

Research Article

An Adaptive Video Transmission Mechanism over MEC-Based Content-Centric Networks

Longzhe Han ¹, Jia Zhao ¹, Xuecai Bao ¹, Guangming Liu ¹, Yan Liu ²,
and Taras Maksymyuk ³

¹Jiangxi Province Key Laboratory of Water Information Cooperative Sensing and Intelligent Processing, Nanchang Institute of Technology, Nanchang Institute of Technology, Nanchang 330099, China

²College of Computer and Software Engineering, East China Normal University, Shanghai 200062, China

³Department of Telecommunications, Lviv Polytechnic National University, Lviv 79013, Ukraine

Correspondence should be addressed to Longzhe Han; longzhehan@gmail.com
and Taras Maksymyuk; taras.maksymyuk@gmail.com

Received 4 March 2021; Accepted 4 July 2021; Published 17 July 2021

Academic Editor: Yuanlong Cao

Copyright © 2021 Longzhe Han et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The rapid growth of video traffic poses serious challenges to the current Internet. Content-Centric Networking (CCN) as a promising candidate has been proposed to reengineer the Internet architecture. The in-network caching and named content communication model of CCN can enhance the video streaming applications and reduce the network workload. Due to the bandwidth-consuming characteristic of video streaming, the aggressive transmission of video data will cause a reduction of overall network efficiency. In this paper, we present an adaptive video transmission mechanism over Mobile Edge Computing-(MEC-) based CCN. The computation and storage resources of the MEC server are utilized to facilitate the video delivery. Our mechanism adopts a scalable video coding scheme to adaptively control transmission rate to cope with the network condition variation. To analyse the equilibrium property of the proposed mechanism, an analytical model is deduced by using network utility function and convex programming. We also take into account the packet loss in wired and wireless links and present a MEC assistant loss recovery algorithm. The experiment results demonstrate the performance improvement of our proposed mechanism.

1. Introduction

With the continuous progress of wireless communication technology and mobile devices, emerging multimedia services (e.g., mobile TV, user-created video, video game, and mobile video calling) have gradually become people's daily applications, and dominated mobile Internet traffic [1–3]. According to Cisco's report, 79 percent of global mobile IP traffic will be video by 2022. The explosive growth of video traffic will bring huge pressure to the mobile network operators [4]. Although several technical solutions (e.g., peer-to-peer network and content delivery network) have been carried out to relieve the network transmission burden, the problem cannot be solved completely [5]. The current Internet protocols, TCP/IP,

were designed in the 1960s. The principle of TCP/IP is to interconnect two hosts across multiple physical links and exchange data. At that time, the main applications were for resource sharing, such as FTP and Telnet. The evolution of Internet applications makes TCP/IP inefficient for the new requirement, which is content sharing [6].

To address the shift of Internet use, Content-Centric Networking (CCN) has been proposed as a new network architecture of the Internet [7]. The major difference between TCP/IP and CCN is changing the protocol core from IP to named content. Instead of connecting two end hosts, the key design goal of CCN is to deliver the requested content to the user [8]. In CCN, every node is equipped with a data cache. The user announces the name of interested

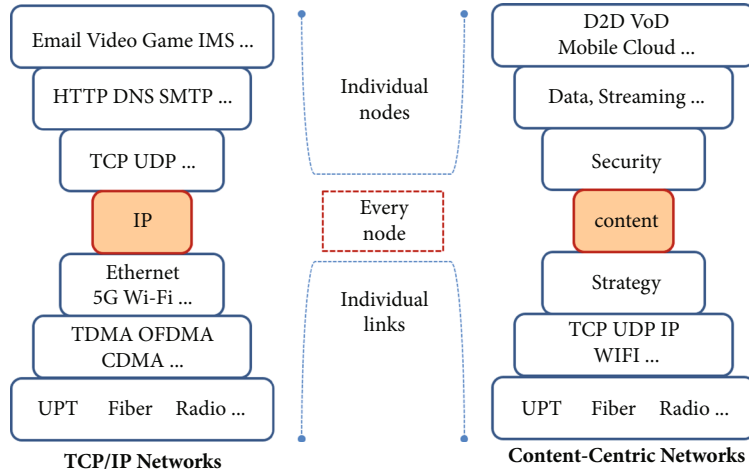


FIGURE 1: The comparison of TCP/IP protocols with CCN protocols.

information to the network, and any node possessing the data can respond [9]. Focusing on content sharing, CCN is able to dramatically improve the network transmission efficiency. Due to its promising characteristics, plentiful research efforts have been dedicated to video streaming over CCN. In [10–12], adaptive video streaming schemes in a wireless network environment are presented. The management of the data cache directly affects the availability of content; efficient cache management algorithms have been proposed [13–15].

Another crucial technique to cope with the challenge of the Internet is Mobile Edge Computing (MEC). In contrast to centralized cloud computing, MEC adopts a distributed architecture. To improve the data transmission rate, reduce network latency, and the workload of backhaul, MEC migrates the computing and storage capability of the central cloud to the edge of the network. By deployment of resource close to the end user, MEC can provide location-aware and high-speed data services. In [16, 17], algorithms are proposed to dynamically adjust the quality of experience for video streaming application in MEC. To utilize the computing resource of the MEC server, adaptive bitrate streaming approaches are presented in [18–20]. By the estimation of wireless channel and assistance of the MEC server, video quality is adapted to the wireless channel variations.

In this paper, we present an adaptive video transmission mechanism, which considering the advantages of both CCN and MEC. The main contributions of the proposed mechanism are as follows. To explore the in-network caching capability and coexist with different network applications, we introduce a transmission control algorithm to dynamically regulate the transmission rate. To adapt the network condition variation, the scalable video coding (SVC) technique is used. The encoded video has a layered structure, and various video qualities (e.g., frame rate, resolution, and fidelity) can be provided by selectively extracting enhancement layers. An analytical model is presented to verify the proposed transmission control algorithm. In addition, we identify the packet loss in wired and wireless links and propose a recovery algorithm for overcoming wireless link error.

The remainder of this paper is organized as follows. Section 2 discusses the main differences between IP network and CCN. Section 3 describes our proposed adaptive transmission control mechanism. In Section 4, the experiment environment and result analysis are presented. Finally, the summary of our work and our future research plan are described in Section 5.

2. TCP/IP Network vs. Content-Centric Network

The TCP/IP protocol suit is the de facto Internet standard [20, 21]. The success of the TCP/IP protocol suit owes to its preeminent network architecture design as depicted in Figure 1. The TCP/IP protocol suit adopts a layered structure, and intricate tasks for the data communication process are encapsulated into each layer [22, 23]. Each layer provides certain functions to its neighbour layers, such as routing, flow control, error detection, and session management. The interactions between adjacent layers are through predefined interfaces. The layered encapsulation makes the TCP/IP protocols can flexibly deal with the constant change of communication techniques [24, 25]. There are several protocols in each layer for different communication purposes, for example, TCP and UDP in the transport layer. However, the network layer only runs IP protocol, which unifies the packet forward process and builds the basis for the protocols in higher layers [26, 27].

CCN inherits the layered design principle of TCP/IP protocol suit as shown in Figure 1 [7]. The most important difference is that CCN has replaced the IP at the network layer with the named content. In TCP/IP, the function of the IP address is to virtually bind sender, receiver, and transmission packets. According to the IP address, the IP routers can forward the packets from the sender to the receiver. Nevertheless, the focus of CCN is to deliver the content to the request instead of connecting the sender and receiver. The named content is the core communication component, and CCN unbinds the content from its location. The advantage of unbinding is that the requested content can be retrieved from any node that holds the content [28].

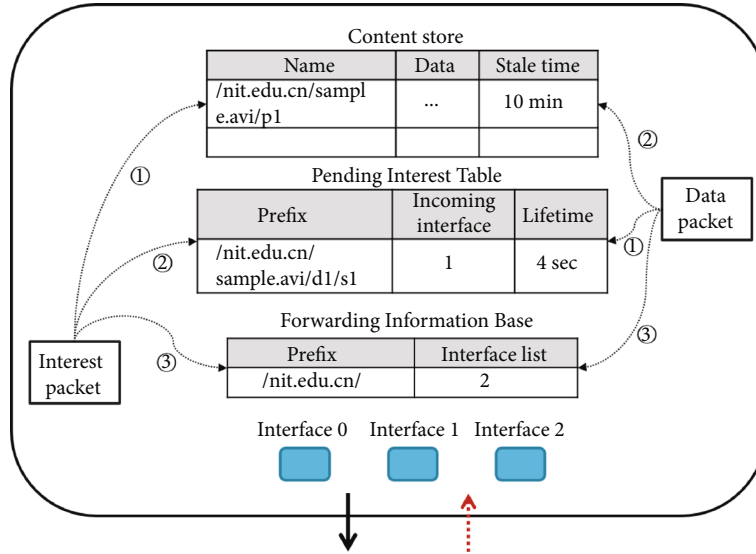


FIGURE 2: The comparison of TCP/IP protocols with CCN protocols.

In order to achieve the goal, CCN introduces new packet types and node structure as presented in Figure 2. A series of interest packets are generated and sent into the network by the receiving node, when the user wants to obtain certain content, for example, listening to a music or watching a movie. The content is encapsulated into data packets. Any CCN node holding the content can send corresponding data packets back to the request node. The CCN packets are processed only based on the name field in the packet header. The content name is recommended to take a form of tree structure, such as the form of Uniform Resource Identifier [29].

A typical CCN node has three data modules: Pending Interest Table (PIT), Content Store (CS), and Forwarding Information Base (FIB). The PIT records every interest packet that the node forwarded and the forwarded interface. When the interest packet arrives at the content holder node, the PIT entries of the forwarding nodes set up a virtual path for the data packet traveling back. The PIT also has a role of transmission rate control and loss recovery. Since content is unbanded with its location, the CCN node can reuse the forwarded data packets other than eliminate them [30]. The CS is a memory space of a CCN node for buffering data packets. To improve the efficiency, memory replacement algorithms are used to manage the CS. The FIB has the same function as the routing table in the IP router. The CCN node references the FIB to forward interest packets to successive nodes and periodically updates the FIB according to routing algorithms.

Figure 2 shows three steps when a CCN node receives an interest packet. The first step is to search the CS by using the name field as the key [7]. If the matched data packet is found, then the data packet is sent back through the incoming network interface, and the interest packet is abandoned. This means that the content has been transmitted by this node and the content is reused. If there are no matched data packets in the CS, the second step is searching the PIT. If there is a matched entry in the PIT, then the same request has been forwarded before. The CCN node appends the

incoming network interface to the existing entry and waits the return of the data packet. In the case of no matched entry in the PIT, the third step is to look up the FIB. If an entry with the same name is found in FIB, then the interest packet is forwarded to the subsequent node. Otherwise, the interest packet is abandoned [30].

There are also three steps for the incoming data packet. The first step is to look up the PIT. If there is a match, then the data packet has been requested. The CCN node sends the data packet to each network interface listed in the entry, and the entry is removed from the PIT after sending. In case of no match in the PIT, the data packet is abandoned and the process is finished. The reason is that malicious nodes may broadcast junk content into the network, which causes the contamination of the CS and useful content cannot be stored. In the second step, the content is buffered in the CS for future reuse. The third step is to check the FIB by using the name of the data packet. If there is no match, the name and incoming interface are added into the FIB. The third step can help the routing algorithm to update the FIB in a distributed manner. After that, the whole process is finished [31].

The content caching capability of CCN nodes is crucial for video streaming applications. In [32], the authors present a progressive caching algorithm for video streaming over CCN. The algorithm generates a metafile that includes the content priority information, and the CCN nodes can decide the cache policy based on the metafile. In [8], a multisource video streaming algorithm is proposed. Due to the caching capability of CCN nodes, the receivers might receive the data from different nodes. By considering the quality of experience, the proposed algorithm switches to different sources. [13] utilizes a scalable video coding scheme and proposes caching algorithms to improve the video delivery services. The content centric communication model is used not only for static networks but also for high dynamic networks [33–35]. In [36], the authors show the feasibility of video streaming over CCN-based vehicular networks.

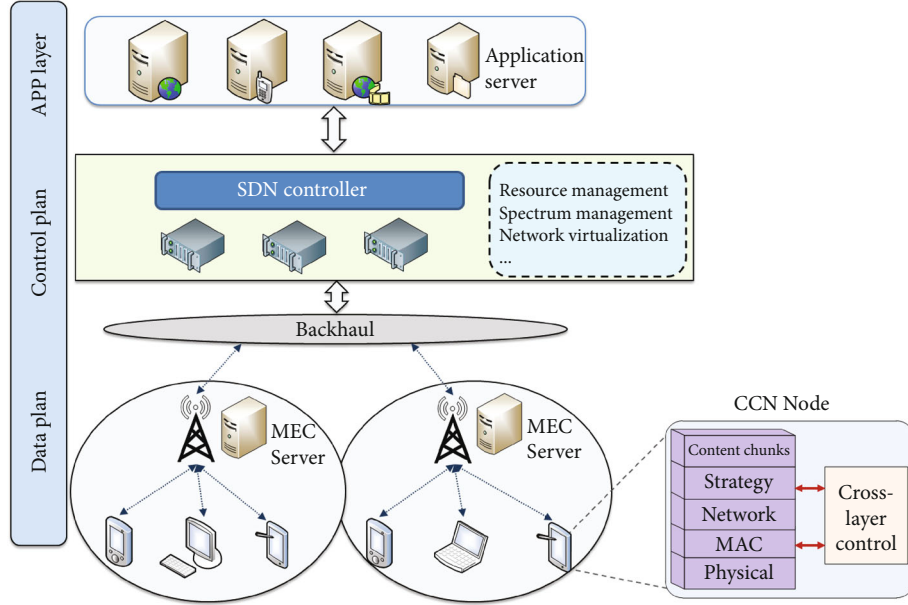


FIGURE 3: The system architecture of MEC-based Content-Centric Network.

3. Adaptive Video Transmission Mechanism

3.1. Architecture of Video Streaming System. The system architecture of our proposed mechanism is presented in Figure 3. For the flexible configuration of network resources, the proposed video streaming system integrates the SDN scheme. SDN consists of the control plan, data plan, and application layer. The application layer includes various types of application servers, for example, video streaming server, web server, and mail server. The different applications have different requirements for quality of service. In addition, CCN is independent from particular transport protocols; any protocol with the function of packet transmission, such as TCP, UDP, IP, and P2P, can be used. The SDN control plan uniformly and flexibly schedules network resources and provides a unified programming interface for the management program to support programmable control. The data plane including the backhaul network and access network is only responsible for data forwarding. Through the encapsulation of control functions and the abstraction of applications and networks, SDN treats the network as a logical or virtual entity to form an architecture similar to a computer operating system. The data plane provides a standardized and open interface for the control plane. In this way, the control plane can achieve more flexible control capabilities based on the global network view, and the data plan and application layer can be flexibly and independently expanded according to actual requirements to meet changing needs.

3.2. Adaptive Transmission Control Algorithm. CCN follows the receiver-driven model, where the receiver takes a major role during the video transmission. Because CCN does not assume the transport layer protocols offering reliable data communication service, flow control, congestion control, etc. It is the receiver's responsibility to regulate its communi-

cation policy to respond to network condition variation. Our proposed transmission control algorithm is based on the receiver-driven model and TCP flow control and congestion control scheme. According to the video bitrate, the receiver periodically sends interest packets for retrieving video data. The ongoing interest packets are stored in the PIT. For each PIT entry, there is a timer with a value called LifeTime. If no data packet is received within LifeTime, either interest or data packet is lost. We consider the time between sending an interest packet and receiving the corresponding data packet as the round trip time (RTT) of the transmission. The control variable W_{PIT} indicates the length of the PIT, which is the number of ongoing interest packets allowed to be sent. When the number of PIT entries reaches W_{PIT} , the sending of interest packets is suspended until receiving a data packet. Initially, the W_{PIT} is set to satisfy the required bitrate of the base layer denoted W_{base} . When a data packet is received, the W_{PIT} is increased proportionally to its current value.

As shown in Figure 4, the communication path between the receiver and the video server contains wired and wireless links. The loss of interest and data packet can happen in any links. In wired links, the CCN nodes use queue management algorithms that actively dropping packets to prevent network congestion, for instance, random early detection, random exponential marking, and stochastic fair BLUE. In wireless links, the packet loss is mainly caused by transmission errors. To distinguish between wired and wireless link loss, the MEC server traces the wireless loss rate denoted as $P_{wireless}$ and sets into the data packet header. The receiver can obtain the total packet loss rate P_{total} , by observing the event of the PIT timeout. If there is a timeout and P_{total} is equal to $P_{wireless}$, it means that packet loss is at wireless links and W_{PIT} remains unchanged. If P_{total} is larger than $P_{wireless}$, then certain nodes along the links are expected to encounter congestion. The receiver needs to reduce the transmission rate; therefore,

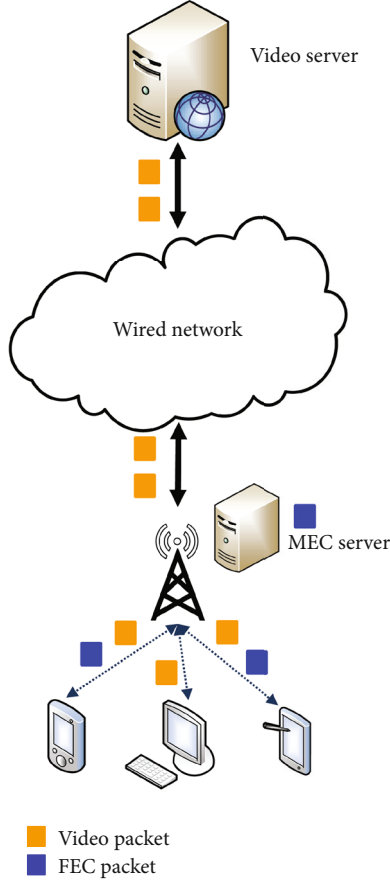


FIGURE 4: Data offloading and computation offloading for video streaming over MEC-based Content-Centric Network.

W_{PIT} is set to W_{base} . We can use ordinary differential equation to model the dynamics of the proposed transmission algorithm as

$$\dot{W}_{\text{PIT}} = \frac{x(t)(1 - P_{\text{total}} + P_{\text{wireless}})\alpha}{W_{\text{PIT}}} - x(t)(P_{\text{total}} - P_{\text{wireless}})W_{\text{base}}, \quad (1)$$

where $x(t)$ is the transmission rate of the receiver and $x(t) = W_{\text{PIT}}/\text{RTT}$. α is the increasing rate of W_{PIT} . With the replacement, the derivative of the transmission rate is

$$\dot{x}(t) = \frac{(1 - P_{\text{total}} + P_{\text{wireless}})\alpha}{\text{RTT}^2} - x(t)(P_{\text{total}} - P_{\text{wireless}})x_{\text{base}}, \quad (2)$$

where $x_{\text{base}} = W_{\text{base}}/\text{RTT}$ is the transmission rate of the base layer.

3.3. Analytical Model of Proposed Algorithm. To analyse the equilibrium behaviour of the proposed transmission control algorithm, we introduce a network utility function $U_i(x_i)$ for the i^{th} receiver. The network utility function is a mapping of user benefit to the transmission rate. By increasing the transmission rate, the user can acquire more enhancement

```

1: if arriving data packet then
2:   obtain packet loss rate  $P_{\text{wireless}}$  from packet header
3:    $W_{\text{PIT}} = (1 + \alpha)W_{\text{PIT}}$ 
4:   clear timeout counter  $n = 0$ .
5: end if
6: if PIT timeout then
7:   calculate total packet loss rate  $P_{\text{total}}$ .
8:   increase timeout counter  $n = n + 1$ 
9: if  $P_{\text{total}} > P_{\text{wireless}}$  then
10:  if  $W_{\text{PIT}} > W_{\text{base}}$  then
11:     $W_{\text{PIT}} = W_{\text{base}}$ 
12:  else
13:     $W_{\text{PIT}} = \beta^n W_{\text{base}}$ 
14:  end if
15: else
16:  calculate number of FEC packet  $W_{\text{FEC}}$ 
17:  send FEC interest packets to overcome loss
18: end if
19: end if
  
```

ALGORITHM 1: Adaptive Transmission Control Algorithm.

layers, and the user benefit is also increased. Hence, the network utility function is assumed to be concave and differentiable. Consider a wireless network with N receivers and K links, the transmission control problem is presented as network utility maximization [37]:

$$\begin{aligned} & \underset{x_i > 0}{\text{maximize}} && \sum_{i=1}^N U_i(x_i), \\ & \text{subject to} && \sum_{i=1}^K d_{ji}x_i \leq C_j, \end{aligned} \quad (3)$$

where C_j is the link capacity and the inequality constraint means that the total transmission rate should be not greater than the link capacity. d_{ji} is the portion of transmission flow carried by the link j . Let λ_j be the Lagrange multiplier for the capacity constraint of the link j . The Lagrangian is defined as

$$L(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{i=1}^N U_i(x_i) - \sum_{j=1}^K \lambda_j \left(\sum_{i=1}^N d_{ji}x_i - C_j \right), \quad (4)$$

where the last term can be considered as the penalty if the total transmission rate exceeds the link capacity. Because the object function of (3) is concave and the inequality constraint is linear, the network utility maximization forms a convex optimization problem. From Karush–Kuhn–Tucker (KKT) conditions, we can derive from taking the derivative of $L(\mathbf{x}, \boldsymbol{\lambda})$ respect to x_i :

$$U_i'(x_i^*) - \sum_{j=1}^K \lambda_j^* d_{ji} = 0. \quad (5)$$

At the equilibrium point, the derivative of the transmission rate (2) is zero. The packet loss caused by the congestion

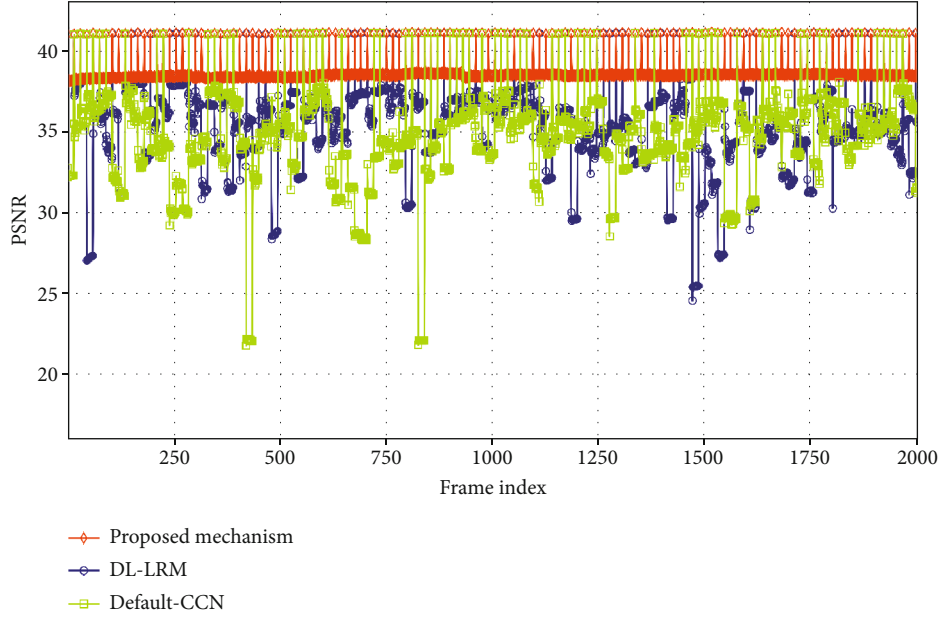


FIGURE 5: Video quality with no wireless link error.

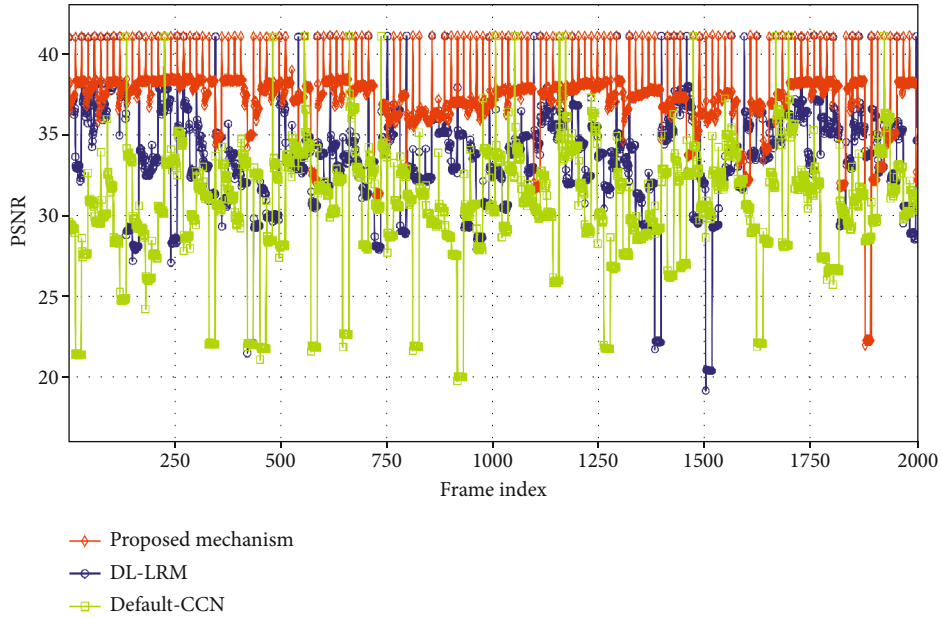


FIGURE 6: Video quality with 8% wireless link error rate.

is also a penalty; with (5), we can obtain

$$U'_i(x_i) = \frac{1}{\alpha + \text{RTT}^2 x_{\text{base}} x_i}. \quad (6)$$

Then, the utility function is

$$U_i(x_i) = \frac{\log(\alpha + \text{RTT}^2 x_{\text{base}} x_i)}{\text{RTT}^2 x_{\text{base}}}. \quad (7)$$

Although there are different optimization algorithms [38–40], with the analytical expression of the utility function, the convex optimization algorithm is carried out to find the optimal solution.

3.4. MEC Assistant Link Loss Recovery. As explained in the previous section, the packet loss can happen in wired and wireless links. Usually, the loss in the wired link is caused by the routing nodes proactively dropping packets to prevent the link congestion. On the other hand, the loss in the wireless link is due to wireless link error. The packet loss can be

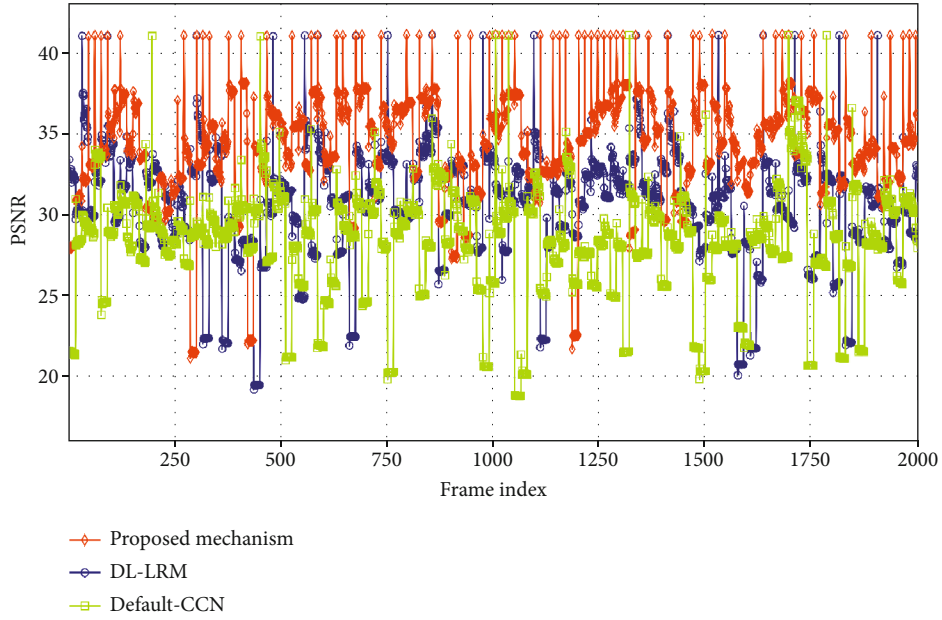


FIGURE 7: Video quality with 12% wireless link error rate.

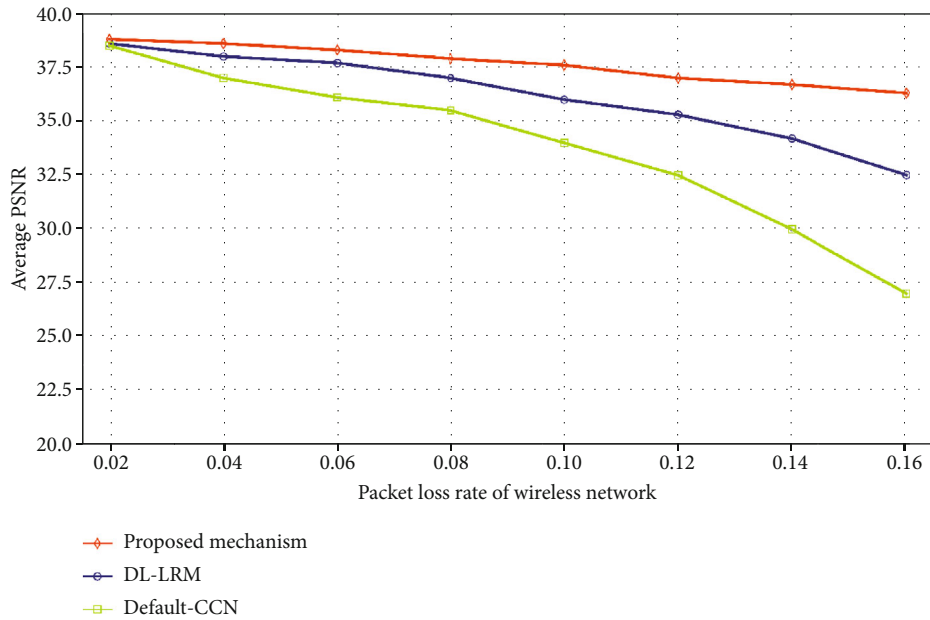


FIGURE 8: Average video quality with various wireless link error rates.

recovered by Automatic Repeat reQuest (ARQ) or forward error correction. In our proposed system scheme as shown in Figure 4, the MEC server, routing nodes, and the video server are able to store the video content. The receiver may obtain data packets from different nodes; hence, the ARQ is improper. Our proposed mechanism adopts the FEC approach to recover packet loss in the wireless link. The generation of the FEC packet is offloaded to the MEC server from the video server. Since all data packets must go through the wireless base station to the receiver. The MEC server collects the video data and generates FEC packets. The FEC packet is retrieved from the MEC server instead of the video

server; the workload of wired links is vastly reduced. When the receiver gets the wireless loss rate from the MEC server, the minimum number of FEC packets required to overcome the loss is

$$W_{\text{FEC}} = \frac{W_{\text{PIT}} P_{\text{wireless}}}{1 - P_{\text{wireless}}}, \quad (8)$$

where $W_{\text{PIT}} P_{\text{wireless}}$ is the expected number of packet loss. Depending on the current transmission rate W_{PIT} , the receiver sends interest packets for FEC. The detailed steps of our proposed mechanism are presented in Algorithm 1.

4. Experiment Results and Analysis

To analyse the performance of our proposed mechanism, a series of experiments were carried out with NS-3 simulator [41]. We used H.264/scalable video coding to compress the raw video sequences “Elephants Dream” [42]. The proposed mechanism was compared with DL-LRM [35] and CCN default approach [7] with the same network conditions. The encoded video consisted of one base layer and three enhancement layers. The network topology of experiments included wired and wireless links. The links from the video server to the base stations were wired and the bandwidth was 1 Gigabit per second. The base stations communicated with end nodes through wireless link, and the bandwidth was 100 Megabits per second. There were 200 end nodes, and the nodes were divided into 10 groups. Each group node requested the same video content to simulate network congestion. The experiments were conducted under different packet loss rates to simulate wireless link error.

Figure 5 shows the video quality of receivers with no wireless link error. Because the DL-LRM and CCN default approach have no control of transmission rate, even the network is congested, the two approaches still aggressively send interest packets. This results in numerous packets being dropped by routing nodes and decreasing video quality.

With increasing wireless link error rate, the number of packet loss is also increased. As described in Section 2, each PIT entry has a timer which value is called LifeTime. The LifeTime is set by the receiver, and it should be larger than the upper bound of RTT samples. The reason is that every node in CCN has a PIT. If a timeout arises at an end node, the node may retransmit the interest packet. If a timeout arises at a routing node, the PIT entry is deleted. Later, if the data packet arrives, and there is no corresponding PIT entry, then the data packet is abandoned. On account of the in-network caching function of CCN, the data packets may be retrieved from different nodes and the RTT value has high fluctuation. The default CCN approach for packet loss is ARQ, and the LifeTime is set to a constant value. The video streaming application is delay sensitive. If the loss packet is recovered, nonetheless it exceeds the playtime, then the packet is useless. From experiment results shown in Figures 6 and 7, the performance of the default CCN approach drops dramatically with the increase of the error rates.

Our proposed mechanism distinguishes the cause of packet loss. If the network is congested, then the transmission is limited to the request for the base layer only to relieve the congestion. If the loss is caused by wireless link error, FEC packets are requested. The DL-LRM does not identify the reason of packet loss and just sends FEC packets to respond to the PIT timeout. As presented in Figure 8, our proposed mechanism outperforms the DL-LRM and default CCN approach under different network conditions.

5. Conclusions

In this paper, we have presented an adaptive video transmission mechanism in Mobile Edge Computing- (MEC-) based

CCN. The main contribution of our mechanism is to adaptively control transmission rate to adapt to the network conditions variation, instead of greedily acquiring video content. Therefore, it can friendly coexist with different types of applications and improve overall network efficiency. We adopt scalable video coding (SVC) scheme to balance the video quality and transmission rate. In addition, the MEC assistant loss recovery algorithm is suggested to reduce the burden of the video streaming server and backhaul network. The experiment results show that our proposed mechanism outperforms the existing approaches. For future work, we will study the scalability issue and analyse the performance in a large network environment.

Data Availability

The data used to support the study are available within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research is jointly supported by the Science and Technology Research Project of Jiangxi Provincial Department of Education (No. GJJ190957), by the National Natural Science Foundation of China (No. 61962036, No. 52069014, No. 61961026), and by the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry.

References

- [1] D. Ghadiyaram, J. Pan, and A. C. Bovik, “A subjective and objective study of stalling events in mobile streaming videos,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 1, pp. 183–197, 2019.
- [2] Y. Cao, L. Zeng, Q. Liu, G. Lei, M. Huang, and H. Wang, “Receiver-assisted partial-reliable multimedia multipathing over multi-homed wireless networks,” *IEEE Access*, vol. 7, pp. 177675–177689, 2019.
- [3] H. Wang, J. Liang, and C. C. J. Kuo, “Overview of robust video streaming with network coding,” *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 1, pp. 36–50, 2010.
- [4] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017-2022*, White Paper, 2019.
- [5] Y. Zhang, C. Gao, Y. Guo et al., “Proactive video push for optimizing bandwidth consumption in hybrid CDN-P2P VoD systems,” *IEEE INFOCOM*, vol. 2018, pp. 2555–2563, 2018.
- [6] C. Fang, H. Yao, Z. Wang, W. Wu, X. Jin, and F. R. Yu, “A survey of mobile information-centric networking: research issues and challenges,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2353–2371, 2018.
- [7] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, “Networking named content,” in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pp. 1–12, Rome, Italy, 2009.

- [8] L. Han, X. Bao, W. Wang, X. Feng, Z. Liu, and W. Tan, "A receiver-driven loss recovery mechanism for video dissemination over information-centric VANET," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 7, 2017.
- [9] M. N. Sadat, R. Dai, L. Kong, and J. Zhu, "QoE-aware multi-source video streaming in content centric networks," *IEEE Transactions on Multimedia*, vol. 22, no. 9, pp. 2321–2330, 2020.
- [10] R. Jia, Z. Liu, X. Wang, X. Gan, X. Wang, and J. J. Xu, "Modeling dynamic adaptive streaming over information-centric networking," *IEEE Access*, vol. 4, pp. 8362–8374, 2016.
- [11] C. Xu, P. Zhang, S. Jia, M. Wang, and G. Muntean, "Video streaming in content-centric mobile networks: challenges and solutions," *IEEE Wireless Communications*, vol. 24, no. 5, pp. 157–165, 2017.
- [12] S. Lederer, C. Mueller, C. Timmerer, and H. Hellwagner, "Adaptive multimedia streaming in information-centric networks," *IEEE Network*, vol. 28, no. 6, pp. 91–96, 2014.
- [13] Z. Zhang, J. Dai, M. Zeng, D. Liu, and S. Mao, "Scalable video caching for information centric wireless networks," *IEEE Access*, vol. 8, pp. 77272–77284, 2020.
- [14] W. Li, S. M. A. Oteafy, and H. S. Hassanein, "Rate-selective caching for adaptive streaming over information-centric networks," *IEEE Transactions on Computers*, vol. 66, no. 9, pp. 1613–1628, 2017.
- [15] T. Zhang, X. Xu, Z. Le, X. Jiang, and J. Loo, "Cache Space Efficient Caching Scheme for Content-Centric Mobile Ad Hoc Networks," *IEEE Systems Journal*, vol. 13, no. 1, pp. 530–541, 2019.
- [16] A. Younis, T. X. Tran, and D. Pompili, "On-demand video-streaming quality of experience maximization in mobile edge computing," in *2019 IEEE 20th International Symposium on "A World of Wireless, Mobile and Multimedia Networks*, pp. 1–9, Washington DC, United States, 2019.
- [17] M. F. Tuysuz and M. E. Aydin, "QoE-based mobility-aware collaborative video streaming on the edge of 5G," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 11, pp. 7115–7125, 2020.
- [18] J. Luo, F. R. Yu, Q. Chen, and L. Tang, "Adaptive video streaming with edge caching and video transcoding over software-defined Mobile networks: a deep reinforcement learning approach," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1577–1592, 2020.
- [19] Y. Guo, F. R. Yu, J. An, K. Yang, C. Yu, and V. C. M. Leung, "Adaptive bitrate streaming in wireless networks with transcoding at network edge using deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 3879–3892, 2020.
- [20] Y. Li, D. Leith, and R. N. Shorten, "Experimental evaluation of TCP protocols for high-speed networks," *IEEE/ACM Transactions on Networking*, vol. 15, no. 5, pp. 1109–1122, 2007.
- [21] Y. Cao, J. Chen, Q. Liu, G. Lei, H. Wang, and I. You, "Can multipath TCP be robust to cyber attacks with incomplete information?," *IEEE Access*, vol. 8, pp. 165872–165883, 2020.
- [22] K. Sasaki, M. Hanai, K. Miyazawa, A. Kobayashi, N. Oda, and S. Yamaguchi, "TCP fairness among modern TCP congestion control algorithms including TCP BBR," in *2018 IEEE 7th International Conference on Cloud Networking*, pp. 1–4, Tokyo, Japan, 2018.
- [23] K. Ratna Pavani and N. Sreenath, "Performance evaluation of TCP-Reno, TCP-Newreno and TCP-Westwood on burstification in an OBS network," in *2012 18th International Conference on Advanced Computing and Communications*, pp. 19–24, Bangalore, India, 2012.
- [24] R. Amrutha and V. Nithya, "Performance analysis of TCP incast with TCP Lite and Abstract TCP," in *2015 Global Conference on Communication Technologies*, pp. 18–22, Thuckalay, India, 2015.
- [25] P. Chaudhary and S. Kumar, "Comparative study of TCP variants for congestion control in wireless network," in *2017 International Conference on Computing, Communication and Automation*, pp. 641–646, Greater Noida, India, 2017.
- [26] M. Polese, R. Jana, and M. Zorzi, "TCP and MP-TCP in 5G mmWave networks," *IEEE Internet Computing*, vol. 21, no. 5, pp. 12–19, 2017.
- [27] Y. Cao, M. Collotta, S. I. Xu, L. Huang, X. Tao, and Z. Zhou, "Towards adaptive multipath managing: a lightweight path management mechanism to aid multihomed mobile computing devices," *Applied Sciences*, vol. 10, no. 1, 2020.
- [28] N. Kamiyama, "Analyzing impact of introducing CCN on profit of ISPs," *IEEE Transactions on Network and Service Management*, vol. 12, no. 2, pp. 176–187, 2015.
- [29] A. Aboodi, T. Wan, and G. Sodhy, "Survey on the incorporation of NDN/CCN in IoT," *IEEE Access*, vol. 7, pp. 71827–71858, 2019.
- [30] R. Jmal and L. Chaari Fourati, "Content-centric networking management based on software defined networks: survey," *IEEE Transactions on Network and Service Management*, vol. 14, no. 4, pp. 1128–1142, 2017.
- [31] Y. Wang, Z. Li, G. Tyson, S. Uhlig, and G. Xie, "Design and evaluation of the optimal cache allocation for content-centric networking," *IEEE Transactions on Computers*, vol. 65, no. 1, pp. 95–107, 2016.
- [32] H. Noh and H. Song, "Progressive caching system for video streaming services over content centric network," *IEEE Access*, vol. 7, pp. 47079–47089, 2019.
- [33] F. Khan and H. Li, "Ensuring trust and confidentiality for adaptive video streaming in ICN," in *Journal of Communications and Networks*, vol. 21, no. 6, pp. 539–547, 2019.
- [34] Z. Zhang, C. Lung, M. St-Hilaire, and I. Lambadaris, "An SDN-based caching decision policy for video caching in information-centric networking," in *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 1069–1083, 2020.
- [35] X. Maksymyuk, J. Zhao, and Y. Liu, "Deep learning based loss recovery mechanism for video streaming over mobile information-centric network," *KSII Transactions on Internet and Information Systems*, vol. 13, no. 9, pp. 4572–4586, 2019.
- [36] Z. Zhang, C. Lung, M. St-Hilaire, and I. Lambadaris, "Smart proactive caching: empower the video delivery for autonomous vehicles in ICN-based networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7955–7965, 2020.
- [37] F. Paganini, Z. Wang, J. C. Doyle, and S. H. Low, "Congestion control for high performance, stability, and fairness in general networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 1, pp. 43–56, 2005.
- [38] J. Pan, N. Liu, S. Chu, and T. Lai, "An efficient surrogate-assisted hybrid optimization algorithm for expensive optimization problems," *Information Sciences*, vol. 561, pp. 304–325, 2021.
- [39] L. Lv, J. Zhao, J. Wang, and T. Fan, "Multi-objective firefly algorithm based on compensation factor and elite learning," *Future Generation Computer Systems*, vol. 91, pp. 37–47, 2019.

- [40] J. Z. Xie, L. Lv, H. Wang, H. Sun, and X. Yu, "Firefly algorithm with deep learning," *Acta Electronica Sinica*, vol. 46, no. 11, pp. 2633–2641, 2018.
- [41] G. F. Riley and T. R. Henderson, *The NS-3 Network Simulator, Modeling and Tools for Network Simulation*, Springer, Berlin, Heidelberg, 2010.
- [42] *Elephants Dream* Video Test Media Collection <https://media.xiph.org/video/derf/>.