in the original mixed 0-1 integer programming problem (7) with the continuous range constraint (11c) in the relaxing problem (11). For this maximization problem, as a typical reformulation method for solving optimization problem, we consider a minimization problem equivalent to the maximization problem (11) as follows:

$$\min_{\tilde{\mathbf{M}}, \mathbf{S}} \sum_{j=1}^F [g(\tilde{m}_{k,i,j}, s_{i,j}) - f(s_{i,j})] \quad (12a)$$

$$\text{s.t. (7b), (7c), (7d), (7e), (11c)} \quad (12b)$$

The minimization problem (12) is actually a non-linear optimization problem. Therefore, we consider utilizing the global optimal algorithm GlobalSearch provided by Global Optimization Toolbox of MATLAB to obtain its solution. Actually, the near-optimal solution of optimization problem (12) can be obtained by the use of GlobalSearch algorithm, which is a kind of heuristic search algorithm. We can denote the near-optimal solution of relaxing problem (12) as $\tilde{\mathbf{M}}_r$ and \mathbf{S}_r , and the corresponding objective function value is Q_r .

B. Near-Optimal Solution Employing Boolean Bound

To obtain the near-optimal solution of the original mixed integer programming, we employ the branch and bound method

[48]. Specifically, we can discuss the solution of the relaxing problem (12) in two cases.

In case one, the solution in $\tilde{\mathbf{M}}_r$ concerning integer-variables $\tilde{\mathbf{M}}_{K \times n \times F} = \{\tilde{m}_{k,i,j}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ are integers. Then, the near-optimal solution $\tilde{\mathbf{M}}_r$ and \mathbf{S}_r of the relaxing problem is also the near-optimal solution of the original non-linear mixed optimization problem and the near-optimal solution is obtained as $\tilde{\mathbf{M}}_{opt} = \tilde{\mathbf{M}}_r$, $\mathbf{S}_{opt} = \mathbf{S}_r$.

In case two, an element of solution $\tilde{\mathbf{M}}_r$ concerning integer-variables $\tilde{\mathbf{M}}_{K \times n \times F} = \{\tilde{m}_{k,i,j}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ is not integer. Then, the non-integer element should be branched. The whole procedure of branching can be divided into the following three steps:

• Deriving Boolean Bound

The integer variable $\tilde{m}_{k',i',j'}, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ is actually called Boolean variable. The near-optimal solution $\tilde{\mathbf{M}}_{opt}$ can be initialized as $\tilde{\mathbf{M}}_{opt} = \tilde{\mathbf{M}}_r$. Without loss of generality, in our non-linear mixed integer optimization problem to be solved, we use $\tilde{m}_{k',i',j'}, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ to denote the Boolean variable which does not satisfy the integer constraint and $b_{k',i',j'}, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ to denote its corresponding non-integer solution. Specifically, branch means that the relaxing problem (12) obtained from the original non-linear mixed integer problem should be further relaxed into two new sub-relaxing problems, which can be named sub-relaxing problem one and sub-relaxing problem two. Relaxing means adding two inequality constraints concerning Boolean variable $\tilde{m}_{k',i',j'}, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ to the relaxing problem (12) and keeping its original objective function and remaining constraints. To be specific, for the sub-relaxing problem one, the added inequality constraint can be expressed as $\tilde{m}_{k',i',j'} \leq \lfloor b_{k',i',j'} \rfloor, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$, while for the sub-relaxing problem two, the added inequality constraint is $\tilde{m}_{k',i',j'} \geq \lfloor b_{k',i',j'} \rfloor + 1, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$, where $\lfloor \cdot \rfloor$ means rounding down.

It can be observed that, the added inequality constraint $\tilde{m}_{k',i',j'} \leq \lfloor b_{k',i',j'} \rfloor, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ for the sub-relaxing problem one can combine with Eq. (11c) and merge into one equality constraint. Hence, the inequality constraint $\tilde{m}_{k',i',j'} \leq \lfloor b_{k',i',j'} \rfloor, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ in the sub-relaxing problem one can be equivalently transformed into equality constraint $\tilde{m}_{k',i',j'} = 0, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$, which is called Boolean bound of 0 branch. Similarly, the added inequality constraint $\tilde{m}_{k',i',j'} \geq \lfloor b_{k',i',j'} \rfloor + 1, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ for the sub-relaxing problem two can combine with Eq. (11c) and merge into one equality constraint. Therefore, the inequality constraint $\tilde{m}_{k',i',j'} \geq \lfloor b_{k',i',j'} \rfloor + 1, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$ in the sub-relaxing problem two can be equivalently transformed into the equality constraint $\tilde{m}_{k',i',j'} = 1, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$, which is called Boolean bound of 1 branch.

• Formulating Two Sub-relaxing Problems

Based on the obtained minimization relaxing problem (12) and Boolean bound of the 0-1 branches, we can easily get the sub-relaxing problem one and sub-relaxing problem two as follows:

Sub-relaxing problem one:

$$\min_{\tilde{\mathbf{M}}, \mathbf{S}} \sum_{j=1}^F [g(\tilde{m}_{k,i,j}, s_{i,j}) - f(s_{i,j})] \quad (13a)$$

$$\text{s.t. (7b), (7c), (7d), (7e)} \quad (13b)$$

$$\tilde{m}_{k',i',j'} = 0, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F} \quad (13c)$$

where constraint (13c) is the Boolean bound of 0 branch for the Boolean variable $\tilde{m}_{k',i',j'}, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$.

Sub-relaxing problem two:

$$\min_{\tilde{\mathbf{M}}, \mathbf{S}} \sum_{j=1}^F [g(\tilde{m}_{k,i,j}, s_{i,j}) - f(s_{i,j})] \quad (14a)$$

$$\text{s.t. (7b), (7c), (7d), (7e)} \quad (14b)$$

$$\tilde{m}_{k',i',j'} = 1, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F} \quad (14c)$$

where constraint (14c) is the Boolean bound of 1 branch for the Boolean variable $\tilde{m}_{k',i',j'}, \forall k' \in \mathcal{K}, \forall i' \in \mathcal{N}, \forall j' \in \mathcal{F}$.

• Obtaining Near-Optimal Solution

The sub-relaxing problem one formulated in Eq. (13) and the sub-relaxing problem two formulated in Eq. (14) can also be solved by GlobalSearch algorithm due to the fact that their forms are similar to the minimization relaxing problem (12).

There are only two possible solutions for the two sub-relaxing problems expressed by Eq. (13) and Eq. (14), i.e., infeasible solution and integer solution due to the Boolean bound of 0-1 branches. For the convenience of solving the problem, if the infeasible solution appears for the sub-relaxing problem, we can directly set it as positive infinity or negative infinity, which depends on whether the relaxing problem is a maximization problem or a minimization problem. Then, the near-optimal solution for one iteration of 0-1 branch and bound can be obtained by comparing the objective function values of two sub-relaxing problems to find the minimization branch and its corresponding solution, denoted as $\tilde{\mathbf{M}}_{min}$ and \mathbf{S}_{min} , which are used to update the near-optimal solution $\tilde{\mathbf{M}}_{opt}$ and \mathbf{S}_{opt} for each iteration by letting $\tilde{\mathbf{M}}_{opt} = \tilde{\mathbf{M}}_{min}$ and $\mathbf{S}_{opt} = \mathbf{S}_{min}$.

A new non-integer solution may appear in the updated solution $\tilde{\mathbf{M}}_{opt}$ due to the 0-1 branch and bound. Therefore, if there is still one non-integer solution in the obtained $\tilde{\mathbf{M}}_{opt}$, the 0-1 branch and bound should be executed continually based on the sub-relaxing problems formulated in Eq. (13) and Eq. (14). To be specific, Boolean bound concerning the new non-integer solution should be added to the sub-relaxing problems (13) and (14). Then, the near-optimal solution $\tilde{\mathbf{M}}_{opt}$ and \mathbf{S}_{opt} can be obtained by updating them as the near-optimal solution of the minimization branch of the current iteration of 0-1 branch and bound. Finally, the near-optimal solution of the original non-linear mixed integer optimization problem can be obtained until there is no non-integer solution in the updated near-optimal solution $\tilde{\mathbf{M}}_{opt}$. The corresponding objective function value Q_{min} can be obtained by substituting the near-optimal solution $\tilde{\mathbf{M}}_{opt}$ and \mathbf{S}_{opt} into the objective function of the optimization problem (12).

Generally speaking, to obtain the near-optimal integer solution of the considered optimization problem formulated in Eq. (12), the maximum iteration to execute 0-1 branch and bound is $K \cdot n \cdot F$, which is actually the number of ele-

ments in the three-dimensional integer variable $\tilde{\mathbf{M}}_{K \times n \times F} = \{\tilde{m}_{k,i,j}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$. Particularly, for each iteration of the 0-1 branch and bound, if there is more than one non-integer element, denoted as $N_n, N_n \geq 2$, in the updated solution $\tilde{\mathbf{M}}_{opt}$, 0-1 branch and bound should be executed for each non-integer solution. Then, $2 \cdot N_n$ branches are generated, which are compared to find the minimization branch and its corresponding solution $\tilde{\mathbf{M}}_{min}$ and \mathbf{S}_{min} to update the near-optimal solution as $\tilde{\mathbf{M}}_{opt} = \tilde{\mathbf{M}}_{min}$, $\mathbf{S}_{opt} = \mathbf{S}_{min}$.

The solving procedure of our proposed near-optimal algorithm is summarized in Algorithm 1 named Near-Optimal Algorithm Based on Relaxation and Boolean bound.

VI. SUBOPTIMAL SOLUTION BASED ON GREEDY METHOD

Considering that the near-optimal algorithm proposed is time-consuming, we propose a greedy algorithm with low computational complexity to obtain the suboptimal solution of original non-linear mixed 0-1 integer programming problem. The main idea of the greedy algorithm is to optimize variables $\tilde{\mathbf{M}} = \{\tilde{m}_{k,i,j}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ and $\mathbf{S} = \{s_{i,j}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ separately. The whole procedure is presented in the following two subsections.

A. Caching Strategy Optimization with Encoding Parameters Fixed

First, we optimize variable $\tilde{\mathbf{M}}_{K \times n \times F} = \{\tilde{m}_{k,i,j}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ with the encoding parameters fixed. The solution of variable $\tilde{\mathbf{M}}_{K \times n \times F} = \{\tilde{m}_{k,i,j}, \forall k \in \mathcal{K}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ can be obtained by solving the optimization problem (7) under the condition of giving the variable $\mathbf{S} = \{s_{i,j}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ a fixed value according to a given encoding rate. Actually, the solving process is the special case of Algorithm 1 with a fixed known \mathbf{S} , which greatly reduces the complexity of solving the optimization problem compared with a variable \mathbf{S} since the dimension of the optimization variable reduces.

According to Algorithm 1, with a fixed known \mathbf{S} , the optimization problem (12), (13) and (14) should be changed accordingly. Specifically, the minimization relaxing problem (12) can be reformulated as

$$\min_{\tilde{\mathbf{M}}} \sum_{j=1}^F g(\tilde{m}_{k,i,j}) \quad (15a)$$

$$\text{s.t. (7b), (7c), (7d), (11c)} \quad (15b)$$

Based on the reformulated minimization relaxing problem (15), the sub-relaxing problem one (13) and sub-relaxing problem two (14) can be reformulated as

Sub-relaxing problem one:

$$\min_{\tilde{\mathbf{M}}} \sum_{j=1}^F g(\tilde{m}_{k,i,j}) \quad (16a)$$

$$\text{s.t. (7b), (7c), (7d), (13c)} \quad (16b)$$

Sub-relaxing problem two:

$$\min_{\tilde{\mathbf{M}}} \sum_{j=1}^F g(\tilde{m}_{k,i,j}) \quad (17a)$$

$$\text{s.t. (7b), (7c), (7d), (14c)} \quad (17b)$$

Algorithm 1 Near-Optimal Algorithm Based on Relaxation and Boolean bound

Input: $myfun1$: the objective function of optimization problem (12); $myfun2$: non-linear constraints of optimization problem (12); \tilde{M}_0 , S_0 : initial solution of \tilde{M} and S .

Output: \tilde{M}_{opt} , S_{opt} and Q_{min} .

Initialize: Obtain \tilde{M}_r , S_r and the objective function value Q_r by solving relaxing problem (12). Calculate the initial total number of non-integer elements in \tilde{M}_r , denoted as N_n , and obtain the position index N_{in} . Initialize the left coefficient matrix of equality constraints (13c) in Eq. (13) and (14c) in Eq. (14) as $A_{eq} = 0$, and the right coefficient vector of equality constraints as $b_{eq} = 0$.

if $N_n = 0$ **then**

$\tilde{M}_{opt} = \tilde{M}_r$;
 $S_{opt} = S_r$;
 $Q_{min} = Q_r$;

else

Initialize $\tilde{M}_{opt} = \tilde{M}_r$. Set the number of non-integer solution in \tilde{M}_{opt} for each iteration of 0-1 branch and bound as $N_b = N_n$ and the corresponding position index vector $N_i = N_{in}$.

for $p = 1$ **to** N_n **do**

for $q = 1$ **to** N_b **do**

Update $A_{eq}(p, N_i(:, q)) = 1$;
 Obtain the left coefficient matrix of equality constraints for 0 and 1 branch $A_{eq_zero} = A_{eq}(1 : p, :)$, $A_{eq_one} = A_{eq}(1 : p, :)$;
 Update $b_{eq}(p, :) = 0$;
 Obtain the right coefficient vector of equality constraints for 0 branch $b_{eq_zero} = b_{eq}(1 : p, :)$;
 Update $b_{eq}(p, :) = 1$;
 Obtain the right coefficient vector of equality constraints $b_{eq_one} = b_{eq}(1 : p, :)$;
 Solving the sub-relaxing problem one (13) and the sub-relaxing problem two (14) by utilizing the global optimal algorithm GlobalSearch.

end

Compare the objective function values of all 0-1 branches of N_b non-integer variables, and find the minimization branch and obtain its corresponding solution, denoted as \tilde{M}_{min} and S_{min} ;

Update \tilde{M}_{opt} by setting $\tilde{M}_{opt} = \tilde{M}_{min}$, $S_{opt} = S_{min}$;
 Update N_b and N_i based on \tilde{M}_{opt} .

end

Obtain \tilde{M}_{opt} , S_{opt} and the objective function value Q_{min} by substituting \tilde{M}_{opt} and S_{opt} into the objective function $myfun1$ of optimization problem (12).

end

B. Encoding Parameters Optimization Based on the Greedy Method

Then, based on the caching strategy \tilde{M}_g obtained, the solution of variable $S = \{\tilde{s}_{i,j}, \forall i \in \mathcal{N}, \forall j \in \mathcal{F}\}$ can be obtained as S_g by gradually increasing the encoding rate for each video file one by one in a certain reasonable step. In this way, we can find the one which brings the maximum MOS value of

Algorithm 2 Suboptimal Algorithm Based on Greedy Method

Input: Caching strategy \tilde{M}_g , which is obtained by utilizing Algorithm 1 with the known S based on the given encoding rate which satisfies constraint (7d) in optimization problem (12).

Output: \tilde{M}_g , S_g and Q_{min} .

Initialize: initialize S_g according to the minimum encoding rates; initialize Q_{min} as $Q_{min} = Q_g$ which is calculated based on the obtained \tilde{M}_g and S_g ; initialize increasing step of encoding rate as $\frac{1}{T_d \cdot n}$; initialize the searching times N_1 and N_2 as $N_1 = (R_{max} - R_{min}) \cdot T_d \cdot N$, $N_2 = F$, where N is the number of video files whose encoding rate lies between R_{min} and R_{max} .

for $p = 1$ **to** N_1 **do**

for $q = 1$ **to** N_2 **do**

Obtain all possible S_g which satisfy constraints in optimization problem (12) according to the increasing step $\frac{1}{T_d \cdot n}$.

Obtain all possible Q_g by substituting all possible S_g , \tilde{M}_g into the objective function of optimization problem (12).

end

Compare all possible Q_g and find the one which brings the most increments to Q_{min} , denoted as Q_{gmin} and its corresponding encoding parameters, denoted as S_{gmin} .

Update S_g by setting $S_g = S_{gmin}$;

Update Q_{min} by setting $Q_{min} = Q_{gmin}$.

end

the current iteration until not meeting the constraints in the optimization problem (12).

Finally, the suboptimal solution of the original non-linear mixed non-integer solution can be obtained as \tilde{M}_g and S_g . The corresponding objective function value Q_{min} can be obtained by substituting the suboptimal solution \tilde{M}_g and S_g into the objective function of the optimization problem (12). The solving procedure for the greedy algorithm is summarized in Algorithm 2 named Suboptimal Algorithm Based on Greedy Method.

VII. SIMULATION RESULTS

In this section, we provide simulation results to highlight the performance of our algorithms. We consider a VoD downlink scenario in the cloud-edge network where the total number of video files is F , the total encoding packet of each video file is n , and the number of the EC servers is K . The total number of video requests received by SBSs during the whole delivery phase is ψ . The capacity of EC server is $\Phi_k, \forall k \in K$. To reveal the performance of the proposed algorithms in scenarios with different configuration, the parameters F, n, K, ψ, Φ_k can be adjusted in different figures. The tilt factor of the Zipf distribution is assumed as $\theta = 0.8$ referring to relevant work [49], [50]. The duration of each video file is assumed as $T_d = 2400s$ because the playback time of a video is generally 40min to 60min, here we choose 40min. The weighting coefficient of transmission latency is $V_j = 0.99, \forall j \in \mathcal{F}$ based on the MOS model in [47]. The transmission rate of the backhaul links is $R_{bk} = 10\text{Gbps}/F$ according to the 3GPP

specification [51]. To verify that encoding rates can be adjusted adaptively according to different caching capacity and security constraints, files of different encoding rates are considered. In particular, we consider three types of video files, the encoding rates of which satisfy different encoding rate ranges:

Type 1: The encoding rates of the 1st to the $(2/F)$ -th video files vary from 0.3 Mbps to 0.7 Mbps;

Type 2: The encoding rates of the $(2/F+1)$ -th to the $(2/F+4/F)$ -th video files vary from 1Mbps to 4 Mbps;

Type 3: The encoding rates of the $(2/F+4/F+1)$ -th to the F -th video files vary from 4 Mbps to 8 Mbps.

The detailed parameter settings are presented in Table I. Our proposed schemes and three existing algorithms are presented and analyzed as follows:

- *EC-VE (Edge Caching and Video Encoding)*: Our proposed near-optimal edge caching and video encoding based algorithm that jointly optimizes video encoding parameters and edge caching strategy.
- *Greedy EC-VE (Greedy Edge Caching and Video Encoding)*: Our proposed greedy edge caching and video encoding based algorithm with lower complexity, which optimizes video encoding parameters and edge caching strategy step by step.
- *ECST (Edge Caching Secure Transmission)* in [42]: The edge caching based secure transmission scheme aims to minimize the backhaul links rate under secrecy constraints by utilizing edge caching strategy for general data transmission.
- *EC-BF (Edge Caching and Beamforming)* in [41]: The secure transmission scheme aims to minimize delay in an edge cache-assisted millimeter-wave cloud radio access network by designing transmit and receive beamforming.
- *EC-TM (Edge Caching and Trust Mechanism)* in [38]: The secure caching scheme is based on a trust mechanism and a Chinese remainder theorem-based privacy preservation protocol in heterogeneous networks for multi-homing users.

Specifically, we validate the algorithm performance in terms of MOS, average MOS, transmission latency and the minimum caching capacity to ensure secure transmission with different simulation configurations. In order to show the performance difference under different video encoding rates and demonstrate the important impact of video encoding on system performance, three different fixed encoding rates are set for video files in the ECST scheme, EC-BF scheme and EC-TM scheme. In addition, ablation study about video encoding module and edge caching module is given in subsection VII-G to show the functions of the two modules and reveal their different effects on system performance. Finally, the computational complexity of the EC-VE algorithm, greedy EC-VE algorithm, the ECST algorithm, EC-BF algorithm and EC-TM algorithm is analyzed in subsection VII-H.

A. MOS of Each Video File

Fig. 3. (a) shows the MOS value of each video file with $n = 20$, $F = 8$, $K = 1$ and $\Phi_k = 26880\text{Mb}$, $\forall k \in \mathcal{K}$. The three fixed encoding rates of video files in the

TABLE I
VIDEO FILE PARAMETERS

Types	Encoding rate range	Video encoding quality parameters
Type 1	0.3Mbps-0.7Mbps	$C_1 = 0.23$, $C_2 = -1.5$, $C_3 = 3.3$, $C_4 = 2.5$
Type 2	1Mbps-4Mbps	$C_1 = 0.0426$, $C_2 = -0.4466$, $C_3 = 1.6369$, $C_4 = 1.8415$
Type 3	4Mbps-8Mbps	$C_1 = 0.0027$, $C_2 = -0.0669$, $C_3 = 0.5842$, $C_4 = 2.5248$

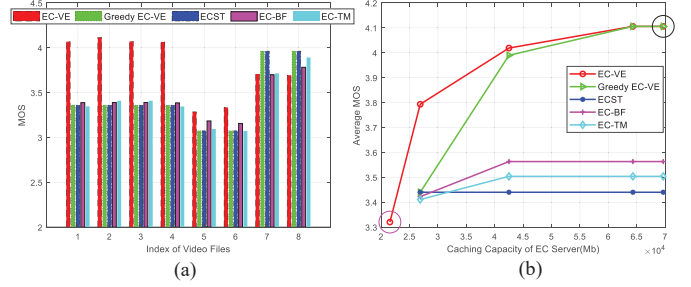


Fig. 3. (a) MOS of each video file. (b) Average MOS vs. different caching capacity of EC server.

ECST scheme, EC-BF scheme and EC-TM scheme are R_{\min} , $R_{\min} + [(R_{\max} - R_{\min})/10]$ and $R_{\min} + [(R_{\max} - R_{\min})/20]$, respectively. Fig. 3. (a) shows that, for each video file, the MOS performance of EC-VE algorithm is not always superior to the MOS performance of other schemes in spite of the obvious advantage in the average sense. The average MOS values of the EC-VE algorithm, Greedy EC-VE algorithm, ECST scheme, EC-BF scheme and EC-TM scheme are 3.7935, 3.4402 and 3.4402, 3.4225 and 3.4110, respectively. Our optimization goal is the whole performance advantage of all video files rather than the performance advantage for a single video file. Thus, there is no absolute superiority of MOS value for each video file in our proposed algorithm. Specifically, to obtain the whole performance advantage of the MOS value, the 1-st to 6-th video files, as depicted in Fig. 3. (a), which can get higher QoE with lower encoding rate, are more likely to be distributed with caching capacity to improve QoE on the condition of limited caching capacity, while the 7-th and 8-th video files, which request higher encoding rate, are distributed with less caching capacity.

Each video file has own caching strategy and encoding parameter. The number and the size of encoding packets cached on the EC server vary for different video files. Therefore, the video encoding quality and transmission latency of each video file are different for the five schemes. This results in different MOS values in the case of the same caching capacity of the EC server. Furthermore, the MOS values of the Greedy EC-VE algorithm and ECST scheme are the same for each video file. This is because each video file of both schemes is encoded into the minimum encoding rate. Meanwhile, the caching strategy for each video file is the same when the caching capacity is $\Phi_k = 26880\text{Mb}$, $\forall k \in \mathcal{K}$. This value of caching capacity allows all encoding packets of each video file to be cached on the EC server at the minimum encoding rate. Besides, the MOS values of the EC-BF scheme and the EC-TM scheme are lower than that of the other schemes. Because for the EC-BF scheme and EC-TM scheme, the secure transmission rate for

each video file is lower than that of other schemes. Moreover, for the EC-TM scheme, the caching capacity for improving MOS is more scarce compared with other schemes due to the fact that the encrypted video files are larger and occupy more caching capacity.

B. Average MOS with Different Caching Capacity of EC Server

In Fig. 3. (b), the average MOS value versus different caching capacity of EC server with $n = 20$, $F = 8$, $K = 1$ is presented. The three fixed encoding rates of video files in the ECST scheme, EC-BF scheme and EC-TM scheme are R_{\min} , $R_{\min} + [(R_{\max} - R_{\min})/10]$ and $R_{\min} + [(R_{\max} - R_{\min})/20]$, respectively. As shown in Fig. 3. (b), the average MOS values of the EC-VE algorithm and the Greedy EC-VE algorithm increase with the increasing caching capacity of the EC server. Both of them eventually tend to the same maximum value, which is determined by the maximum encoding rate of each video file and the optimal caching strategy. At this time, the caching capacity of EC server is sufficient to cache all encoding packets. The EC-VE algorithm also achieves a considerable performance gain compared with the Greedy EC-VE algorithm. The minimum average MOS value of the proposed EC-VE algorithm is obtained when the caching capacity of the EC server is just enough to cache the minimum number of encoding packets to satisfy the security constraints. However, the Greedy EC-VE algorithm cannot find the feasible solution that satisfies the security constraints at this point, marked by the purple circle in Fig. 3. (b). As the caching capacity of the EC server increases, the gap of the average MOS values is initially large, then gradually narrows. This is because large caching capacity can provide more choices of caching strategy and encoding rate to bring better performance on video encoding quality and transmission latency.

Furthermore, the average MOS value of the ECST scheme, which is always worse than the average MOS value of the proposed EC-VE algorithm and Greedy EC-VE algorithm, remains unchanged with the increasing caching capacity. This scenario is reasonable since the minimum encoding rate is adopted in the ECST scheme. At this time, the caching strategy is the same regardless of the caching capacity because its aim is to minimize the backhaul links rate. Thus, the video encoding packets cached on the EC server will be as many as possible if the caching capacity is large enough and all encoding packets are cached on the EC server when each video file is encoded into its minimum encoding rate.

Besides, the average MOS values of the EC-BF scheme and the EC-TM scheme, which are much worse than that of the proposed EC-VE algorithm and Greedy EC-VE algorithm, increase with the increasing caching capacity initially and gradually remain stable. In the beginning, less encoding packets are cached on EC servers to ensure security due to the limited caching capacity. With the increasing caching capacity, all encoding packets can be cached on EC servers to ensure security and reduce transmission latency at the same time. However, the MOS values of the EC-BF scheme and EC-TM scheme keep stable when the caching capacity continues to increase as a result of the fixed encoding rates. The gap

TABLE II
MOS OF EACH VIDEO FILE UNDER DIFFERENT CACHING CAPACITY

MOS of Each Video File under the Smallest Caching Capacity								
Index of Video File	1	2	3	4	5	6	7	8
Encoding Rate (Mbps)	0.3	0.3	0.3	0.3	1	1	4	4
Number of Cached Packets	15	19	17	14	18	18	16	15
MOS	3.302	3.303	3.303	3.300	2.984	2.966	2.708	2.695
MOS of Each Video File under the Largest Caching Capacity								
Index of Video File	1	2	3	4	5	6	7	8
Encoding Rate (Mbps)	0.7	0.7	0.7	0.7	4	4	8	8
Number of Cached Packets	20	20	20	20	20	20	20	20
MOS	4.116	4.116	4.116	4.116	3.932	3.932	4.261	4.261

of the EC-BF scheme and the EC-TM scheme results from different encoding rate and the effective utilization rate of caching capacity.

The Greedy EC-VE algorithm, the ECST scheme, the EC-BF scheme and the EC-TM scheme cannot achieve secure transmission when the caching capacity value is very small, i.e., at the first point marked by the purple circle in Fig. 3. (b) with caching capacity $\Phi_k = 21540\text{Mb}, \forall k \in \mathcal{K}$, while the proposed EC-VE algorithm can ensure secure transmission and obtain a feasible average MOS value by adjusting caching strategy and encoding parameters at the same time. The observations above demonstrate the benefits of the proposed algorithm, which comprehensively considers the interaction of caching strategy and encoding parameters, and adjusts both of them simultaneously to adapt to different caching capacity.

To show the changes of encoding parameters and caching strategy under different caching capacity, encoding rates and the number of cached packets for two selected points in Fig. 3. (b) are given in Table II. At the first point marked by the purple circle in Fig. 3. (b), the minimum average MOS value of the proposed EC-VE algorithm is obtained. At the last point marked by the black circle in Fig. 3. (b), the same maximum average MOS values of the proposed EC-VE algorithm and Greedy EC-VE algorithm are obtained. Table II gives the MOS value of each video file at the first point when the EC server has the smallest caching capacity $\Phi_k = 21540\text{Mb}, \forall k \in \mathcal{K}$ and at the last point when the EC server has the largest caching capacity $\Phi_k = 69660\text{Mb}, \forall k \in \mathcal{K}$. The minimum MOS of each video file is obtained at the first point. In this situation, all video files in the EC-VE algorithm are encoded into the minimum encoding rates, and the minimum number of encoding packets are cached on the EC server to satisfy the security constraints due to the limited caching capacity. Meanwhile, the minimum number of cached packets varies for different video files, because different video files have a different number of requests. At the last point, the maximum MOS of each video file is obtained, because all video files of both algorithms are encoded into the maximum encoding rates and all encoding packets are cached on the EC server due to the sufficient caching capacity.

C. Average MOS with Different Parameters of Video Files

Fig. 4. (a) shows the average MOS value versus different number of video files with $n = 10$, $K = 1$ and $\Phi_k =$

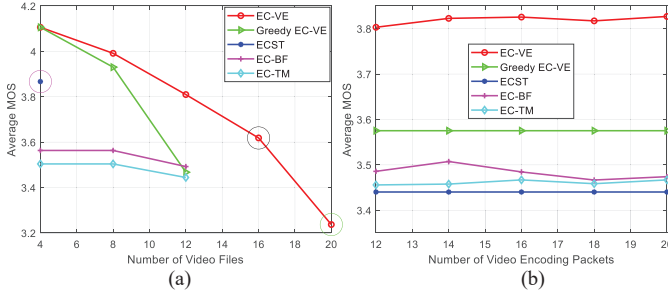


Fig. 4. (a) Average MOS vs. different number of video files. (b) Average MOS vs. different number of video encoding packets.

39000Mb, $\forall k \in \mathcal{K}$. The three fixed encoding rates of video files in the ECST scheme, EC-BF scheme and EC-TM scheme are $R_{\min} + [(R_{\max} - R_{\min})/2]$, $R_{\min} + [(R_{\max} - R_{\min})/10]$ and $R_{\min} + [(R_{\max} - R_{\min})/20]$, respectively. From Fig. 4. (a), the average MOS value decreases with the increasing number of video files for the proposed EC-VE algorithm and the Greedy EC-VE algorithm. The gap between them increases with the increasing video files when the number of video files is less than or equal to $F = 12$. However, the Greedy EC-VE algorithm has no solution as the number of video files continues to increase to $F = 16$ and $F = 20$, marked by the black and cyan circles in Fig. 4. (a), while the EC-VE algorithm still has the feasible and effective solution. Specifically, the minimum average MOS value is obtained at the last point marked by the cyan circle. At this time, all video files are encoded into the minimum encoding rates, and the minimum number of encoding packets are cached on the EC server. Besides, the average MOS values of the EC-BF scheme and the EC-TM scheme, which are worse than our proposed schemes, keep stable when the number of video files is less than or equal to $F = 8$. Then, the MOS values of them decrease when the video files increase to $F = 12$.

The reasons contributing to these observations are that when there are four video files, all of them are encoded into the maximum encoding rates in the proposed near-optimal and greedy algorithm, and encoded into the fixed encoding rates in the EC-BF scheme and the EC-TM scheme. Meanwhile, all encoding packets are cached on the EC server due to the sufficient caching capacity. This brings the best MOS values for each scheme as shown by the first points. As the number of video files increases, only a part of the encoding packets can be cached on the EC server due to the limited caching capacity. This cannot allow all video files to be encoded into the maximum encoding rates. In this situation, the caching strategy and encoding parameters of the proposed two algorithms are different. For the EC-VE algorithm, the near-optimal average MOS value can be obtained by jointly optimizing the caching strategy and encoding parameters flexibly, while the Greedy EC-VE algorithm achieves worse MOS values or even an infeasible solution. The reason is that it optimizes the caching strategy and encoding parameters step by step. Thus, it cannot satisfy the complex and strict security constraints when there are many video files. For the EC-BF scheme and the EC-TM scheme, they cannot ensure security when the video files increase to $F = 12$ because of the fixed encoding

rates and limited caching capacity. This phenomenon further demonstrates the performance advantage of the near-optimal EC-VE algorithm.

Furthermore, it is interesting to find that there is only one feasible point marked by the purple circle in Fig. 4. (a) for the ECST scheme when there are four video files in the considered scenario, and the performance of its average MOS value is much worse than the proposed algorithms. This is because the ECST scheme cannot satisfy the security constraints by adjusting its caching strategy when there are many video files and the caching capacity is limited. Meanwhile, the simple caching strategy of the ECST scheme demands that the number of encoding packets cached in different EC servers must be the same for the same video file.

In Fig. 4. (b), the average MOS value versus different number of video encoding packets with $F = 8$, $K = 1$ and $\Phi_k = 28000\text{Mb}$, $\forall k \in \mathcal{K}$ is presented. The three fixed encoding rates of video files in the ECST scheme, EC-BF scheme and EC-TM scheme are R_{\min} , $R_{\min} + [(R_{\max} - R_{\min})/10]$ and $R_{\min} + [(R_{\max} - R_{\min})/20]$, respectively. As shown in Fig. 4. (b), the average MOS values of the proposed EC-VE algorithm, Greedy EC-VE algorithm, the ECST scheme, EC-BF scheme and EC-TM scheme are almost stable with the increasing number of encoding packets. The MOS performance of the proposed near-optimal EC-VE algorithm fluctuates slightly when the number of encoding packets changes. Meanwhile, it is significantly superior to the Greedy EC-VE algorithm, the ECST scheme, the EC-BF scheme and EC-TM scheme. Specifically, the MOS performance increases slowly at first, then decreases when the encoding packets increase to $n = 18$, and finally increases to the maximum value when the number of encoding packets is $n = 20$. The fluctuations result from the caching strategy and the encoding parameters which vary slightly with the changing total number of encoding packets, when the caching capacity and the number of video files remain unchanged. A tradeoff exists between the total number of encoding packets and the size of each encoding packet, which results in the falling fluctuation when the number of encoding packets is $n = 18$. In this situation, the proposed near-optimal EC-VE algorithm will wisely consider this balance and achieve the near-optimal results by joint adjustment of caching strategy and encoding parameters. This also brings the reasonable fluctuations.

It can also be observed that for the EC-BF scheme and EC-TM scheme, there are also fluctuations, especially for the EC-BF scheme. This is because the caching strategy and secure transmission rate adjusts within the available caching capacity when the number of encoding packets changes, which brings difference of transmission latency. As for the Greedy EC-VE algorithm, its performance of the average MOS value is still better than that of the comparable ECST scheme, while the average MOS values of both schemes almost remain stable when the number of encoding packets changes. This is because both of them optimize the caching strategy with the fixed encoding parameters, which results in the same caching strategy when the caching capacity of the EC server and the number of video files keep unchanged in spite of the changing number of total encoding packets.

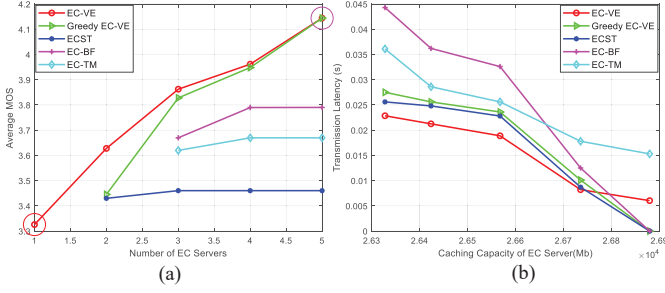


Fig. 5. (a) Average MOS vs. different number of EC servers. (b) Transmission latency vs. different caching capacity of EC server.

D. Average MOS with Different Number of EC Servers

Fig. 5. (a) shows the average MOS value versus different number of EC servers with $n = 10$, $F = 12$. The three fixed encoding rates of video files in the ECST scheme, EC-BF scheme and EC-TM scheme are $R_{\min} + [(R_{\max} - R_{\min})/10]$, $R_{\min} + [(R_{\max} - R_{\min})/2]$ and $R_{\min} + [(R_{\max} - R_{\min})/5]$, respectively. Considering more general scenarios, EC servers are assumed to have different caching capacity. To be specific, when the number of EC servers is $K > 1$, the caching capacity values of EC servers are [21271 18272]Mb, [21271 18272 13017]Mb, [21271 18272 13017 22000]Mb, [21271 18272 13017 22000 22224]Mb, respectively. As shown in Fig. 5. (a), the average MOS values of the EC-VE algorithm and Greedy EC-VE algorithm increase with the increasing number of EC servers. They reach to the same maximum value when the total caching capacity of all EC servers is sufficient. In this case, encoding packets of all video files are cached on EC servers in the maximum encoding rates, just as the last point shows marked by purple circle in Fig. 5. (a). The performance of the EC-VE scheme is significantly superior to other schemes.

Besides, the average MOS values of the ECST scheme, the EC-BF scheme and the EC-TM scheme initially rise with the increasing number of EC servers, then remain stable. The reason is that less encoding packets are cached on EC servers due to the limited total caching capacity of EC servers. As the number of EC servers increases, the provided caching capacity is sufficient to cache more encoding packets on EC servers until all encoding packets are cached on EC servers in the fixed encoding rates. This leads to the stable average MOS of the three schemes when the number of EC servers increase to four. It is also interesting to find that there exist infeasible solutions for each scheme except for the EC-VE scheme. This is because the joint optimization of video encoding and caching strategy can wisely utilize the limited caching capacity provided by one EC server to improve QoE and ensure security by flexibly adjusting encoding parameters and caching strategy while other schemes cannot, just as the first point shows marked by red circle in Fig. 5. (a).

E. Transmission Latency with Different Caching Capacity of EC Server

In Fig. 5. (b), the transmission latency versus different caching capacity of EC server with $n = 20$, $F = 8$, $K = 1$ is presented. To solely show the performance of

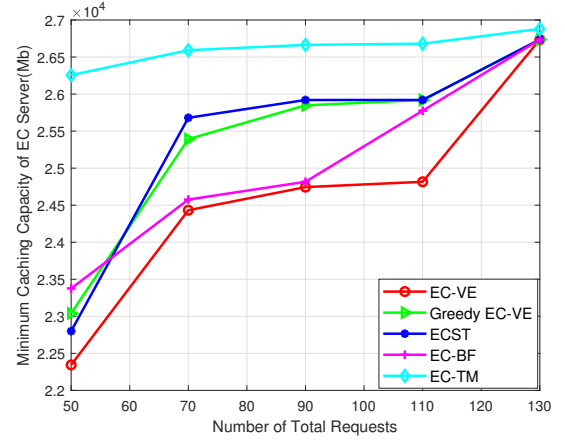


Fig. 6. Minimum caching capacity vs. different number of total requests.

transmission latency, the fixed encoding rates of video files in the ECST scheme, EC-BF scheme and EC-TM scheme are set as the same encoding rate, i.e., R_{\min} . As shown in Fig. 5. (b), the transmission latency of the EC-VE scheme, Greedy EC-VE scheme, ECST scheme, EC-BF scheme and EC-TM scheme decreases with the increasing caching capacity of EC server. The transmission latency of the Greedy EC-VE scheme, ECST scheme and EC-TM scheme decreases to zero when the caching capacity is $\Phi_k = 26880\text{Mb}$, $\forall k \in \mathcal{K}$ because all encoding packets can be cached on EC server at the fixed minimum encoding rates due to the sufficient caching capacity. In general, the transmission latency of the EC-VE scheme is lower than the ECST scheme, which is lower than other schemes. This phenomenon strongly proves the performance advantage of the proposed EC-VE scheme in reducing transmission latency. It can find the tradeoff between video encoding quality and transmission latency by adjusting the cached encoding packets sizes when caching capacity is limited, which leads to its non-zero value at the last point in Fig. 5. (b). Besides, the ECST scheme outperforms the Greedy EC-VE scheme. This is because the goal of the ECST scheme is to minimize the backhaul links rate. It aims to cache as many encoding packets as possible on EC servers with available caching capacity. Besides, the transmission latency of the EC-TM scheme does not decrease to zero because more caching capacity is needed to cache the encrypted video files.

F. Minimum Caching Capacity with Different Number of Requests of Users

Fig. 6 shows the minimum caching capacity of the EC server versus different number of total requests with $n = 10$, $F = 8$, $K = 1$. Because the minimum caching capacity is required, the fixed encoding rates of video files in the ECST scheme, EC-BF scheme and EC-TM scheme are set as the minimum encoding rate, i.e., R_{\min} . The minimum caching capacity of the five schemes increases with the increasing number of total requests and eventually tends to the same value when the number of total requests reaches $\psi = 130$. This is because more caching capacity is needed to cache more video encoding packets to ensure the requirements of video encoding quality, transmission latency and security constraints

as the number of total requests increases. Furthermore, Fig. 6 shows that the minimum caching capacity of the proposed EC-VE algorithm is much smaller than that of the Greedy EC-VE algorithm, the ECST scheme and the EC-TM scheme. This phenomenon shows that the proposed EC-VE algorithm still has relatively good performance even with limited caching resources. Meanwhile, it can spare more caching resources to achieve better video encoding quality and low latency when satisfying the security constraints.

It is interesting to find that the gap of the minimum caching capacity between EC-VE algorithm and the EC-BF scheme is relatively small when the number of total requests is $\psi = 70$ and $\psi = 90$. This reflects the advantage of the rate adaptive scheme EC-BF compared with other schemes. Meanwhile, the minimum caching capacity of the EC-TM scheme is much larger than other schemes because encrypted video files require more caching capacity under the same conditions. The minimum caching capacity of other four schemes except EC-TM scheme finally reaches the same value as the number of total requests increases to $\psi = 130$, because more requests mean that more video encoding packets should be cached on the EC server for any scheme. As a result, almost all video encoding packets are cached on the EC server at the minimum encoding rates to satisfy the security constraints and guarantee the video encoding quality and latency requirements when the number of total requests reaches $\psi = 130$.

G. Ablation Study about Video Encoding and Edge Caching

In this subsection, the impact of video encoding module and edge caching module on the performance of MOS, average MOS and the minimum caching capacity to ensure secure transmission is analyzed. For comparison, in addition to our proposed near-optimal EC-VE algorithm, the other two algorithms which are solely based on video encoding or edge caching are considered and presented:

- *EC (Edge Caching)*: Edge caching based algorithm which solely utilizes edge caching to improve video encoding quality and reduce latency under secure transmission for the backhaul links.
- *VE (Video Encoding)*: Video encoding based algorithm which solely utilizes video encoding to improve video encoding quality and reduce latency under secure transmission for the backhaul links.

Fig. 7 shows the MOS value of each video file with $F = 8$, $n = 18$, $K = 1$ and $\Phi_k = 28000\text{Mb}, \forall k \in \mathcal{K}$. From Fig. 7, for each video file, the MOS values of the EC-VE algorithm, EC algorithm and VE algorithm are different. The average MOS values of them are 3.8172, 3.4075 and 3.5371, respectively. Although the EC-VE algorithm outperforms the other two in the average sense, for each video file, it is not always better. This is because the EC-VE scheme can obtain the whole performance superiority by jointly adjusting edge caching strategy and video encoding parameters, while the EC scheme and VE scheme can improve MOS for certain files by only utilizing edge caching or video encoding strategy. As shown in Fig. 7, the 1-st to 6-th video files are more likely to be allocated more caching capacity to achieve higher QoE

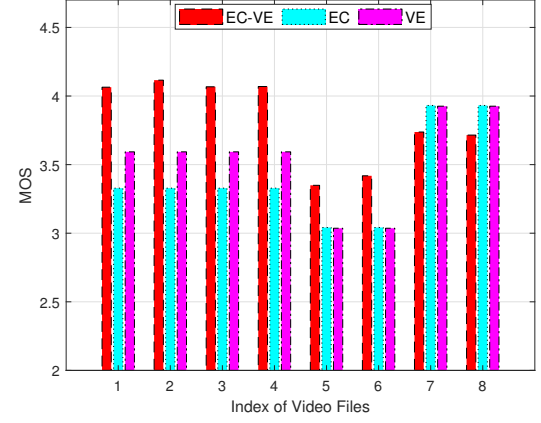


Fig. 7. MOS of each video file.

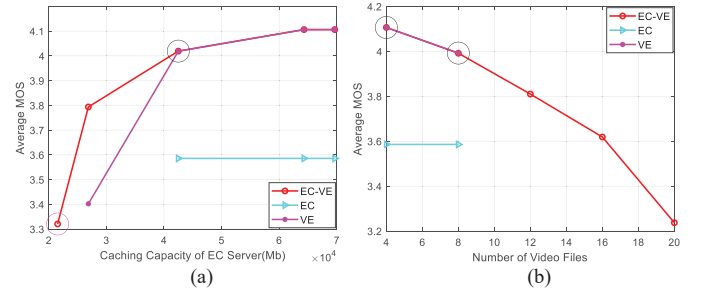


Fig. 8. (a) Average MOS vs. different caching capacity of EC server. (b) Average MOS vs. different number of video files.

with relatively low encoding rate, while the 7-th and 8-th video files are allocated less caching capacity. This is because the QoE requirements of the 7-th and 8-th video files with high encoding rates demand more caching capacity. Furthermore, the MOS performance of the VE scheme performs better than that of the EC scheme. This demonstrates the advantage of video encoding strategy with limited caching capacity.

Fig. 8. (a) shows the average MOS versus different caching capacity of EC server with $F = 8$, $n = 20$ and $K = 1$. As shown in Fig. 8. (a), the average MOS values of the EC-VE scheme and VE scheme rise with increasing caching capacity, while the average MOS value of the EC scheme keeps stable when caching capacity increases. This is because for the EC scheme, all packets are cached on EC server with the fixed minimum encoding rates which cannot be adjusted when the caching capacity changes, while the EC-VE scheme and VE scheme can obtain higher QoE by encoding video files into higher encoding rates when more caching capacity are provided. Furthermore, it is interesting to find that the VE scheme and the EC scheme cannot achieve secure transmission when caching capacity is very low as marked by the purple circle in Fig. 8. (a). The MOS values of the EC-VE scheme and the VE scheme are the same at the point marked by the black circle in Fig. 8. (a) when caching capacity increases. This is because all encoding packets of video files are cached on EC server when caching capacity is enough. For the VE scheme, all video files are assumed to be cached on the EC server to ensure secure transmission because caching strategy

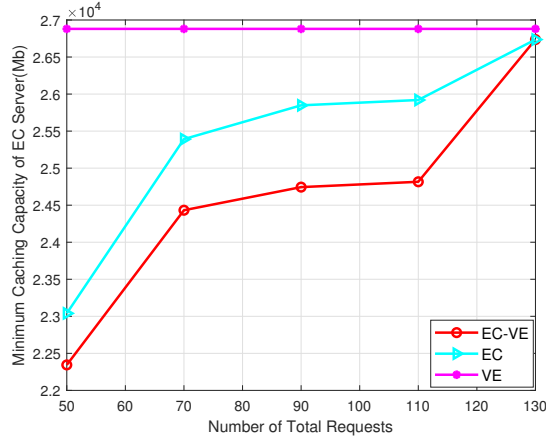


Fig. 9. Minimum caching capacity vs. different number of total requests.

cannot be adjusted. The MOS superiority of the VE scheme to the EC scheme also proves the advantage of encoding strategy compared with caching strategy.

Fig. 8. (b) shows the average MOS versus different number of video files with $n = 10$, $K = 1$ and $\Phi_k = 39000\text{Mb}$, $\forall k \in \mathcal{K}$. As marked by the black circles in Fig. 8. (b), when the number of video files is 4 and 8, the average MOS values of the EC-VE scheme and the VE scheme decrease with the increasing number of video files, while the average MOS of the EC scheme remains stable. The EC-VE scheme and the VE scheme have the same performance, which is significantly superior to that of the EC scheme. This is because the EC-VE scheme and the VE scheme achieve the highest average MOS values with the maximum encoding rates when the caching capacity is sufficient to cache all encoding packets on EC server. The EC scheme shows the constant average MOS value with all video files being encoded into the minimum encoding rates. When the number of video files is more than 8, only the EC-VE scheme can ensure secure transmission of all video files by caching less encoding packets of each video file and encoding each video file into lower encoding rate under limited caching capacity. The EC scheme and the VE scheme solely depending on caching strategy or encoding strategy cannot find feasible solutions when the number of video files exceeds 8.

Fig. 9 shows the minimum caching capacity of the EC server versus different number of total requests with $n = 10$, $F = 8$, $K = 1$. The minimum caching capacity of the EC-VE scheme and the EC scheme increases with the number of video total requests and reaches the same value at the last point. The EC-VE scheme requires much less caching capacity to ensure security than the EC scheme. Even if the minimum encoding rates are adopted by the EC scheme to obtain the minimum caching capacity to ensure security, the minimum caching capacity is still significantly more than that of the EC-VE scheme. This further demonstrates the advantage of joint optimization of caching and encoding strategy for EC-VE scheme. Meanwhile, the minimum caching capacity of the VE scheme keeps constant as the number of total requests increases because all encoding packets are cached on the EC server to ensure security. This demands at least caching capacity of $\Phi_k = 26880\text{Mb}$ with each video file being encoded

into the minimum encoding rate.

From the analysis above, we can know that the edge caching module is responsible for caching enough packets with as small caching capacity as possible to ensure secure transmission and sparing as large caching capacity as possible for the video encoding module to obtain high video encoding quality, while the video encoding module is responsible for encoding video files into as high encoding rates as possible under secure transmission. When the edge caching module acts alone and the encoding rates of video files cannot be adjusted, the system performance is poor even if the caching capacity is sufficient. Similarly, when the video encoding module acts alone and the caching strategy cannot be adjusted, the limited caching capacity is used to ensure security and there is no enough margin to improve video encoding quality. Therefore, it is necessary to jointly optimize edge caching module and video encoding module because they interact with each other and cooperatively act on QoE improvement and security guarantee.

H. Complexity Analysis

In this subsection, the computational complexity of our proposed EC-VE algorithm, Greedy EC-VE algorithm, the ECST scheme in [42], the EC-BF scheme in [41] and the EC-TM scheme in [38] is analyzed.

(1) EC-VE: For our proposed near-optimal EC-VE algorithm, the computational complexity is determined by the GlobalSearch algorithm and 0-1 branch and bound method as shown in Algorithm 1. For the GlobalSearch algorithm, the dimension of the optimization variable is $(K + 1) \cdot n \cdot F$. Then, the computational complexity of the GlobalSearch algorithm can be given by $O(h_g((K + 1) \cdot n \cdot F))$, where $h_g(\cdot)$ is the number of operations positively related to the dimension of the optimization variable. Finally, the computational complexity of the EC-VE algorithm is $O(h_g((K + 1) \cdot n \cdot F) + N_n \cdot N_b \cdot h_g((K + 1) \cdot n \cdot F))$.

(2) Greedy EC-VE: As shown in Algorithm 2, the computational complexity of our proposed Greedy EC-VE algorithm is determined by the Algorithm 1 with a fixed known \mathbf{S} and the optimization algorithm of encoding parameters based on the optimized caching strategy $\tilde{\mathbf{M}}_g$. For Algorithm 1 with a fixed known \mathbf{S} , since the dimension of optimization variable is $K \cdot n \cdot F$, then the computational complexity can be given by $O(h_g(K \cdot n \cdot F))$. Finally, the computational complexity of the Greedy EC-VE algorithm is $O(h_g(K \cdot n \cdot F) + N_n \cdot N_b \cdot h_g(K \cdot n \cdot F) + N_1 \cdot N_2)$.

(3) ECST and EC-TM: For the ECST scheme and EC-TM scheme, since they only aim to optimize caching strategy without considering video encoding, then the computational complexity is $O(h_g(K \cdot n \cdot F) + N_n \cdot N_b \cdot h_g(K \cdot n \cdot F))$.

(4) EC-BF: For the EC-BF scheme, since it aims to optimize caching strategy and secure transmission rate without considering video encoding, then the computational complexity is $O(h_g((K \cdot n + 1) \cdot F) + N_n \cdot N_b \cdot h_g((K \cdot n + 1) \cdot F))$.

Based on the analysis above, compared with the Greedy EC-VE algorithm, the increase in the dimension of the optimization variable in our proposed near-optimal EC-VE algorithm

brings higher computational complexity. It has a much greater impact on the running time than the increase in the number of iterations caused by optimizing the encoding parameters alone based on the greedy method. In our simulation, the running time of the near-optimal EC-VE algorithm is approximately 30 times the running time of the Greedy EC-VE algorithm. For the ECST in [42] and the EC-TM scheme in [38], the computational complexity is lower than that of our Greedy EC-VE algorithm because the video encoding is not considered. However, as mentioned above, the increase in the number of iterations for optimizing the encoding parameters has a trivial effect on the running time. Therefore, the difference of computational complexity among the Greedy EC-VE algorithm, the ECST scheme and the EC-TM scheme is very small and can be considered almost the same to some extent. For the EC-BF scheme in [41], the computational complexity is higher than that of Greedy EC-VE scheme, ECST scheme and EC-TM scheme due to the fact that it aims to optimize caching strategy and secure transmission rate simultaneously.

VIII. CONCLUSION

In this paper, we proposed a QoE-driven cross-layer optimization scheme for secure video transmission over the backhaul links in cloud-edge networks. First, we developed a secure transmission model based on video encoding and edge caching. By employing this model as the security constraint and considering the interaction of video encoding parameters and edge caching strategy, we formulated a joint optimization problem of video encoding parameters and edge caching strategy to improve QoE. Then, a near-optimal algorithm was designed to solve the joint optimization problem. Furthermore, we proposed a greedy algorithm with low complexity to obtain the suboptimal solution. Simulation results were presented to show that our proposed algorithms can greatly improve video encoding quality and reduce transmission latency under the condition of ensuring secure transmission of the backhaul links compared with the existing algorithms. In addition, our proposed algorithms were proven to be more robust for caching capacity and could ensure secure transmission for more videos with limited caching capacity of edge caching servers.

REFERENCES

- [1] X. Tian, C. Zhao, H. Liu, and J. Xu, "Video on-demand service via wireless broadcasting," *IEEE Trans. Mobile Comput.*, vol. 16, no. 10, pp. 2970-2982, Oct. 2017.
- [2] Y. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing: A key technology towards 5g," *ETSI White Paper*, vol. 11, no. 11, pp. 1-16, 2015.
- [3] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450-465, Feb. 2018.
- [4] R. Sun, Y. Wang, N. Cheng, L. Lyu, S. Zhang, H. Zhou, and X. Shen, "QoE-driven transmission-aware cache placement and cooperative beamforming design in cloud-RANs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 636-650, Jan. 2020.
- [5] Z. Qu, B. Ye, B. Tang, S. Guo, S. Lu, and W. Zhuang, "Cooperative caching for multiple bitrate videos in small cell edges," *IEEE Trans. Mobile Comput.*, vol. 19, no. 2, pp. 288-299, Feb. 2020.
- [6] X. Chen, L. He, S. Xu, S. Hu, Q. Li, and G. Liu, "Hit ratio driven mobile edge caching scheme for video on demandservices," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Shanghai, China, Jul. 2019, pp. 1702-1707.
- [7] L. Dong, Z. Han, A. P. Petropulu, and H. V. Poor, "Improving wireless physical layer security via cooperating relays," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1875-1888, Mar. 2010.
- [8] N. Yang, L. Wang, G. Geraci, M. Elkaslan, J. Yuan, and M. D. Renzo, "Safeguarding 5G wireless communication networks using physical layer security," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 20-27, Apr. 2015.
- [9] Y. Liu, H. Chen, and L. Wang, "Physical layer security for next generation wireless networks: Theory, technologies, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 347-376, 1st Quart., 2016.
- [10] J. Chen, L. Yang, and M. Alouini, "Physical layer security for cooperative NOMA systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4645-4649, May 2018.
- [11] J. Zhang, C. Kundu, O. A. Dobre, E. Garcia-Palacios, and N. Vo, "Secrecy performance of small-cell networks with transmitter selection and unreliable backhaul under spectrum sharing environment," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 10895-10908, Nov. 2019.
- [12] H. T. Nguyen, J. Zhang, N. Yang, T. Q. Duong, and W. Hwang, "Secure cooperative single carrier systems under unreliable backhaul and dense networks impact," *IEEE Access*, vol. 5, pp. 18310-18324, 2017.
- [13] C. Yin, H. T. Nguyen, C. Kundu, Z. Kaleem, E. Garcia-Palacios, and T. Q. Duong, "Secure energy harvesting relay networks with unreliable backhaul connections," *IEEE Access*, vol. 6, pp. 12074-12084, 2018.
- [14] T. Truong, M. Nguyen, C. Kundu, and L. D. Nguyen, "Secure cognitive radio networks with source selection and unreliable backhaul connections," *IET Commun.*, vol. 12, no. 15, pp. 1771-1777, Mar. 2018.
- [15] J. Ren, G. Yu, Y. He, and G. Y. Li, "Collaborative cloud and edge computing for latency minimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 5031-5044, May 2019.
- [16] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322-2358, 4th Quart., 2017.
- [17] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637-646, Oct. 2016.
- [18] J. Li, Y. Chen, Z. Lin, W. Chen, B. Vucetic, and L. Hanzo, "Distributed caching for data dissemination in the downlink of heterogeneous networks," *IEEE Trans. Commun.*, vol. 63, no. 10, pp. 3553-3568, Oct. 2015.
- [19] A. Sengupta, R. Tandon, and O. Simeone, "Cache aided wireless networks: Tradeoffs between storage and latency," in *Proc. Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2016, pp. 320-325.
- [20] H. Hsu and K. Chen, "A resource allocation perspective on caching to achieve low latency," *IEEE Commun. Lett.*, vol. 20, no. 1, pp. 145-148, Jan. 2016.
- [21] S. Miller, O. Atan, and M. V. D. Schaar, "Context-aware proactive content caching with service differentiation in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 1024-1036, Feb. 2017.
- [22] S. Li, J. Xu, M. V. D. Schaar, and W. Li, "Trend-aware video caching through online learning," *IEEE Trans. Multimedia*, vol. 18, no. 12, pp. 2503-2516, Dec. 2016.
- [23] W. Liu, J. Zhang, Z. Liang, L. Peng, and J. Cai, "Content popularity prediction and caching for ICN: A deep learning approach with SDN," *IEEE Access*, vol. 6, pp. 5075-5089, Dec. 2017.
- [24] T. Hou, G. Feng, S. Qin, and W. Jiang, "Proactive content caching by exploiting transfer learning for mobile edge computing," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, Singapore, Dec. 2017, pp. 1-6.
- [25] Z. Wang, L. Sun, C. Wu, and S. Yang, "Enhancing internet-scale video service deployment using microblog-based prediction," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 3, pp. 775-785, Mar. 2015.
- [26] Z. Zhang, M. Hua, C. Li, Y. Huang, and L. Yang, "Placement delivery array design via attention-based sequence-to-sequence model with deep neural network," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 372-375, Apr. 2019.
- [27] Y. M. Saputra, D. T. Hoang, D. N. Nguyen, E. Dutkiewicz, D. Niyato, and D. I. Kim, "Distributed deep learning at the edge: A novel proactive and cooperative caching framework for mobile edge networks," *IEEE Wireless Commun. Lett.*, vol. 8, no. 4, pp. 1220-1223, Aug. 2019.
- [28] L. Ale, N. Zhang, H. Wu, D. Chen, and T. Han, "Online proactive caching in mobile edge computing using bidirectional deep recurrent neural network," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5520-5530, Jun. 2019.
- [29] C. Li, L. Toni, J. Zou, H. Xiong, and P. Frossard, "QoE-driven mobile edge caching placement for adaptive video streaming," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 965-984, Apr. 2018.
- [30] X. Huang, L. He, X. Chen, G. Liu, and F. Li, "A more refined mobile edge cache replacement scheme for adaptive video streaming with mutual cooperation in multi-mec servers," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, London, United Kingdom, Jul. 2020, pp. 1-6.

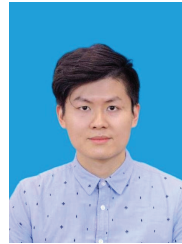
- [31] C. Liang, Y. He, F. Yu, and N. Zhao, "Enhancing QoE-aware wireless edge caching with software-defined wireless networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6912-6925, Oct. 2017.
- [32] D. Huang, X. Tao, C. Jiang, S. Cui, and J. Lu, "Trace-driven QoE-aware proactive caching for mobile video streaming in metropolis," *IEEE Wireless Commun.*, vol. 19, no. 1, pp. 62-76, Jan. 2020.
- [33] A. Mehrabi, M. Siekkinen, and A. Y. Jski, "Energy-aware QoE and backhaul traffic optimization in green edge adaptive mobile video streaming," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 3, pp. 828-839, Sep. 2019.
- [34] A. A. Zewail and A. Yener, "Device-to-device secure coded caching," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1513-1524, Sep. 2019.
- [35] M. K. Kiskani and H. R. Sadjadpour, "A secure approach for caching contents in wireless ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10249-10258, Nov. 2017.
- [36] L. Xiao, X. Wan, C. Dai, X. Du, X. Chen, and M. Guizani, "Security in mobile edge caching with reinforcement learning," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 116-122, Jun. 2018.
- [37] Z. Su, Q. Xu, J. Luo, H. Pu, Y. Peng, and R. Lu, "A secure content caching scheme for disaster backup in fog computing enabled mobile social networks," *IEEE Trans. Ind. Informat.*, vol. 14, no. 10, pp. 4579-4589, Oct. 2018.
- [38] Q. Xu, Z. Su, Q. Zheng, M. Luo, B. Dong, and K. Zhang, "Game theoretical secure caching scheme in multihoming edge computing-enabled heterogeneous networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4536-4546, Jun. 2019.
- [39] L. Xiang, D. W. K. Ng, R. Schober, and V. W. S. Wong, "Cache-enabled physical layer security for video streaming in backhaul-limited cellular networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 2, pp. 736-751, Feb. 2018.
- [40] T. Zheng, H. Wang and J. Yuan, "Secure and energy-efficient transmissions in cache-enabled heterogeneous cellular networks: Performance analysis and optimization," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5554-5567, Nov. 2018.
- [41] W. Hao, M. Zeng, G. Sun, and P. Xiao, "Edge cache-assisted secure low-latency millimeter-wave transmission," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 1815-1825, Mar. 2020.
- [42] F. Gabry, V. Bioglio, and I. Land, "On edge caching with secrecy constraints," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1-6.
- [43] C. Saha and H. S. Dhillon, "Millimeter wave integrated access and backhaul in 5G: Performance analysis and design insights," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 12, pp. 2669-2684, Dec. 2019.
- [44] R. Hamdi, E. Driouch, and W. Ajib, "New efficient transmission technique for HetNets with massive MIMO wireless backhaul," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 663-675, Jan. 2020.
- [45] L. Yang, S. Guo, L. Yi, Q. Wang, and Y. Yang, "NOSCM: A novel offloading strategy for NOMA-enabled hierarchical small cell mobile edge computing," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 8107-8118, May, 2021.
- [46] H. Zhao, Q. Zheng, W. Zhang, B. Du, and H. Li, "A segment-based storage and transcoding trade-off strategy for multi-version VoD systems in the cloud," *IEEE Trans. Multimedia*, vol. 19, no. 1, pp. 149-159, Jan. 2017.
- [47] M. Zhao, X. Gong, J. Liang, W. Wang, X. Que, and S. Cheng, "QoE-driven cross-layer optimization for wireless dynamic adaptive streaming of scalable videos over HTTP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 451-465, Mar. 2015.
- [48] V. I. Norkin and G. C. Pflug, "A branch and bound method for stochastic global optimization," *Mathematical programming*, vol. 83, pp. 425-450, Jan. 1998.
- [49] C. Li, L. Toni, J. Zou, H. Xiong, and P. Frossard, "QoE-driven mobile edge caching placement for adaptive video streaming," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 965-984, Apr. 2018.
- [50] C. Liang, Y. He, F. Yu, and N. Zhao, "Enhancing QoE-aware wireless edge caching with software-defined wireless networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6912-6925, Oct. 2017.
- [51] 3GPP, "High speed downlink packet access (HSDPA); Overall description; Stage 2, UE Radio Access capabilities," TS 25.308, v9.10.0, Release 9, 2017.



Tantan Zhao received the B.S. degree in communication engineering from Shanghai University, Shanghai, China, in 2014, and the M.S. degree in information and communications engineering from Xi'an Jiaotong University, Xi'an, China, in 2018. She is currently pursuing the Ph.D. degree with the School of Information and Communications Engineering, Xi'an Jiaotong University, China. Her research interests include video communication and transmission, mobile edge computing and deep reinforcement learning.



Lijun He received the B.S. and Ph.D. degrees from the School of Information and Communications Engineering, Xi'an Jiaotong University, Xi'an, China, in 2008 and 2016, respectively. She is currently a Associate Professor with the School of Information and Communications Engineering, Xi'an Jiaotong University. Her research interests include video communication and transmission, video analysis, processing, and compression techniques.



Xinyu Huang received the B.S. degree from the Tsien Elite Class in the Xidian University, in 2018. He is currently pursuing the M.S. degree on the Information and Communications Engineering in the Xi'an Jiaotong University. His research interests include edge computing, video transmission and deep learning.



Fan Li (M'10) obtained his B.S. and Ph.D. degrees in information engineering from Xi'an Jiaotong University, Xi'an, China, in 2003 and 2010, respectively. From 2017 to 2018, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of California, San Diego. He is currently a Professor with the School of Information and Communications Engineering, Xi'an Jiaotong University. He has published more than 60 technical papers. His research interests include multimedia communication, image/video coding and image/video quality assessment. He served as the Local Chair for ICST Wicon 2011, and was a member of the Organizing Committee for IET VIE 2008.